

Mobile-C

1.10.12

Generated by Doxygen 1.6.1

Fri May 14 13:39:38 2010

Contents

Chapter 1

MobileC

1.1 Introduction

Welcome to the MobileC documentation. For a quick user-api reference, please refer to the file [libmc.h](#).

This documentation is provided supplementary to the main Mobile-C User Guide. The official user guide may be obtained at <http://www.mobilec.org> , and should also be included with your Mobile-C Distribution package in the file docs/mobilec.pdf .

Chapter 2

Installing LibMC.NET

Installing LibMC.NET is straightforward but involves several steps.

2.1 Requirements

In order to use LibMC.NET you will need the following:

- Ch version 6.0.0 or greater from <http://www.softintegration.com/>.
- Embedded Ch version 6.0.0 or greater, also from <http://www.softintegration.com/>.
- Mobile-C 1.10.0 or greater. See Section [Downloading Mobile-C](#) for instructions on how to obtain Mobile-C.
- Visual Studio 2005 or later. Express versions of Visual Studio can be found at <http://www.microsoft.com/express/>.

2.2 Downloading Mobile-C

First, you must obtain a version of the Mobile-C source code. If you are reading this, chances are you have already completed this step. If you have not already downloaded the source code, it can be done in one of three ways:

- Download a supported release of Mobile-C. Visit [the Mobile-C website](#) for more information on supported releases.
- Download the latest source (unsupported) from [Sourceforge](#). This will give you the most current version of Mobile-C, but not necessarily the most stable version.
- Check out the latest source code from the SVN repository. This requires that you have a subversion client installed. More information can be found at [this location](#).

2.3 Building the Mobile-C Libraries

Once you have obtained the Mobile-C source, please see the Mobile-C User's Guide for information on compiling Mobile-C under Windows. Currently, only the Visual Studio .NET 2005 project is supported for LibMC.NET. Section 2.3 of the User's Guide describes how to compile Mobile-C into a static library. For LibMC.NET, at least one of two configurations are required: the "Debug_DLL" or "Release_DLL" versions. To build either one, select the appropriate configuration (this replaces step 3 in the User's Guide, Section 2.3.1) and build the solution (step 4). Alternatively, you may select "Batch Build" from the "Build" menu and build all four possible configurations.

2.4 Install the Mobile-C Libraries

After building the Mobile-C DLL files, the project will automatically copy the files to the system directory. By default, the files are copied to C:/Windows/System32/. If your system is configured differently or you wish to change the installation directory, right-click on the mc_lib_win32 project in the Solution Explorer and select "Properties." In the mc_lib_win32 Property Pages treeview, select "Configuration Properties," then "Build Events," and finally "Post-Build Event." You can then change the "Command Line" field to copy the files to the directory of your choice. If you change the installation directory, be sure that your

chosen directory is in the system path and that you remove any other versions of the files. You will also need to execute a "Rebuild" on the project to ensure the files are copied to the new location.

2.5 Build LibMC.NET

Once you have built the Mobile-C DLL files, you can build LibMC.NET. Open the LibMC.NET solution file located in the directory you installed or checked out Mobile-C to at src/win32/LibMC.NET/LibMC.Net.sln. From the "Build" menu, select "Rebuild Solution." You may want to build both the "Debug" and "Release" versions, or perform a batch build as described previously.

Chapter 3

Getting Started

LibMC.NET is very easy to use. The demo programs provided with the download are a good place to start. Please see the Examples section for more information.

3.1 Build the Demo Programs

The LibMC.NET demo program solution is located in the directory you installed or checked out Mobile-C to at `demos/win32/LibMC.NET/LibMCDemos.sln`. As before, select "Rebuild Solution" from the "Build" menu. Note that the demo program solution contains the LibMC.NET project as well. You may also build LibMC.NET from within the demo program solution.

By default, the LibMCGui demo is selected in the demo program solution. You may run this program by selecting the "Debug" menu then "Start Debugging" or by pressing F5. Other demo programs can be started by right-clicking the project in the Solution Explorer and selecting "Debug" then "Start new instance." The demo programs have their own documentation as well. See the README file in `src/win32/LibMC.NET` or `demos/win32/LibMC.NET` for information on how to build the demo program documentation.

Chapter 4

Using LibMC.NET

This section explains how to use the LibMC.NET class library in your .NET project. Currently, it only describes the process for using the library in a C# console or GUI application. Other languages, such as VB and managed C++, will require similar actions.

4.1 Create a Project

First, create the type of project you would like to use from the Visual Studio "Start Page" or the "File" menu. Select the name and location of the project as you would any other project. Second, add a reference to the configuration of LibMC.NET you would like to use. For debugging purposes, the "Debug" configuration is probably best. To add the reference, right click the "References" item in the Solution Explorer for the project you just created. Select the "Browse" tab and navigate to the output directory of the LibMC.NET project. The directory is located at `src/win32/LibMC.NET/bin/Configuration/` in the Mobile-C source directory, where Configuration is either "Debug" or "Release." Select the DLL file and click "Ok." The References item in the Solution Explorer should now list "LibMC." Be sure to save the solution at this point.

4.2 Using LibMC.NET Classes and Functions

As with any other namespace, you must add the declaration "using LibMC;" to any file you want to have access to the class libraries. Once you have added the using statement, you can declare objects from the library as you normally would declare any other objects. See the example programs for more details.

4.3 Other Options

You may want to enable one or more features in your project that can help you use LibMC.NET or debug problems. If you add any XML files to your project, you probably will want to set their properties in the project to copy the files to the output directory. This is done by selecting the file in the Solution Explorer, opening its properties, and setting two fields:

- Set the "Build Action" field to "Content" if it is not already set. This will make the file part of the project should you decide to publish or package it.
- Set the "Copy to output directory" to "Copy if newer" or "Copy always." This will copy the file when you build the project.

There is also one important note regarding XML files in Visual Studio. *Do not create XML files from within Visual Studio.* The Visual Studio XML file template contains a few leading characters that specify the encoding of the file. They are hidden and you will not be able to change them. These characters are not currently supported by Mobile-C and will crash a receiving agency.

To open the project properties, right-click the project in the Solution Explorer and select "Properties." In the "Debug" pane, you may wish to set an alternate working directory for the project if you want easy access to XML files outside of the project. This is useful for debugging, but may result in errors finding files if you package the project or create an installer. In general, it is best to specify all files with full paths because the Mobile-C library loads from a different location than the project. If you would like to be able to debug the Mobile-C library, you should select the "Enable unmanaged code debugging" check box. This will allow you to more easily see any errors that may occur in the unmanaged library, though hopefully none will.

Chapter 5

Common Operations

This section contains examples of commonly used operations for three main [LibMC](#) classes:

- [MCAgency](#) The mobile agent agency.
- [MCAgent](#) Mobile agents.
- [MCAclMessage](#) Agent communication language messages.

For complete programs and more detailed examples, see the Examples section.

5.1 MCAgency

Examples of commonly used MCAgency operations:

Declare an agency as a member of a class:

```
public static MCAgency Agency = new MCAgency();
```

Set the agency's port:

```
int temp = 5051;
Agency.Port = temp;
```

Start an agency:

```
int temp;
temp = Agency.Initialize();
if (temp != 0)
    Console.WriteLine("Initialize: " + temp.ToString());
```

Pause and resume an agency:

```
Agency.HaltAgency();
Agency.ResumeAgency();
```

Turn off the command prompt thread:

```
temp = Agency.SetThreadOff(MCAgency.MC_ThreadIndex_e.MC_THREAD_CP);
if (temp != 0)
    Console.WriteLine("SetThreadOff: " + temp.ToString());
```

Load an agent into a local agency:

```
String filename = "agent.xml";
try
{
    Agency.LoadAgentMigrationMessageFile(filename);
}
catch (Exception ex)
{
    Console.WriteLine("Error loading file: " + ex.Message);
}
```

Ideally, the file name should be specified absolutely.

Send an agent to a remote agency:

```
String filename = "agent.xml";
String ip = "192.168.23.93";
int port = 5051;
try
{
    Agency.SendAgentMigrationMessageFile(filename, ip, port);
}
catch (Exception ex)
{
    Console.WriteLine("Error sending file: " + ex.Message);
}
```

Find an agent by name:

```
MCAgent agent;
try
{
    agent = Agency.FindAgentByName("persistent1");
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Wait for an agent to arrive:

```
MCAgent agent;
Agency.ResetSignal();
try
{
    agent = Agency.WaitRetrieveAgent();
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Wait indefinitely while an agency runs:

```
Agency.MainLoop();
```

5.2 MCAgent

Examples of commonly used MCAgent operations: Find an agent by name (assumes an MCAgency named Agency):

```
MCAgent agent;
try
{
    agent = Agency.FindAgentByName("persistent1");
}
catch (Exception e)
{
    Console.WriteLine("Exception: " + e.Message);
}
```

Terminate an agent

```
int temp;
try
{

```

```

        temp = agent.TerminateAgent();
        Console.WriteLine("TerminateAgent() returned " +
            temp.ToString() + ".");
    }
    catch (Exception e)
    {
        Console.WriteLine("Exception: " + e.Message);
    }

```

Print information about an agent:

```

Console.WriteLine(agent.ToString());
Console.WriteLine(agent.GetAgentXMLString());
Console.WriteLine(agent.RetrieveAgentCode());

```

5.3 MCAclMessage

Examples of commonly used MCAclMessage operations:

Create a new, blank ACL message:

```

MCAclMessage tmp = new MCAclMessage();
tmp.New();

```

Set the performative field:

```

tmp.SetPerformative(MCAclMessage.MC_FipaPerformative_e.FIPA_INFORM);

```

Set the sender:

```

tmp.SetSender("agency", "http://" + host + ":" +
    localport.ToString() + "/acc");

```

Add an alternate reply-to field:

```

tmp.AddReplyTo("mobagent2", "http://" + host + ":" +
    localport.ToString() + "/acc");

```

Add a receiver to the message:

```

tmp.AddReceiver("mobagent1", "http://" + host + ":" +
    localport.ToString() + "/acc");

```

Set the content of the message:

```

tmp.SetContent("This is content. Yay!");

```

Finally, send and destroy the message:

```

Agency.AclSend(tmp);
tmp.Destroy();

```

Note that messages contain a pointer to allocated unmanaged memory and need to be disposed of after they are used. The agency creates a copy of the message when it is sent, and therefore the MCAclMessage object is no longer needed.

Chapter 6

Todo List

Global **LibMC::MCAgency::RegisterService**(MCAgent agent, int agentID, String agentName, String[] serviceNames, int agentType)
Test MC_RegisterService and MC_SearchForService.

Global **LibMC::MCAgency::SearchForService**(String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentType)
Implement SearchForService

Global **LibMC::MCAgency::Steer**(IntPtr funcptr, IntPtr arg) Implement MC_Steer

Global **LibMC::MCAgency::SteerControl**() Test MC_SteerControl, MC_Steer.

Global **LibMC::MCAgent::GetAgentExecEngine**() Wrap MC_GetAgentExecEngine with an object for the void* pointer return type (Ch interpreter).

Global **LibMC::MCAgent::GetAgentReturnData**(int task_num, IntPtr data, IntPtr dim, IntPtr extent)
Implement GetAgentReturnData

Chapter 7

Bug List

Global **LibMC::MCAgency::SearchForService**(String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr arg) MC_SearchForService is not yet implemented.

Global **LibMC::MCAgency::Steer**(IntPtr funcptr, IntPtr arg) MC_Steer is not yet implemented.

Chapter 8

Namespace Index

8.1 Namespace List

Here is a list of all namespaces with brief descriptions:

EmbeddedCh	??
LibMC	(Namespace for the .NET wrapper for Mobile-C)	??
LibMC::Properties	(Namespace for the .NET wrapper properties class)	??
Program1	??

Chapter 9

Data Structure Index

9.1 Data Structures

Here are the data structures with brief descriptions:

_hr_time	??
_ssl_context	??
_ssl_session	??
_x509_buf	??
_x509_cert	??
_x509_name	??
_x509_node	??
_x509_raw	??
_x509_time	??
aes_context (AES context structure)	??
agency_s (The agency handle)	??
agent_datastate_s	??
agent_mailbox_s	??
agent_s	??
agent_task_s	??
agent_thread_arg_s	??
AP_GENERIC_s	??
arc4_context (ARC4 context structure)	??
barrier_node_s	??
barrier_queue_s	??
EmbeddedCh::ChBlock_t	??
EmbeddedCh::ChInfo_t	??
EmbeddedCh::ChInterp	??
EmbeddedCh::ChMemInfo_t	??
EmbeddedCh::ChOptions_t	??
LibMC::MCAgency::ChOptions_t (ChOptions structures)	??
EmbeddedCh::ChUserDefinedInfo_t	??
EmbeddedCh::ChUserDefinedTag	??
EmbeddedCh::ChVaList	??
cmd_prompt_s	??
command_s	??
connection_s	??
fipa_expression_s::content_u	??

des3_context (Triple-DES context structure)	??
des_context (DES context structure)	??
dhm_context	??
dynstring_s	??
fipa_acl_envelope_Received_s	??
fipa_acl_envelope_s	??
fipa_acl_message_s	??
fipa_acl_Param_s	??
fipa_agent_identifier_s	??
fipa_agent_identifier_set_s	??
fipa_comm_action_t	??
fipa_comm_message_check_t	??
fipa_comm_performative_t	??
fipa_comm_protocol_cn_t	??
fipa_comm_protocol_t	??
fipa_comm_reply_t	??
fipa_comm_t	??
fipa_DateTime_s	??
fipa_expression_s	??
fipa_list_t	??
fipa_message_string_s	??
fipa_number_s	??
fipa_string_s	??
fipa_url_s	??
fipa_url_sequence_s	??
fipa_word_s	??
foo_c	??
foo_s	??
havege_state (HAVEGE state structure)	??
host_id_s	??
hr_time (Timer structure)	??
interpreter_variable_data_s	??
LibMC::InvalidAgencyException (Exception class for use with null agency pointers)	??
LibMC::InvalidAgentException (Exception class for use with null agent pointers)	??
list_s	??
listNode_s	??
mc_platform_s	??
mc_rwlock_s	??
LibMC::MCAclMessage (Encapsulates ACL messages in the Mobile-C library)	??
LibMC::MCAgency (Wrapper class for MCAgency_t structure)	??
LibMC::MCAgency::MCAgency_t	??
MCAgencyOptions_s (User modifiable agency options)	??
LibMC::MCAgency::MCAgencyOptions_t	??
LibMC::MCAgent (Wrapper class for MCAgent_t structure)	??
md2_context (MD2 context structure)	??
md4_context (MD4 context structure)	??
md5_context (MD5 context structure)	??
message_s	??
message_send_arg_s	??
mpi (MPI structure)	??
mtp_http_content_s	??
mtp_http_s	??
mxml_attr_s	??
mxml_custom_s	??

mxml_fdbuf_s	??
mxml_index_s	??
mxml_node_s	??
mxml_text_s	??
mxml_value_s	??
mxml_value_u	??
options	??
Program1::Program	??
rsa_context (RSA context structure)	??
LibMC::Properties::Settings	??
sha1_context (SHA-1 context structure)	??
sha2_context (SHA-256 context structure)	??
sha4_context (SHA-512 context structure)	??
syncList_s	??
syncListNode_s	??

Chapter 10

File Index

10.1 File List

Here is a list of all files with brief descriptions:

/home/dko/Projects/mobilec/trunk/src/acc.c	??
/home/dko/Projects/mobilec/trunk/src/agent.c	??
/home/dko/Projects/mobilec/trunk/src/agent_datastate.c	??
/home/dko/Projects/mobilec/trunk/src/agent_mailbox.c	??
/home/dko/Projects/mobilec/trunk/src/agent_return_data.c	??
/home/dko/Projects/mobilec/trunk/src/agent_task.c	??
/home/dko/Projects/mobilec/trunk/src/ams.c	??
/home/dko/Projects/mobilec/trunk/src/barrier.c	??
/home/dko/Projects/mobilec/trunk/src/cmd_prompt.c	??
/home/dko/Projects/mobilec/trunk/src/connection.c	??
/home/dko/Projects/mobilec/trunk/src/data_structures.c	??
/home/dko/Projects/mobilec/trunk/src/df.c	??
/home/dko/Projects/mobilec/trunk/src/dynstring.c	??
/home/dko/Projects/mobilec/trunk/src/fipa_acl.c	??
/home/dko/Projects/mobilec/trunk/src/fipa_envelope.c	??
/home/dko/Projects/mobilec/trunk/src/libmc.c	??
/home/dko/Projects/mobilec/trunk/src/mc_platform.c	??
/home/dko/Projects/mobilec/trunk/src/mc_rwlock.c	??
/home/dko/Projects/mobilec/trunk/src/message.c	??
/home/dko/Projects/mobilec/trunk/src/message_queue.c	??
/home/dko/Projects/mobilec/trunk/src/mtp_http.c	??
/home/dko/Projects/mobilec/trunk/src/winconfig.h	??
/home/dko/Projects/mobilec/trunk/src/xml_compose.c	??
/home/dko/Projects/mobilec/trunk/src/xml_helper.c	??
/home/dko/Projects/mobilec/trunk/src/xml_parser.c	??
/home/dko/Projects/mobilec/trunk/src/include/acc.h	??
/home/dko/Projects/mobilec/trunk/src/include/agent.h	??
/home/dko/Projects/mobilec/trunk/src/include/agent_datastate.h	??
/home/dko/Projects/mobilec/trunk/src/include/agent_lib.h	??
/home/dko/Projects/mobilec/trunk/src/include/agent_mailbox.h	??
/home/dko/Projects/mobilec/trunk/src/include/agent_task.h	??
/home/dko/Projects/mobilec/trunk/src/include/ams.h	??
/home/dko/Projects/mobilec/trunk/src/include/ap_queue_template.h	??

/home/dko/Projects/mobilec/trunk/src/include/barrier.h	??
/home/dko/Projects/mobilec/trunk/src/include/cmd_prompt.h	??
/home/dko/Projects/mobilec/trunk/src/include/commands.h	??
/home/dko/Projects/mobilec/trunk/src/include/commands.x.h	??
/home/dko/Projects/mobilec/trunk/src/include/connection.h	??
/home/dko/Projects/mobilec/trunk/src/include/data_structures.h	??
/home/dko/Projects/mobilec/trunk/src/include/df.h	??
/home/dko/Projects/mobilec/trunk/src/include/df_request.x.h	??
/home/dko/Projects/mobilec/trunk/src/include/dynstring.h	??
/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h	??
/home/dko/Projects/mobilec/trunk/src/include/fipa_acl_envelope.h	??
/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h	??
/home/dko/Projects/mobilec/trunk/src/include/host_id.h	??
/home/dko/Projects/mobilec/trunk/src/include/interpreter_variable_data.h	??
/home/dko/Projects/mobilec/trunk/src/include/libmc.h (MobileC api header file)	??
/home/dko/Projects/mobilec/trunk/src/include/macros.h	??
/home/dko/Projects/mobilec/trunk/src/include/mc_error.h	??
/home/dko/Projects/mobilec/trunk/src/include/mc_platform.h	??
/home/dko/Projects/mobilec/trunk/src/include/mc_rwlock.h	??
/home/dko/Projects/mobilec/trunk/src/include/message.h	??
/home/dko/Projects/mobilec/trunk/src/include/mobilec.h	??
/home/dko/Projects/mobilec/trunk/src/include/mtp_http.h	??
/home/dko/Projects/mobilec/trunk/src/include/xml_compose.h	??
/home/dko/Projects/mobilec/trunk/src/include/xml_helper.h	??
/home/dko/Projects/mobilec/trunk/src/include/xml_parser.h	??
/home/dko/Projects/mobilec/trunk/src/mc_list/list.c	??
/home/dko/Projects/mobilec/trunk/src/mc_list/list.h	??
/home/dko/Projects/mobilec/trunk/src/mc_sync/sync_list.c	??
/home/dko/Projects/mobilec/trunk/src/mc_sync/sync_list.h	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/config.h	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-attr.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-entity.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-index.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-node.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-private.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-search.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-set.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-string.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml.h	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxmldoc.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/testmxml.c	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/enum.cxx	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/function.cxx	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/struct.cxx	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/vcnet/config.h	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/vcnet2005/config.h	??
/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/vcnet2008/config.h	??
/home/dko/Projects/mobilec/trunk/src/security/asm.c	??
/home/dko/Projects/mobilec/trunk/src/security/asm.h	??
/home/dko/Projects/mobilec/trunk/src/security/asm_message_composer.c	??
/home/dko/Projects/mobilec/trunk/src/security/asm_message_composer.h	??
/home/dko/Projects/mobilec/trunk/src/security/asm_message_parser.c	??

/home/dko/Projects/mobilec/trunk/src/security/asm_message_parser.h	??
/home/dko/Projects/mobilec/trunk/src/security/asm_node.c	??
/home/dko/Projects/mobilec/trunk/src/security/asm_node.h	??
/home/dko/Projects/mobilec/trunk/src/security/interface.c	??
/home/dko/Projects/mobilec/trunk/src/security/interface.h	??
/home/dko/Projects/mobilec/trunk/src/security/mc_dh.c	??
/home/dko/Projects/mobilec/trunk/src/security/mc_dh.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/aes.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/arc4.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/base64.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/bignum.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/bn_mul.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/certs.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/config.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/debug.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/des.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/dhm.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/havege.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md2.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md4.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md5.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/net.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/openssl.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/padlock.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/rsa.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha1.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha2.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha4.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/ssl.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/timing.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/aes.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/arc4.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/base64.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/bignum.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/certs.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/debug.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/des.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/dhm.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/havege.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/md2.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/md4.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/md5.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/net.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/padlock.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/rsa.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/sha1.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/sha2.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/sha4.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/ssl_cli.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/ssl_srv.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/ssl_tls.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/timing.c	??
/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/x509parse.c	??

/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/aes/aescript2.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/hello.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/md5sum.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/sha1sum.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/sha2sum.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/dh_client.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/dh_genprime.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/dh_server.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/mpi_demo.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/rsa_genkey.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/rsa_sign.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/rsa_verify.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/ssl/ssl_client1.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/ssl/ssl_client2.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/ssl/ssl_server.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/benchmark.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/selftest.c ??
 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/ssl_test.c ??
 /home/dko/Projects/mobilec/trunk/src/util/mc_genkey.c ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/Ch.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/ChInterp.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/ChUserDefinedTag.cs ??
 ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/ChVaList.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/EmbedCh.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/Properties/AssemblyInfo.cs ??
 ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/Program1/Program.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/Program1/Properties/AssemblyInfo.cs ??
 ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclMessage.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Settings.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/AssemblyInfo.cs ??
 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/Settings.Designer.cs ??
 ??

Chapter 11

Namespace Documentation

11.1 EmbeddedCh Namespace Reference

Data Structures

- struct [ChInfo_t](#)
- class [ChInterp](#)
- class [ChUserDefinedTag](#)
- class [ChVaList](#)
- struct [ChOptions_t](#)
- struct [ChBlock_t](#)
- struct [ChUserDefinedInfo_t](#)
- struct [ChMemInfo_t](#)

Enumerations

- enum [ChType_t](#) {
 [CH_UNDEFINETYPE](#), [CH_CHARTYPE](#) = 10, [CH_UCHARTYPE](#), [CH_SHORTTYPE](#),
 [CH_USHORTTYPE](#), [CH_INTTYPE](#), [CH_UINTTYPE](#), [CH_LLINTTYPE](#),
 [CH_ULLINTTYPE](#), [CH_FLOATTYPE](#), [CH_DOUBLETTYPE](#), [CH_LDOUBLETTYPE](#),
 [CH_COMPLEXTYPE](#), [CH_LCOMPLEXTYPE](#), [CH_STRINGTYPE](#), [CH_FILETYPE](#),
 [CH_VOIDTYPE](#), [CH_PROCTYPE](#), [CH_STRUCTTYPE](#), [CH_CLASSTYPE](#),
 [CH_UNIONTYPE](#), [CH_ENUMTYPE](#), [CH_CARRAYTYPE](#) = 80, [CH_CARRAYPTRTYPE](#),
 [CH_CARRAYVLATTYPE](#), [CH_CHARRAYTYPE](#), [CH_CHARRAYPTRTYPE](#), [CH_-](#)
 [CHARRAYVLATTYPE](#),
 [CH_NULLTYPE](#) = 100, [CH_VOIDPTRTYPE](#), [CH_CHARPTRTYPE](#), [CH_UCHARPTRTYPE](#),
 [CH_SHORTPTRTYPE](#), [CH_USHORTPTRTYPE](#), [CH_INTPTRTYPE](#), [CH_UINTPTRTYPE](#),
 [CH_LLINTPTRTYPE](#), [CH_ULLINTPTRTYPE](#), [CH_FLOATPTRTYPE](#), [CH_DOUBLEPTRTYPE](#),
 [CH_LDOUBLEPTRTYPE](#), [CH_COMPLEXPTRTYPE](#), [CH_LCOMPLEXPTRTYPE](#), [CH_-](#)
 [STRINGPTRTYPE](#),
 [CH_PROCPTRTYPE](#), [CH_FILEPTRTYPE](#), [CH_STRUCTPTRTYPE](#), [CH_CLASSPTRTYPE](#),
 [CH_UNIONPTRTYPE](#), [CH_ENUMPTRTYPE](#), [CH_VOIDPTR2TYPE](#) = 200, [CH_-](#)
 [CHARPTR2TYPE](#),
}

```

CH_UCHARPTR2TYPE,    CH_SHORTPTR2TYPE,    CH_USHORTPTR2TYPE,    CH_-
INTPTR2TYPE,
CH_UINTPTR2TYPE,    CH_LLINTPTR2TYPE,    CH_ULLINTPTR2TYPE,    CH_-
FLOATPTR2TYPE,
CH_DOUBLEPTR2TYPE,    CH_LDOUBLEPTR2TYPE,    CH_COMPLEXPTR2TYPE,    CH_-
LCOMPLEXPTR2TYPE,
CH_STRINGPTR2TYPE,    CH_FILEPTR2TYPE,    CH_STRUCTPTR2TYPE,    CH_-
CLASSPTR2TYPE,
CH_UNIONPTR2TYPE, CH_ENUMPTR2TYPE }
• enum ChRetVal { CH_OK = 0, CH_ERROR = -1, CH_ABORT = 1 }
• enum ChFuncType_t {
    CH_NOTFUNCTYPE, CH_FUNCTYPE, CH_FUNCPROTOTYPE, CH_FUNCPTRTYPE,
    CH_FUNCMEMBERTYPE, CH_FUNCCONSTYPE, CH_FUNCDESTTYPE }
• enum ChVarType_t { CH_NOTVARTYPE, CH_GLOBALVARTYPE, CH_LOCALVARTYPE }
• enum ChShellType { CH_REGULARCH = 0, CH_SAFECH = 1 }
    Ch shell type.

• enum ChFileDescriptor { STDIN_FILENO = 0, STDOUT_FILENO = 1, STDERR_FILENO = 2 }
• enum ChCallbackMask {
    CH_MASKNONE = 0X0000, CH_MASKCALL = 0X0001, CH_MASKRET = 0X0002, CH_-
    MASKBLOCK = 0X0004,
    CH_MASKEND = 0X0008, CH_MASKLINE = 0X0010, CH_MASKCOUNT = 0X0020, CH_-
    MASKABORT = 0X0040 }

```

11.1.1 Enumeration Type Documentation

11.1.1.1 enum EmbeddedCh::ChCallbackMask

Enumerator:

```

CH_MASKNONE
CH_MASKCALL
CH_MASKRET
CH_MASKBLOCK
CH_MASKEND
CH_MASKLINE
CH_MASKCOUNT
CH_MASKABORT

```

Definition at line 126 of file EmbedCh.cs.

11.1.1.2 enum EmbeddedCh::ChFileDescriptor

Enumerator:

```

STDIN_FILENO
STDOUT_FILENO
STDERR_FILENO

```

Definition at line 117 of file EmbedCh.cs.

11.1.1.3 enum EmbeddedCh::ChFuncType_t

Enumerator:

CH_NOTFUNCTYPE
CH_FUNCTYPE
CH_FUNCPROTOTYPE
CH_FUNCPTRTYPE
CH_FUNCMEMBERTYPE
CH_FUNCCONSTTYPE
CH_FUNCDESTTYPE

Definition at line 24 of file EmbedCh.cs.

11.1.1.4 enum EmbeddedCh::ChRetVal

Enumerator:

CH_OK
CH_ERROR
CH_ABORT

Definition at line 106 of file Ch.cs.

11.1.1.5 enum EmbeddedCh::ChShellType

Ch shell type. Used to set the shell type for the Ch interpreter.

Enumerator:

CH_REGULARCH Default, regular shell
CH_SAFECH Safe shell

Definition at line 110 of file EmbedCh.cs.

11.1.1.6 enum EmbeddedCh::ChType_t

Enumerator:

CH_UNDEFINETYPE
CH_CHARTYPE
CH_UCHARTYPE
CH_SHORTTYPE
CH_USHORTTYPE
CH_INTTYPE
CH_UINTTYPE
CH_LLINTTYPE
CH_ULLINTTYPE

CH_FLOATTYPE
CH_DOUBLETTYPE
CH_LDOUBLETTYPE
CH_COMPLEXTYPE
CH_LCOMPLEXTYPE
CH_STRINGTYPE
CH_FILETYPE
CH_VOIDTYPE
CH_PROCTYPE
CH_STRUCTTYPE
CH_CLASSTYPE
CH_UNIONTYPE
CH_ENUMTYPE
CH_CARRAYTYPE
CH_CARRAYPTRTYPE
CH_CARRAYVLATTYPE
CH_CHARRAYTYPE
CH_CHARRAYPTRTYPE
CH_CHARRAYVLATTYPE
CH_NULLTYPE
CH_VOIDPTRTYPE
CH_CHARPTRTYPE
CH_UCHARPTRTYPE
CH_SHORTPTRTYPE
CH_USHORTPTRTYPE
CH_INTPTRTYPE
CH_UINTPTRTYPE
CH_LLINTPTRTYPE
CH_ULLINTPTRTYPE
CH_FLOATPTRTYPE
CH_DOUBLEPTRTYPE
CH_LDOUBLEPTRTYPE
CH_COMPLEXPTRTYPE
CH_LCOMPLEXPTRTYPE
CH_STRINGPTRTYPE
CH_PROCPTRTYPE
CH_FILEPTRTYPE
CH_STRUCTPTRTYPE
CH_CLASSPTRTYPE
CH_UNIONPTRTYPE
CH_ENUMPTRTYPE

CH_VOIDPTR2TYPE
CH_CHARPTR2TYPE
CH_UCHARPTR2TYPE
CH_SHORTPTR2TYPE
CH_USHORTPTR2TYPE
CH_INTPTR2TYPE
CH_UINTPTR2TYPE
CH_LLINTPTR2TYPE
CH_ULLINTPTR2TYPE
CH_FLOATPTR2TYPE
CH_DOUBLEPTR2TYPE
CH_LDOUBLEPTR2TYPE
CH_COMPLEXPTR2TYPE
CH_LCOMPLEXPTR2TYPE
CH_STRINGPTR2TYPE
CH_FILEPTR2TYPE
CH_STRUCTPTR2TYPE
CH_CLASSPTR2TYPE
CH_UNIONPTR2TYPE
CH_ENUMPTR2TYPE

Definition at line 19 of file Ch.cs.

11.1.1.7 enum EmbeddedCh::ChVarType_t

Enumerator:

CH_NOTVARTYPE
CH_GLOBALVARTYPE
CH_LOCALVARTYPE

Definition at line 36 of file EmbedCh.cs.

11.2 LibMC Namespace Reference

Namespace for the .NET wrapper for Mobile-C.

Namespaces

- namespace [Properties](#)
Namespace for the .NET wrapper properties class.

Data Structures

- class [MCAclMessage](#)
Encapsulates ACL messages in the Mobile-C library.
- class [MCAgency](#)
Wrapper class for [MCAgency_t](#) structure.
- class [InvalidAgencyException](#)
Exception class for use with null agency pointers.
- class [MCAgent](#)
Wrapper class for [MCAgent_t](#) structure.
- class [InvalidAgentException](#)
Exception class for use with null agent pointers.

11.2.1 Detailed Description

Namespace for the .NET wrapper for Mobile-C. [LibMC](#) encapsulates the Mobile-C DLL for windows in an .NET class library. .NET programs can access the library to create agencies, connect to agencies, interact with agents, etc.

11.3 LibMC::Properties Namespace Reference

Namespace for the .NET wrapper properties class.

Data Structures

- class [Settings](#)

11.3.1 Detailed Description

Namespace for the .NET wrapper properties class. Any user or global properties that should be preserved from session to session can be added here through the designer. There are currently no properties in use.

11.4 Program1 Namespace Reference

Data Structures

- class [Program](#)

Chapter 12

Data Structure Documentation

12.1 `_hr_time` Struct Reference

Data Fields

- struct timeval [start](#)

12.1.1 Detailed Description

Definition at line 45 of file `timing.c`.

12.1.2 Field Documentation

12.1.2.1 `struct timeval _hr_time::start` [`read`]

Definition at line 47 of file `timing.c`.

Referenced by `get_timer()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/timing.c`

12.2 `_ssl_context` Struct Reference

```
#include <ssl.h>
```

Data Fields

- `int` `state`
- `int` `major_ver`
- `int` `minor_ver`
- `int` `max_major_ver`
- `int` `max_minor_ver`
- `int(* f_rng)(void *)`
- `void(* f_dbg)(void *, int, char *)`
- `int(* f_recv)(void *, unsigned char *, int)`
- `int(* f_send)(void *, unsigned char *, int)`
- `void *` `p_rng`
- `void *` `p_dbg`
- `void *` `p_recv`
- `void *` `p_send`
- `int` `resume`
- `int` `timeout`
- `ssl_session *` `session`
- `int(* s_get)(ssl_context *)`
- `int(* s_set)(ssl_context *)`
- `unsigned char *` `in_ctr`
- `unsigned char *` `in_hdr`
- `unsigned char *` `in_msg`
- `unsigned char *` `in_offt`
- `int` `in_msgtype`
- `int` `in_msglen`
- `int` `in_left`
- `int` `in_hhlen`
- `int` `nb_zero`
- `unsigned char *` `out_ctr`
- `unsigned char *` `out_hdr`
- `unsigned char *` `out_msg`
- `int` `out_msgtype`
- `int` `out_msglen`
- `int` `out_left`
- `rsa_context *` `rsa_key`
- `x509_cert *` `own_cert`
- `x509_cert *` `ca_chain`
- `x509_cert *` `peer_cert`
- `char *` `peer_cn`
- `int` `endpoint`
- `int` `authmode`
- `int` `client_auth`
- `int` `verify_result`
- `dhm_context` `dhm_ctx`

- `md5_context` `fin_md5`
- `sha1_context` `fin_sha1`
- `int` `do_crypt`
- `int *` `ciphers`
- `int` `pmslen`
- `int` `keylen`
- `int` `minlen`
- `int` `ivlen`
- `int` `maclen`
- `unsigned char` `randbytes` [64]
- `unsigned char` `premaster` [256]
- `unsigned char` `iv_enc` [16]
- `unsigned char` `iv_dec` [16]
- `unsigned char` `mac_enc` [32]
- `unsigned char` `mac_dec` [32]
- `unsigned long` `ctx_enc` [128]
- `unsigned long` `ctx_dec` [128]
- `unsigned char *` `hostname`
- `unsigned long` `hostname_len`

12.2.1 Detailed Description

Definition at line 149 of file `ssl.h`.

12.2.2 Field Documentation

12.2.2.1 `int _ssl_context::authmode`

verification mode

Definition at line 220 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, `ssl_set_authmode()`, and `ssl_write_certificate_request()`.

12.2.2.2 `x509_cert* _ssl_context::ca_chain`

own trusted CA chain

Definition at line 215 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, `ssl_set_ca_chain()`, and `ssl_write_certificate_request()`.

12.2.2.3 `int* _ssl_context::ciphers`

allowed ciphersuites

Definition at line 232 of file `ssl.h`.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_set_ciphers()`, and `ssl_write_client_hello()`.

12.2.2.4 int _ssl_context::client_auth

flag for client auth.

Definition at line 221 of file ssl.h.

Referenced by ssl_parse_certificate_request(), ssl_parse_server_hello_done(), ssl_write_certificate(), and ssl_write_certificate_verify().

12.2.2.5 unsigned long _ssl_context::ctx_dec[128]

decryption context

Definition at line 249 of file ssl.h.

Referenced by ssl_decrypt_buf(), and ssl_derive_keys().

12.2.2.6 unsigned long _ssl_context::ctx_enc[128]

encryption context

Definition at line 248 of file ssl.h.

Referenced by ssl_derive_keys(), and ssl_encrypt_buf().

12.2.2.7 dhm_context _ssl_context::dhm_ctx

DHM key exchange

Definition at line 227 of file ssl.h.

Referenced by ssl_free(), ssl_parse_client_key_exchange(), ssl_parse_server_key_exchange(), ssl_set_dh_param(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

12.2.2.8 int _ssl_context::do_crypt

en(de)cryption flag

Definition at line 231 of file ssl.h.

Referenced by ssl_parse_change_cipher_spec(), ssl_parse_finished(), ssl_read_record(), ssl_write_change_cipher_spec(), ssl_write_finished(), and ssl_write_record().

12.2.2.9 int _ssl_context::endpoint

0: client, 1: server

Definition at line 219 of file ssl.h.

Referenced by ssl_derive_keys(), ssl_handshake(), ssl_parse_certificate(), ssl_parse_finished(), ssl_set_endpoint(), ssl_write_certificate(), and ssl_write_finished().

12.2.2.10 void(* _ssl_context::f_dbg)(void *,int, char *)

Referenced by debug_print_buf(), debug_print_crt(), debug_print_mpi(), debug_print_msg(), debug_print_ret(), and ssl_set_dbg().

12.2.2.11 `int(* _ssl_context::f_recv)(void *, unsigned char *, int)`

Referenced by `ssl_fetch_input()`, and `ssl_set_bio()`.

12.2.2.12 `int(* _ssl_context::f_rng)(void *)`

Referenced by `ssl_parse_client_key_exchange()`, `ssl_set_rng()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

12.2.2.13 `int(* _ssl_context::f_send)(void *, unsigned char *, int)`

Referenced by `ssl_flush_output()`, and `ssl_set_bio()`.

12.2.2.14 `md5_context _ssl_context::fin_md5`

Finished MD5 checksum

Definition at line 228 of file `ssl.h`.

Referenced by `ssl_calc_verify()`, `ssl_init()`, `ssl_parse_client_hello()`, `ssl_parse_finished()`, `ssl_read_record()`, `ssl_write_finished()`, and `ssl_write_record()`.

12.2.2.15 `sha1_context _ssl_context::fin_sha1`

Finished SHA-1 checksum

Definition at line 229 of file `ssl.h`.

Referenced by `ssl_calc_verify()`, `ssl_init()`, `ssl_parse_client_hello()`, `ssl_parse_finished()`, `ssl_read_record()`, `ssl_write_finished()`, and `ssl_write_record()`.

12.2.2.16 `unsigned char* _ssl_context::hostname`

Definition at line 254 of file `ssl.h`.

Referenced by `ssl_free()`, `ssl_init()`, `ssl_set_hostname()`, and `ssl_write_client_hello()`.

12.2.2.17 `unsigned long _ssl_context::hostname_len`

Definition at line 255 of file `ssl.h`.

Referenced by `ssl_free()`, `ssl_init()`, `ssl_set_hostname()`, and `ssl_write_client_hello()`.

12.2.2.18 `unsigned char* _ssl_context::in_ctr`

64-bit incoming message counter

Definition at line 187 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, `ssl_free()`, and `ssl_init()`.

12.2.2.19 unsigned char* _ssl_context::in_hdr

5-byte record header (in_ctr+8)

Definition at line 188 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_fetch_input(), ssl_init(), ssl_parse_client_hello(), and ssl_read_record().

12.2.2.20 int _ssl_context::in_hslen

current handshake message length

Definition at line 196 of file ssl.h.

Referenced by ssl_parse_certificate(), ssl_parse_certificate_verify(), ssl_parse_client_key_exchange(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), and ssl_read_record().

12.2.2.21 int _ssl_context::in_left

amount of data read so far

Definition at line 194 of file ssl.h.

Referenced by ssl_fetch_input(), ssl_parse_client_hello(), and ssl_read_record().

12.2.2.22 unsigned char* _ssl_context::in_msg

the message contents (in_hdr+5)

Definition at line 189 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_init(), ssl_parse_certificate(), ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_change_cipher_spec(), ssl_parse_client_hello(), ssl_parse_client_key_exchange(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), ssl_read(), and ssl_read_record().

12.2.2.23 int _ssl_context::in_msglen

record header: message length

Definition at line 193 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_get_bytes_avail(), ssl_parse_certificate(), ssl_parse_change_cipher_spec(), ssl_read(), and ssl_read_record().

12.2.2.24 int _ssl_context::in_msgtype

record header: message type

Definition at line 192 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_parse_certificate(), ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_change_cipher_spec(), ssl_parse_client_key_exchange(), ssl_parse_

`finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read()`, and `ssl_read_record()`.

12.2.2.25 `unsigned char* _ssl_context::in_offt`

read offset in application data

Definition at line 190 of file `ssl.h`.

Referenced by `ssl_get_bytes_avail()`, and `ssl_read()`.

12.2.2.26 `unsigned char _ssl_context::iv_dec[16]`

IV (decryption)

Definition at line 243 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, and `ssl_derive_keys()`.

12.2.2.27 `unsigned char _ssl_context::iv_enc[16]`

IV (encryption)

Definition at line 242 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

12.2.2.28 `int _ssl_context::ivlen`

IV length

Definition at line 236 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

12.2.2.29 `int _ssl_context::keylen`

symmetric key length

Definition at line 234 of file `ssl.h`.

Referenced by `ssl_derive_keys()`.

12.2.2.30 `unsigned char _ssl_context::mac_dec[32]`

MAC (decryption)

Definition at line 246 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`, and `ssl_derive_keys()`.

12.2.2.31 `unsigned char _ssl_context::mac_enc[32]`

MAC (encryption)

Definition at line 245 of file ssl.h.

Referenced by `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

12.2.2.32 `int _ssl_context::maclen`

MAC length

Definition at line 237 of file ssl.h.

Referenced by `ssl_decrypt_buf()`, `ssl_derive_keys()`, and `ssl_encrypt_buf()`.

12.2.2.33 `int _ssl_context::major_ver`

equal to `SSL_MAJOR_VERSION_3`

Definition at line 156 of file ssl.h.

Referenced by `ssl_parse_client_hello()`, `ssl_read_record()`, `ssl_write_client_hello()`, `ssl_write_record()`, and `ssl_write_server_hello()`.

12.2.2.34 `int _ssl_context::max_major_ver`

max. major version from client

Definition at line 159 of file ssl.h.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_write_client_hello()`, and `ssl_write_client_key_exchange()`.

12.2.2.35 `int _ssl_context::max_minor_ver`

max. minor version from client

Definition at line 160 of file ssl.h.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_write_client_hello()`, and `ssl_write_client_key_exchange()`.

12.2.2.36 `int _ssl_context::minlen`

min. ciphertext length

Definition at line 235 of file ssl.h.

Referenced by `ssl_decrypt_buf()`, `ssl_derive_keys()`, and `ssl_read_record()`.

12.2.2.37 `int _ssl_context::minor_ver`

either 0 (SSL3) or 1 (TLS1.0)

Definition at line 157 of file ssl.h.

Referenced by `ssl_calc_finished()`, `ssl_calc_verify()`, `ssl_decrypt_buf()`, `ssl_derive_keys()`, `ssl_encrypt_buf()`, `ssl_parse_certificate()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_`

`finished()`, `ssl_parse_server_hello()`, `ssl_read_record()`, `ssl_write_certificate()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, and `ssl_write_server_hello()`.

12.2.2.38 `int _ssl_context::nb_zero`

of 0-length encrypted messages

Definition at line 197 of file `ssl.h`.

Referenced by `ssl_decrypt_buf()`.

12.2.2.39 `unsigned char* _ssl_context::out_ctr`

64-bit outgoing message counter

Definition at line 202 of file `ssl.h`.

Referenced by `ssl_encrypt_buf()`, `ssl_free()`, and `ssl_init()`.

12.2.2.40 `unsigned char* _ssl_context::out_hdr`

5-byte record header (`out_ctr+8`)

Definition at line 203 of file `ssl.h`.

Referenced by `ssl_flush_output()`, `ssl_init()`, and `ssl_write_record()`.

12.2.2.41 `int _ssl_context::out_left`

amount of data not yet written

Definition at line 208 of file `ssl.h`.

Referenced by `ssl_flush_output()`, `ssl_write()`, and `ssl_write_record()`.

12.2.2.42 `unsigned char* _ssl_context::out_msg`

the message contents (`out_hdr+5`)

Definition at line 204 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_encrypt_buf()`, `ssl_init()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

12.2.2.43 `int _ssl_context::out_msglen`

record header: message length

Definition at line 207 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_encrypt_buf()`, `ssl_flush_output()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

12.2.2.44 `int _ssl_context::out_msgtype`

record header: message type

Definition at line 206 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_encrypt_buf()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

12.2.2.45 `x509_cert* _ssl_context::own_cert`

own X.509 certificate

Definition at line 214 of file `ssl.h`.

Referenced by `ssl_set_own_cert()`, and `ssl_write_certificate()`.

12.2.2.46 `void* _ssl_context::p_dbg`

context for the debug function

Definition at line 171 of file `ssl.h`.

Referenced by `debug_print_buf()`, `debug_print_crt()`, `debug_print_mpi()`, `debug_print_msg()`, `debug_print_ret()`, and `ssl_set_dbg()`.

12.2.2.47 `void* _ssl_context::p_recv`

context for reading operations

Definition at line 172 of file `ssl.h`.

Referenced by `ssl_fetch_input()`, and `ssl_set_bio()`.

12.2.2.48 `void* _ssl_context::p_rng`

context for the RNG function

Definition at line 170 of file `ssl.h`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_set_rng()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

12.2.2.49 `void* _ssl_context::p_send`

context for writing operations

Definition at line 173 of file `ssl.h`.

Referenced by `ssl_flush_output()`, and `ssl_set_bio()`.

12.2.2.50 `x509_cert* _ssl_context::peer_cert`

peer X.509 cert chain

Definition at line 216 of file `ssl.h`.

Referenced by `main()`, `ssl_free()`, `ssl_parse_certificate()`, `ssl_parse_certificate_verify()`, `ssl_parse_server_key_exchange()`, and `ssl_write_client_key_exchange()`.

12.2.2.51 `char* _ssl_context::peer_cn`

expected peer CN

Definition at line 217 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, and `ssl_set_ca_chain()`.

12.2.2.52 `int _ssl_context::pmslen`

premaster length

Definition at line 233 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_parse_client_key_exchange()`, and `ssl_write_client_key_exchange()`.

12.2.2.53 `unsigned char _ssl_context::premaster[256]`

premaster secret

Definition at line 240 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_parse_client_key_exchange()`, and `ssl_write_client_key_exchange()`.

12.2.2.54 `unsigned char _ssl_context::randbytes[64]`

random bytes

Definition at line 239 of file `ssl.h`.

Referenced by `ssl_derive_keys()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_hello()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

12.2.2.55 `int _ssl_context::resume`

session resuming flag

Definition at line 178 of file `ssl.h`.

Referenced by `my_get_session()`, `ssl_derive_keys()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_set_session()`, `ssl_write_client_hello()`, `ssl_write_finished()`, and `ssl_write_server_hello()`.

12.2.2.56 `rsa_context* _ssl_context::rsa_key`

own RSA private key

Definition at line 213 of file `ssl.h`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_set_own_cert()`, `ssl_write_certificate_verify()`, and `ssl_write_server_key_exchange()`.

12.2.2.57 int(* _ssl_context::s_get)(ssl_context *)

(server) get callback

Referenced by `ssl_set_scb()`, and `ssl_write_server_hello()`.

12.2.2.58 int(* _ssl_context::s_set)(ssl_context *)

(server) set callback

Referenced by `ssl_parse_client_key_exchange()`, and `ssl_set_scb()`.

12.2.2.59 ssl_session* _ssl_context::session

current session data

Definition at line 180 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_calc_finished()`, `ssl_calc_verify()`, `ssl_derive_keys()`, `ssl_get_cipher()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_parse_server_key_exchange()`, `ssl_set_session()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

12.2.2.60 int _ssl_context::state

SSL handshake: current state

Definition at line 154 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_handshake_client()`, `ssl_handshake_server()`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

12.2.2.61 int _ssl_context::timeout

sess. expiration time

Definition at line 179 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_set_session()`, and `ssl_write_client_hello()`.

12.2.2.62 int _ssl_context::verify_result

verification result

Definition at line 222 of file `ssl.h`.

Referenced by `ssl_get_verify_result()`, and `ssl_parse_certificate()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/ssl.h`

12.3 _ssl_session Struct Reference

```
#include <ssl.h>
```

Data Fields

- [time_t start](#)
- [int cipher](#)
- [int length](#)
- unsigned char [id](#) [32]
- unsigned char [master](#) [48]
- [ssl_session](#) * [next](#)

12.3.1 Detailed Description

Definition at line 139 of file `ssl.h`.

12.3.2 Field Documentation

12.3.2.1 `int _ssl_session::cipher`

chosen cipher

Definition at line 142 of file `ssl.h`.

Referenced by `my_get_session()`, `ssl_derive_keys()`, `ssl_get_cipher()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_key_exchange()`, `ssl_write_server_hello()`, and `ssl_write_server_key_exchange()`.

12.3.2.2 `unsigned char _ssl_session::id[32]`

session identifier

Definition at line 144 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_write_client_hello()`, and `ssl_write_server_hello()`.

12.3.2.3 `int _ssl_session::length`

session id length

Definition at line 143 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_write_client_hello()`, and `ssl_write_server_hello()`.

12.3.2.4 `unsigned char _ssl_session::master[48]`

the master secret

Definition at line 145 of file `ssl.h`.

Referenced by `my_get_session()`, `ssl_calc_finished()`, `ssl_calc_verify()`, and `ssl_derive_keys()`.

12.3.2.5 `ssl_session* _ssl_session::next`

next session entry

Definition at line 146 of file `ssl.h`.

Referenced by `main()`, `my_get_session()`, and `my_set_session()`.

12.3.2.6 `time_t _ssl_session::start`

starting time

Definition at line 141 of file `ssl.h`.

Referenced by `my_get_session()`, `my_set_session()`, `ssl_parse_server_hello()`, and `ssl_write_client_hello()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/ssl.h`

12.4 `_x509_buf` Struct Reference

```
#include <x509.h>
```

Data Fields

- `int tag`
- `int len`
- `unsigned char * p`

12.4.1 Detailed Description

Definition at line 96 of file `x509.h`.

12.4.2 Field Documentation

12.4.2.1 `int _x509_buf::len`

Definition at line 99 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_free()`, `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, `x509parse_cert_info()`, `x509parse_crt()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

12.4.2.2 `unsigned char* _x509_buf::p`

Definition at line 100 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_free()`, `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, `x509parse_cert_info()`, `x509parse_crt()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

12.4.2.3 `int _x509_buf::tag`

Definition at line 98 of file `x509.h`.

Referenced by `x509_get_alg()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_serial()`, `x509_get_sig()`, and `x509_get_uid()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h`

12.5 `_x509_cert` Struct Reference

```
#include <x509.h>
```

Data Fields

- `x509_buf raw`
- `x509_buf tbs`
- `int version`
- `x509_buf serial`
- `x509_buf sig_oid1`
- `x509_buf issuer_raw`
- `x509_buf subject_raw`
- `x509_name issuer`
- `x509_name subject`
- `x509_time valid_from`
- `x509_time valid_to`
- `x509_buf pk_oid`
- `rsa_context rsa`
- `x509_buf issuer_id`
- `x509_buf subject_id`
- `x509_buf v3_ext`
- `int ca_istrue`
- `int max_pathlen`
- `x509_buf sig_oid2`
- `x509_buf sig`
- `struct _x509_cert * next`

12.5.1 Detailed Description

Definition at line 119 of file `x509.h`.

12.5.2 Field Documentation

12.5.2.1 `int _x509_cert::ca_istrue`

Definition at line 144 of file `x509.h`.

Referenced by `x509parse_cert()`, and `x509parse_verify()`.

12.5.2.2 `x509_name _x509_cert::issuer`

Definition at line 131 of file `x509.h`.

Referenced by `x509_free()`, `x509parse_cert_info()`, and `x509parse_cert()`.

12.5.2.3 `x509_buf _x509_cert::issuer_id`

Definition at line 140 of file `x509.h`.

Referenced by `x509parse_cert()`.

12.5.2.4 `x509_buf_x509_cert::issuer_raw`

Definition at line 128 of file `x509.h`.

Referenced by `x509parse_cert()`, and `x509parse_verify()`.

12.5.2.5 `int_x509_cert::max_pathlen`

Definition at line 145 of file `x509.h`.

Referenced by `x509parse_cert()`, and `x509parse_verify()`.

12.5.2.6 `struct_x509_cert*_x509_cert::next` [`read`]

Definition at line 150 of file `x509.h`.

Referenced by `debug_print_cert()`, `main()`, `ssl_test()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `x509_free()`, `x509parse_cert()`, and `x509parse_verify()`.

12.5.2.7 `x509_buf_x509_cert::pk_oid`

Definition at line 137 of file `x509.h`.

Referenced by `x509parse_cert()`.

12.5.2.8 `x509_buf_x509_cert::raw`

Definition at line 121 of file `x509.h`.

Referenced by `ssl_write_certificate()`, `x509_free()`, and `x509parse_cert()`.

12.5.2.9 `rsa_context_x509_cert::rsa`

Definition at line 138 of file `x509.h`.

Referenced by `debug_print_cert()`, `ssl_parse_certificate_verify()`, `ssl_parse_server_key_exchange()`, `ssl_write_client_key_exchange()`, `x509_free()`, `x509parse_cert_info()`, `x509parse_cert()`, and `x509parse_verify()`.

12.5.2.10 `x509_buf_x509_cert::serial`

Definition at line 125 of file `x509.h`.

Referenced by `x509parse_cert_info()`, and `x509parse_cert()`.

12.5.2.11 `x509_buf_x509_cert::sig`

Definition at line 148 of file `x509.h`.

Referenced by `x509parse_cert()`, and `x509parse_verify()`.

12.5.2.12 x509_buf_x509_cert::sig_oid1

Definition at line 126 of file x509.h.

Referenced by x509parse_cert_info(), x509parse_cert(), and x509parse_verify().

12.5.2.13 x509_buf_x509_cert::sig_oid2

Definition at line 147 of file x509.h.

Referenced by x509parse_cert().

12.5.2.14 x509_name_x509_cert::subject

Definition at line 132 of file x509.h.

Referenced by x509_free(), x509parse_cert_info(), x509parse_cert(), and x509parse_verify().

12.5.2.15 x509_buf_x509_cert::subject_id

Definition at line 141 of file x509.h.

Referenced by x509parse_cert().

12.5.2.16 x509_buf_x509_cert::subject_raw

Definition at line 129 of file x509.h.

Referenced by ssl_write_certificate_request(), x509parse_cert(), and x509parse_verify().

12.5.2.17 x509_buf_x509_cert::tbs

Definition at line 122 of file x509.h.

Referenced by x509parse_cert(), and x509parse_verify().

12.5.2.18 x509_buf_x509_cert::v3_ext

Definition at line 142 of file x509.h.

Referenced by x509parse_cert().

12.5.2.19 x509_time_x509_cert::valid_from

Definition at line 134 of file x509.h.

Referenced by x509parse_cert_info(), and x509parse_cert().

12.5.2.20 x509_time_x509_cert::valid_to

Definition at line 135 of file x509.h.

Referenced by x509parse_cert_info(), x509parse_cert(), and x509parse_expired().

12.5.2.21 `int _x509_cert::version`

Definition at line 124 of file `x509.h`.

Referenced by `x509parse_cert_info()`, `x509parse_cert()`, and `x509parse_verify()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h`

12.6 `_x509_name` Struct Reference

```
#include <x509.h>
```

Data Fields

- [x509_buf oid](#)
- [x509_buf val](#)
- [struct `_x509_name` * next](#)

12.6.1 Detailed Description

Definition at line 104 of file `x509.h`.

12.6.2 Field Documentation

12.6.2.1 `struct _x509_name* _x509_name::next` **[read]**

Definition at line 108 of file `x509.h`.

Referenced by `x509_free()`, `x509_get_name()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

12.6.2.2 `x509_buf _x509_name::oid`

Definition at line 106 of file `x509.h`.

Referenced by `x509_get_name()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

12.6.2.3 `x509_buf _x509_name::val`

Definition at line 107 of file `x509.h`.

Referenced by `x509_get_name()`, `x509parse_dn_gets()`, and `x509parse_verify()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h`

12.7 `_x509_node` Struct Reference

```
#include <x509.h>
```

Data Fields

- unsigned char * [data](#)
- unsigned char * [p](#)
- unsigned char * [end](#)
- size_t [len](#)

12.7.1 Detailed Description

Definition at line 157 of file `x509.h`.

12.7.2 Field Documentation

12.7.2.1 unsigned char* `_x509_node::data`

Definition at line 159 of file `x509.h`.

12.7.2.2 unsigned char* `_x509_node::end`

Definition at line 161 of file `x509.h`.

12.7.2.3 size_t `_x509_node::len`

Definition at line 163 of file `x509.h`.

12.7.2.4 unsigned char* `_x509_node::p`

Definition at line 160 of file `x509.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h`

12.8 `_x509_raw` Struct Reference

```
#include <x509.h>
```

Data Fields

- [x509_node raw](#)
- [x509_node tbs](#)
- [x509_node version](#)
- [x509_node serial](#)
- [x509_node tbs_signalg](#)
- [x509_node issuer](#)
- [x509_node validity](#)
- [x509_node subject](#)
- [x509_node subpubkey](#)
- [x509_node signalg](#)
- [x509_node sign](#)

12.8.1 Detailed Description

Definition at line 167 of file x509.h.

12.8.2 Field Documentation

12.8.2.1 `x509_node _x509_raw::issuer`

Definition at line 175 of file x509.h.

12.8.2.2 `x509_node _x509_raw::raw`

Definition at line 169 of file x509.h.

12.8.2.3 `x509_node _x509_raw::serial`

Definition at line 173 of file x509.h.

12.8.2.4 `x509_node _x509_raw::sign`

Definition at line 181 of file x509.h.

12.8.2.5 `x509_node _x509_raw::signalg`

Definition at line 180 of file x509.h.

12.8.2.6 `x509_node _x509_raw::subject`

Definition at line 177 of file x509.h.

12.8.2.7 `x509_node_x509_raw::subpubkey`

Definition at line 178 of file `x509.h`.

12.8.2.8 `x509_node_x509_raw::tbs`

Definition at line 170 of file `x509.h`.

12.8.2.9 `x509_node_x509_raw::tbs_signalg`

Definition at line 174 of file `x509.h`.

12.8.2.10 `x509_node_x509_raw::validity`

Definition at line 176 of file `x509.h`.

12.8.2.11 `x509_node_x509_raw::version`

Definition at line 172 of file `x509.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h`

12.9 `_x509_time` Struct Reference

```
#include <x509.h>
```

Data Fields

- `int year`
- `int mon`
- `int day`
- `int hour`
- `int min`
- `int sec`

12.9.1 Detailed Description

Definition at line 112 of file `x509.h`.

12.9.2 Field Documentation

12.9.2.1 `int _x509_time::day`

Definition at line 114 of file `x509.h`.

Referenced by `x509_get_dates()`, `x509parse_cert_info()`, and `x509parse_expired()`.

12.9.2.2 `int _x509_time::hour`

Definition at line 115 of file `x509.h`.

Referenced by `x509_get_dates()`, and `x509parse_cert_info()`.

12.9.2.3 `int _x509_time::min`

Definition at line 115 of file `x509.h`.

Referenced by `x509_get_dates()`, and `x509parse_cert_info()`.

12.9.2.4 `int _x509_time::mon`

Definition at line 114 of file `x509.h`.

Referenced by `x509_get_dates()`, `x509parse_cert_info()`, and `x509parse_expired()`.

12.9.2.5 `int _x509_time::sec`

Definition at line 115 of file `x509.h`.

Referenced by `x509_get_dates()`, and `x509parse_cert_info()`.

12.9.2.6 `int _x509_time::year`

Definition at line 114 of file `x509.h`.

Referenced by `x509_get_dates()`, `x509parse_cert_info()`, and `x509parse_expired()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h`

12.10 aes_context Struct Reference

AES context structure.

```
#include <aes.h>
```

Data Fields

- `int nr`
- `unsigned long * rk`
- `unsigned long buf[68]`

12.10.1 Detailed Description

AES context structure.

Definition at line 13 of file `aes.h`.

12.10.2 Field Documentation

12.10.2.1 `unsigned long aes_context::buf[68]`

unaligned data

Definition at line 17 of file `aes.h`.

Referenced by `aes_setkey_dec()`, and `aes_setkey_enc()`.

12.10.2.2 `int aes_context::nr`

number of rounds

Definition at line 15 of file `aes.h`.

Referenced by `aes_crypt_ecb()`, `aes_setkey_dec()`, and `aes_setkey_enc()`.

12.10.2.3 `unsigned long* aes_context::rk`

AES round keys

Definition at line 16 of file `aes.h`.

Referenced by `aes_crypt_ecb()`, `aes_setkey_dec()`, and `aes_setkey_enc()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/aes.h`

12.11 agency_s Struct Reference

The agency handle.

```
#include <libmc.h>
```

Data Fields

- [int client](#)
- [int server](#)
- [char * hostName](#)
- [char * filename](#)
- [int portno](#)
- [int portnoc](#)
- [int initInterps](#)
- [struct mc_platform_s * mc_platform](#)
- [int default_agentstatus](#)
- [int threads](#)
- [int enable_security](#)
- [int stack_size](#) [MC_THREAD_ALL]
- [char * priv_key_filename](#)
- [char * known_host_filename](#)
- [error_code_t last_error](#)

12.11.1 Detailed Description

The agency handle.

Definition at line 224 of file libmc.h.

12.11.2 Field Documentation

12.11.2.1 int agency_s::client

Definition at line 225 of file libmc.h.

Referenced by MC_Initialize().

12.11.2.2 int agency_s::default_agentstatus

Agency default agent status

Definition at line 233 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

12.11.2.3 int agency_s::enable_security

Security flag

Definition at line 235 of file libmc.h.

12.11.2.4 char* agency_s::filename

Definition at line 228 of file libmc.h.

12.11.2.5 char* agency_s::hostName

Local Hostname

Definition at line 227 of file libmc.h.

Referenced by handler_COMPOSE_SEND(), MC_End(), and MC_Initialize().

12.11.2.6 int agency_s::initInterps

Definition at line 231 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

12.11.2.7 char* agency_s::known_host_filename

Definition at line 239 of file libmc.h.

Referenced by listen_Thread(), MC_Initialize(), and message_send_Thread().

12.11.2.8 error_code_t agency_s::last_error

Definition at line 240 of file libmc.h.

Referenced by mc_platform_Initialize().

12.11.2.9 struct mc_platform_s* agency_s::mc_platform [read]

Local MobileC Platform

Definition at line 232 of file libmc.h.

Referenced by MC_AclSend(), MC_AclSend_chdl(), MC_AddAgent(), MC_AddAgent_chdl(), MC_-AddStationaryAgent(), MC_Barrier(), MC_Barrier_chdl(), MC_BarrierDelete(), MC_BarrierDelete_chdl(), MC_BarrierInit(), MC_BarrierInit_chdl(), MC_CondBroadcast(), MC_CondBroadcast_chdl(), MC_CondReset(), MC_CondReset_chdl(), MC_CondSignal(), MC_CondSignal_chdl(), MC_CondWait(), MC_CondWait_chdl(), MC_DeregisterService(), MC_DeregisterService_chdl(), MC_End(), MC_End_chdl(), MC_FindAgentByID(), MC_FindAgentByID_chdl(), MC_FindAgentByName(), MC_FindAgentByName_chdl(), MC_GetAllAgents(), MC_HaltAgency(), MC_HaltAgency_chdl(), MC_Initialize(), MC_LoadAgentFromFile(), MC_MainLoop(), MC_MutexLock(), MC_MutexLock_chdl(), MC_MutexUnlock(), MC_MutexUnlock_chdl(), MC_RegisterService(), MC_RegisterService_chdl(), MC_ResetSignal(), MC_ResumeAgency(), MC_ResumeAgency_chdl(), MC_RetrieveAgent(), MC_RetrieveAgent_chdl(), MC_SearchForService(), MC_SearchForService_chdl(), MC_SemaphorePost(), MC_SemaphorePost_chdl(), MC_SemaphoreWait(), MC_SemaphoreWait_chdl(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessage_chdl(), MC_SendAgentMigrationMessageFile(), MC_SendSteerCommand(), MC_SendSteerCommand_chdl(), MC_SetDefaultAgentStatus(), MC_SetDefaultAgentStatus_chdl(), MC_Steer(), MC_SyncDelete(), MC_SyncDelete_chdl(), MC_SyncInit(), MC_SyncInit_chdl(), MC_WaitAgent(), MC_WaitRetrieveAgent(), and MC_WaitSignal().

12.11.2.10 int agency_s::portno

Local port number

Definition at line 229 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

12.11.2.11 int agency_s::portnoc

Definition at line 230 of file libmc.h.

12.11.2.12 char* agency_s::priv_key_filename

Definition at line 238 of file libmc.h.

Referenced by MC_Initialize().

12.11.2.13 int agency_s::server

Definition at line 226 of file libmc.h.

Referenced by MC_Initialize().

12.11.2.14 int agency_s::stack_size[MC_THREAD_ALL]

Definition at line 236 of file libmc.h.

Referenced by MC_Initialize(), and mc_platform_Initialize().

12.11.2.15 int agency_s::threads

flag which determines which threads to start

Definition at line 234 of file libmc.h.

Referenced by MC_End(), MC_Initialize(), and mc_platform_Initialize().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/libmc.h](#)

12.12 agent_datastate_s Struct Reference

```
#include <agent_datastate.h>
```

Data Fields

- char ** agent_code_ids
- char ** agent_codes
- char * agent_code
- agent_task_p * tasks
- mxml_node_t * xml_agent_root
- mxml_node_t * xml_root
- int task_progress
- int return_data
- int number_of_tasks
- int persistent
- int init_agent_status
- int progress_modifier

12.12.1 Detailed Description

Definition at line 41 of file agent_datastate.h.

12.12.2 Field Documentation

12.12.2.1 char* agent_datastate_s::agent_code

Definition at line 45 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), agent_datastate_New(), agent_xml_parse__agent_code(), agent_xml_parse__tasks(), MC_ComposeAgentS(), MC_PrintAgentCode(), and MC_RetrieveAgentCode().

12.12.2.2 char** agent_datastate_s::agent_code_ids

Definition at line 43 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), agent_datastate_Destroy(), agent_xml_compose__agent_code(), agent_xml_parse__agent_code(), agent_xml_parse__tasks(), and MC_ComposeAgentS().

12.12.2.3 char** agent_datastate_s::agent_codes

Definition at line 44 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), agent_datastate_Destroy(), agent_xml_compose__agent_code(), agent_xml_parse__agent_code(), agent_xml_parse__tasks(), and MC_ComposeAgentS().

12.12.2.4 int agent_datastate_s::init_agent_status

Definition at line 63 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), and agent_datastate_New().

12.12.2.5 int agent_datastate_s::number_of_tasks

Definition at line 59 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), agent_datastate_Destroy(), agent_datastate_New(), agent_xml_compose__tasks(), agent_xml_parse__agent_code(), agent_xml_parse__tasks(), MC_ComposeAgentS(), MC_GetAgentNumTasks(), MC_GetAgentReturnData(), MC_PrintAgentCode(), and message_InitializeFromAgent().

12.12.2.6 int agent_datastate_s::persistent

Definition at line 62 of file agent_datastate.h.

Referenced by acc_MessageHandlerThread(), agent_datastate_Copy(), agent_datastate_New(), agent_xml_compose__task(), and MC_ComposeAgentS().

12.12.2.7 int agent_datastate_s::progress_modifier

Definition at line 71 of file agent_datastate.h.

Referenced by agent_datastate_New(), and MC_MigrateAgent().

12.12.2.8 int agent_datastate_s::return_data

Definition at line 56 of file agent_datastate.h.

Referenced by agent_datastate_Copy(), and agent_datastate_New().

12.12.2.9 int agent_datastate_s::task_progress

Definition at line 55 of file agent_datastate.h.

Referenced by agent_AddPersistentVariable(), agent_datastate_Copy(), agent_datastate_New(), agent_RunChScriptThread(), agent_xml_compose__tasks(), agent_xml_parse__agent_code(), agent_xml_parse__tasks(), interpreter_variable_data_InitializeFromAgent(), MC_AgentVariableRetrieve(), MC_AgentVariableRetrieveInfo(), MC_AgentVariableSave(), MC_MigrateAgent(), MC_PrintAgentCode(), MC_RetrieveAgentCode(), MC_SaveData_chdl(), and message_InitializeFromAgent().

12.12.2.10 agent_task_p* agent_datastate_s::tasks

Definition at line 48 of file agent_datastate.h.

Referenced by agent_AddPersistentVariable(), agent_datastate_Copy(), agent_datastate_Destroy(), agent_datastate_New(), agent_xml_compose__task(), agent_xml_parse__agent_code(), agent_xml_parse__data(), agent_xml_parse__task(), agent_xml_parse__tasks(), interpreter_variable_data_InitializeFromAgent(), MC_AgentVariableRetrieve(), MC_AgentVariableRetrieveInfo(), MC_AgentVariableSave(), MC_ComposeAgentS(), MC_GetAgentReturnData(), MC_MigrateAgent(), MC_SaveData_chdl(), and message_InitializeFromAgent().

12.12.2.11 mxml_node_t* agent_datastate_s::xml_agent_root

Definition at line 51 of file agent_datastate.h.

Referenced by `agent_datastate_New()`, `agent_Initialize()`, `agent_xml_parse()`, and `MC_GetAgentXMLString()`.

12.12.2.12 `mxml_node_t* agent_datastate_s::xml_root`

Definition at line 52 of file `agent_datastate.h`.

Referenced by `agent_datastate_Destroy()`, `agent_datastate_New()`, `agent_Initialize()`, and `agent_return_xml_parse()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/agent_datastate.h`

12.13 agent_mailbox_s Struct Reference

```
#include <agent_mailbox.h>
```

Data Fields

- struct mail_queue_s * [mail_queue](#)

12.13.1 Detailed Description

Definition at line 8 of file agent_mailbox.h.

12.13.2 Field Documentation

12.13.2.1 struct mail_queue_s* agent_mailbox_s::mail_queue [read]

Definition at line 10 of file agent_mailbox.h.

Referenced by agent_mailbox_Copy(), agent_mailbox_Destroy(), agent_mailbox_New(), agent_mailbox_Post(), agent_mailbox_Retrieve(), and agent_mailbox_WaitRetrieve().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[agent_mailbox.h](#)

12.14 agent_s Struct Reference

```
#include <agent.h>
```

Data Fields

- u_long [id](#)
- char * [name](#)
- u_long [connect_id](#)
- time_t [arrival_time](#)
- char * [owner](#)
- char * [home](#)
- char * [sender](#)
- int [home_port](#)
- char * [wg_code](#)
- char * [agent_address](#)
- int [orphan](#)
- [agent_datastate_p](#) [datastate](#)
- enum [MC_AgentType_e](#) [agent_type](#)
- enum [MC_AgentStatus_e](#) [agent_status](#)
- int [return_data](#)
- ChInterp_t * [agent_interp](#)
- MUTEX_T * [run_lock](#)
- int [agent_thread_id](#)
- THREAD_T [agent_thread](#)
- [agent_mailbox_p](#) [mailbox](#)
- int [agent_pipe_active](#)
- int [agent_ready_to_send](#)
- int [agent_pipe_ready_to_read](#)
- int [agent_script_ready](#)
- int [agent_persistent](#)
- struct [mc_platform_s](#) * [mc_platform](#)
- MUTEX_T * [lock](#)
- int [binary](#)

12.14.1 Detailed Description

Definition at line 48 of file [agent.h](#).

12.14.2 Field Documentation

12.14.2.1 char* agent_s::agent_address

Definition at line 66 of file [agent.h](#).

Referenced by [agent_Initialize\(\)](#), and [agent_NewBinary\(\)](#).

12.14.2.2 ChInterp_t* agent_s::agent_interp

Definition at line 79 of file agent.h.

Referenced by agent_AddPersistentVariable(), agent_Copy(), agent_Destroy(), agent_RunChScriptThread(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), MC_CallAgentFunc(), MC_CallAgentFuncArg(), MC_CallAgentFuncV(), MC_CallAgentFuncVar(), MC_GetAgentExecEngine(), MC_TerminateAgent(), and MC_TerminateAgentWG().

12.14.2.3 int agent_s::agent_persistent

Definition at line 99 of file agent.h.

Referenced by agent_Copy().

12.14.2.4 int agent_s::agent_pipe_active

Definition at line 95 of file agent.h.

Referenced by agent_Initialize(), and agent_NewBinary().

12.14.2.5 int agent_s::agent_pipe_ready_to_read

Definition at line 97 of file agent.h.

Referenced by agent_Initialize(), and agent_NewBinary().

12.14.2.6 int agent_s::agent_ready_to_send

Definition at line 96 of file agent.h.

Referenced by agent_Initialize(), and agent_NewBinary().

12.14.2.7 int agent_s::agent_script_ready

Definition at line 98 of file agent.h.

Referenced by agent_Initialize(), and agent_NewBinary().

12.14.2.8 enum MC_AgentStatus_e agent_s::agent_status

Definition at line 74 of file agent.h.

Referenced by acc_MessageHandlerThread(), agent_Copy(), agent_Destroy(), agent_Initialize(), agent_NewBinary(), agent_queue_Flush(), agent_RunChScript(), ams_ManageAgentList(), ams_Print(), AP_QUEUE_STD_DEFN_TEMPLATE(), MC_ComposeAgentS(), MC_GetAgentStatus(), MC_RetrieveAgent(), MC_SendAgentMigrationMessageFile(), and MC_SetAgentStatus().

12.14.2.9 THREAD_T agent_s::agent_thread

Definition at line 89 of file agent.h.

Referenced by agent_RunChScript().

12.14.2.10 int agent_s::agent_thread_id

Definition at line 85 of file agent.h.

Referenced by agent_Initialize(), and agent_NewBinary().

12.14.2.11 enum MC_AgentType_e agent_s::agent_type

Definition at line 73 of file agent.h.

Referenced by agent_Copy(), agent_Initialize(), agent_xml_compose__message(), MC_ComposeAgentS(), and MC_GetAgentType().

12.14.2.12 time_t agent_s::arrival_time

Definition at line 55 of file agent.h.

Referenced by agent_Copy(), agent_Initialize(), agent_NewBinary(), and MC_GetAgentArrivalTime().

12.14.2.13 int agent_s::binary

Definition at line 105 of file agent.h.

Referenced by agent_NewBinary(), and ams_ManageAgentList().

12.14.2.14 u_long agent_s::connect_id

Definition at line 53 of file agent.h.

Referenced by ams_Print().

12.14.2.15 agent_datastate_p agent_s::datastate

Definition at line 72 of file agent.h.

Referenced by acc_MessageHandlerThread(), agent_AddPersistentVariable(), agent_Copy(), agent_Destroy(), agent_Initialize(), agent_return_xml_parse(), agent_RunChScriptThread(), agent_xml_compose__agent_code(), agent_xml_compose__task(), agent_xml_compose__tasks(), agent_xml_parse(), agent_xml_parse__agent_code(), agent_xml_parse__data(), agent_xml_parse__task(), agent_xml_parse__tasks(), interpreter_variable_data_InitializeFromAgent(), MC_AgentVariableRetrieve(), MC_AgentVariableRetrieveInfo(), MC_AgentVariableSave(), MC_ComposeAgentS(), MC_GetAgentNumTasks(), MC_GetAgentReturnData(), MC_GetAgentXMLString(), MC_MigrateAgent(), MC_PrintAgentCode(), MC_RetrieveAgentCode(), MC_SaveData_chdl(), and message_InitializeFromAgent().

12.14.2.16 char* agent_s::home

Definition at line 61 of file agent.h.

Referenced by agent_Copy(), agent_Destroy(), agent_Initialize(), agent_NewBinary(), agent_xml_compose__home(), agent_xml_parse__home(), MC_ComposeAgentS(), and message_InitializeFromAgent().

12.14.2.17 int agent_s::home_port

Definition at line 63 of file agent.h.

Referenced by agent_Copy().

12.14.2.18 u_long agent_s::id

Definition at line 51 of file agent.h.

Referenced by agent_Copy(), agent_Initialize(), agent_NewBinary(), agent_queue_Flush(), agent_RunChScriptThread(), ams_Print(), AP_QUEUE_STD_DEFN_TEMPLATE(), MC_GetAgentID(), and MC_RegisterService().

12.14.2.19 MUTEX_T* agent_s::lock

Definition at line 103 of file agent.h.

Referenced by acc_MessageHandlerThread(), agent_Copy(), agent_Destroy(), agent_Initialize(), agent_New(), agent_NewBinary(), ams_ManageAgentList(), MC_GetAgentName(), MC_GetAgentStatus(), MC_PrintAgentCode(), MC_RetrieveAgentCode(), and MC_SetAgentStatus().

12.14.2.20 agent_mailbox_p agent_s::mailbox

Definition at line 92 of file agent.h.

Referenced by acc_connection_Thread(), agent_Copy(), agent_Destroy(), agent_Initialize(), agent_NewBinary(), MC_AclPost(), MC_AclRetrieve(), and MC_AclWaitRetrieve().

12.14.2.21 struct mc_platform_s* agent_s::mc_platform [read]

Definition at line 101 of file agent.h.

Referenced by acc_connection_Thread(), agent_Destroy(), agent_Initialize(), agent_NewBinary(), agent_RunChScript(), agent_RunChScriptThread(), MC_AddAgent(), and MC_SetAgentStatus().

12.14.2.22 char* agent_s::name

Definition at line 52 of file agent.h.

Referenced by acc_MessageHandlerThread(), agent_AddPersistentVariable(), agent_Copy(), agent_Destroy(), agent_queue_Flush(), agent_RunChScriptThread(), agent_xml_compose__name(), agent_xml_parse__name(), ams_ManageAgentList(), AP_QUEUE_STD_DEFN_TEMPLATE(), MC_AddStationaryAgent(), MC_ComposeAgentS(), MC_GetAgentName(), MC_RegisterService(), and message_InitializeFromAgent().

12.14.2.23 `int agent_s::orphan`

Definition at line 68 of file agent.h.

Referenced by `agent_Copy()`, `agent_Initialize()`, `agent_NewBinary()`, `ams_ManageAgentList()`, `MC_ComposeAgentS()`, and `MC_SetAgentStatus()`.

12.14.2.24 `char* agent_s::owner`

Definition at line 60 of file agent.h.

Referenced by `agent_Copy()`, `agent_Destroy()`, `agent_xml_compose__owner()`, `agent_xml_parse__owner()`, and `MC_ComposeAgentS()`.

12.14.2.25 `int agent_s::return_data`

Definition at line 76 of file agent.h.

Referenced by `agent_Copy()`.

12.14.2.26 `MUTEX_T* agent_s::run_lock`

Definition at line 82 of file agent.h.

Referenced by `agent_Copy()`, `agent_Destroy()`, `agent_Initialize()`, `agent_New()`, `agent_NewBinary()`, `ams_ManageAgentList()`, `interpreter_variable_data_Initialize()`, `MC_CallAgentFunc()`, `MC_CallAgentFuncArg()`, `MC_CallAgentFuncV()`, and `MC_CallAgentFuncVar()`.

12.14.2.27 `char* agent_s::sender`

Definition at line 62 of file agent.h.

Referenced by `agent_Destroy()`, `agent_Initialize()`, `agent_NewBinary()`, and `agent_xml_parse__sender()`.

12.14.2.28 `char* agent_s::wg_code`

Definition at line 64 of file agent.h.

Referenced by `agent_Destroy()`, `agent_xml_compose__wg_code()`, `agent_xml_parse__wg_code()`, `MC_ComposeAgentS()`, `MC_DeleteAgent()`, `MC_DeleteAgentWG()`, and `MC_TerminateAgentWG()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/agent.h`

12.15 agent_task_s Struct Reference

```
#include <agent_task.h>
```

Data Fields

- [int number_of_elements](#)
- [int size_of_element_array](#)
- [int persistent](#)
- [int init_agent_status](#)
- [char * var_name](#)
- [char * server_name](#)
- [char * code_id](#)
- [interpreter_variable_data_t * agent_return_data](#)
- [struct agent_variable_list_s * agent_variable_list](#)
- [char ** saved_variables](#)
- [int num_saved_variables](#)

12.15.1 Detailed Description

Definition at line 40 of file agent_task.h.

12.15.2 Field Documentation

12.15.2.1 [interpreter_variable_data_t* agent_task_s::agent_return_data](#)

Definition at line 52 of file agent_task.h.

Referenced by [agent_task_Copy\(\)](#), [agent_task_Destroy\(\)](#), [agent_xml_compose__task\(\)](#), [agent_xml_parse__data\(\)](#), and [MC_GetAgentReturnData\(\)](#).

12.15.2.2 [struct agent_variable_list_s* agent_task_s::agent_variable_list](#) **[read]**

Definition at line 53 of file agent_task.h.

Referenced by [agent_AddPersistentVariable\(\)](#), [agent_task_Copy\(\)](#), [agent_task_Destroy\(\)](#), [agent_task_New\(\)](#), [agent_xml_compose__task\(\)](#), [agent_xml_parse__data\(\)](#), [MC_AgentVariableRetrieve\(\)](#), [MC_AgentVariableRetrieveInfo\(\)](#), and [MC_SaveData_chdl\(\)](#).

12.15.2.3 [char* agent_task_s::code_id](#)

Definition at line 49 of file agent_task.h.

Referenced by [agent_task_Copy\(\)](#), [agent_task_Destroy\(\)](#), [agent_xml_compose__task\(\)](#), [agent_xml_parse__agent_code\(\)](#), and [agent_xml_parse__task\(\)](#).

12.15.2.4 [int agent_task_s::init_agent_status](#)

Definition at line 45 of file agent_task.h.

Referenced by [agent_task_Copy\(\)](#).

12.15.2.5 `int agent_task_s::num_saved_variables`

Definition at line 55 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_New()`, and `MC_AgentVariableSave()`.

12.15.2.6 `int agent_task_s::number_of_elements`

Definition at line 42 of file `agent_task.h`.

Referenced by `agent_task_Copy()`.

12.15.2.7 `int agent_task_s::persistent`

Definition at line 44 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_xml_compose__task()`, `agent_xml_parse__data()`, and `agent_xml_parse__task()`.

12.15.2.8 `char** agent_task_s::saved_variables`

Definition at line 54 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_Destroy()`, `agent_task_New()`, and `MC_AgentVariableSave()`.

12.15.2.9 `char* agent_task_s::server_name`

Definition at line 48 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_Destroy()`, `agent_xml_compose__task()`, `agent_xml_parse__task()`, `MC_ComposeAgentS()`, `MC_MigrateAgent()`, and `message_InitializeFromAgent()`.

12.15.2.10 `int agent_task_s::size_of_element_array`

Definition at line 43 of file `agent_task.h`.

Referenced by `agent_task_Copy()`.

12.15.2.11 `char* agent_task_s::var_name`

Definition at line 47 of file `agent_task.h`.

Referenced by `agent_task_Copy()`, `agent_task_Destroy()`, `agent_xml_compose__task()`, `agent_xml_parse__task()`, `interpreter_variable_data_InitializeFromAgent()`, and `MC_ComposeAgentS()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/agent_task.h`

12.16 agent_thread_arg_s Struct Reference

```
#include <libmc.h>
```

Data Fields

- void * [args](#)
- struct [agent_s](#) * [agent](#)
- [MCAgency_t](#) [attr](#)
- [THREAD_T](#) [thread](#)

12.16.1 Detailed Description

Definition at line 264 of file libmc.h.

12.16.2 Field Documentation

12.16.2.1 struct agent_s* agent_thread_arg_s::agent [read]

Definition at line 266 of file libmc.h.

Referenced by [MC_AddStationaryAgent\(\)](#), [MC_ComposeAgentFromFileS\(\)](#), and [MC_ComposeAgentS\(\)](#).

12.16.2.2 void* agent_thread_arg_s::args

Definition at line 265 of file libmc.h.

Referenced by [MC_AddStationaryAgent\(\)](#).

12.16.2.3 MCAgency_t agent_thread_arg_s::attr

Definition at line 267 of file libmc.h.

Referenced by [MC_AddStationaryAgent\(\)](#).

12.16.2.4 THREAD_T agent_thread_arg_s::thread

Definition at line 268 of file libmc.h.

Referenced by [MC_AddStationaryAgent\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/libmc.h](#)

12.17 AP_GENERIC_s Struct Reference

```
#include <ap_queue_template.h>
```

Data Fields

- void * [none](#)

12.17.1 Detailed Description

Definition at line 41 of file `ap_queue_template.h`.

12.17.2 Field Documentation

12.17.2.1 void* AP_GENERIC_s::none

Definition at line 41 of file `ap_queue_template.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/ap_queue_template.h`

12.18 arc4_context Struct Reference

ARC4 context structure.

```
#include <arc4.h>
```

Data Fields

- `int x`
- `int y`
- unsigned char `m` [256]

12.18.1 Detailed Description

ARC4 context structure.

Definition at line 10 of file arc4.h.

12.18.2 Field Documentation

12.18.2.1 unsigned char arc4_context::m[256]

permutation table

Definition at line 14 of file arc4.h.

Referenced by arc4_crypt(), and arc4_setup().

12.18.2.2 int arc4_context::x

permutation index

Definition at line 12 of file arc4.h.

Referenced by arc4_crypt(), and arc4_setup().

12.18.2.3 int arc4_context::y

permutation index

Definition at line 13 of file arc4.h.

Referenced by arc4_crypt(), and arc4_setup().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/[arc4.h](#)

12.19 barrier_node_s Struct Reference

```
#include <barrier.h>
```

Data Fields

- `MUTEX_T * lock`
- `COND_T * cond`
- `int id`
- `int num_registered`
- `int num_waiting`

12.19.1 Detailed Description

Definition at line 42 of file barrier.h.

12.19.2 Field Documentation

12.19.2.1 `COND_T* barrier_node_s::cond`

Definition at line 44 of file barrier.h.

Referenced by `barrier_node_Destroy()`, `barrier_node_Initialize()`, and `MC_Barrier()`.

12.19.2.2 `int barrier_node_s::id`

Definition at line 45 of file barrier.h.

Referenced by `barrier_node_Initialize()`, `barrier_queue_Add()`, and `barrier_queue_Delete()`.

12.19.2.3 `MUTEX_T* barrier_node_s::lock`

Definition at line 43 of file barrier.h.

Referenced by `barrier_node_Destroy()`, `barrier_node_Initialize()`, and `MC_Barrier()`.

12.19.2.4 `int barrier_node_s::num_registered`

Definition at line 46 of file barrier.h.

Referenced by `barrier_node_Initialize()`, and `MC_Barrier()`.

12.19.2.5 `int barrier_node_s::num_waiting`

Definition at line 47 of file barrier.h.

Referenced by `barrier_node_Initialize()`, and `MC_Barrier()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/barrier.h`

12.20 barrier_queue_s Struct Reference

```
#include <barrier.h>
```

Data Fields

- `RWLOCK_T * lock`
- `list_p list`
- `int size`

12.20.1 Detailed Description

Definition at line 51 of file barrier.h.

12.20.2 Field Documentation

12.20.2.1 list_p barrier_queue_s::list

Definition at line 54 of file barrier.h.

Referenced by `barrier_queue_Add()`, `barrier_queue_Delete()`, `barrier_queue_Destroy()`, `barrier_queue_Get()`, `barrier_queue_New()`, and `barrier_queue_Pop()`.

12.20.2.2 RWLOCK_T* barrier_queue_s::lock

Definition at line 52 of file barrier.h.

Referenced by `barrier_queue_Add()`, `barrier_queue_Delete()`, `barrier_queue_Destroy()`, `barrier_queue_Get()`, and `barrier_queue_New()`.

12.20.2.3 int barrier_queue_s::size

Definition at line 55 of file barrier.h.

Referenced by `barrier_queue_Add()`, and `barrier_queue_Delete()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/barrier.h`

12.21 EmbeddedCh::ChBlock_t Struct Reference

Data Fields

- [int event_](#)
- [int count](#)
- [int level](#)
- [int linecurrent](#)
- [int linefuncbegin](#)
- [int linefuncend](#)
- [String source](#)
- [String funcname](#)
- [String classname](#)
- [int isconstructor](#)
- [int isdestructor](#)

12.21.1 Detailed Description

Definition at line 46 of file EmbedCh.cs.

12.21.2 Field Documentation

12.21.2.1 String EmbeddedCh::ChBlock_t::classname

Definition at line 58 of file EmbedCh.cs.

12.21.2.2 int EmbeddedCh::ChBlock_t::count

Definition at line 48 of file EmbedCh.cs.

12.21.2.3 int EmbeddedCh::ChBlock_t::event_

Definition at line 47 of file EmbedCh.cs.

12.21.2.4 String EmbeddedCh::ChBlock_t::funcname

Definition at line 57 of file EmbedCh.cs.

12.21.2.5 int EmbeddedCh::ChBlock_t::isconstructor

Definition at line 59 of file EmbedCh.cs.

12.21.2.6 int EmbeddedCh::ChBlock_t::isdestructor

Definition at line 60 of file EmbedCh.cs.

12.21.2.7 int EmbeddedCh::ChBlock_t::level

Definition at line 49 of file EmbedCh.cs.

12.21.2.8 int EmbeddedCh::ChBlock_t::linecurrent

Definition at line 52 of file EmbedCh.cs.

12.21.2.9 int EmbeddedCh::ChBlock_t::linefuncbegin

Definition at line 53 of file EmbedCh.cs.

12.21.2.10 int EmbeddedCh::ChBlock_t::linefuncend

Definition at line 54 of file EmbedCh.cs.

12.21.2.11 String EmbeddedCh::ChBlock_t::source

Definition at line 55 of file EmbedCh.cs.

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/[EmbedCh.cs](#)

12.22 EmbeddedCh::ChInfo_t Struct Reference

Data Fields

- String [edition](#)
- String [releasedate](#)
- String [version](#)
- uint [vermajor](#)
- uint [verminor](#)
- uint [vermicro](#)
- uint [verbuild](#)

12.22.1 Detailed Description

Definition at line 8 of file Ch.cs.

12.22.2 Field Documentation

12.22.2.1 String EmbeddedCh::ChInfo_t::edition

Definition at line 9 of file Ch.cs.

12.22.2.2 String EmbeddedCh::ChInfo_t::releasedate

Definition at line 10 of file Ch.cs.

12.22.2.3 uint EmbeddedCh::ChInfo_t::verbuild

Definition at line 15 of file Ch.cs.

12.22.2.4 uint EmbeddedCh::ChInfo_t::vermajor

Definition at line 12 of file Ch.cs.

12.22.2.5 uint EmbeddedCh::ChInfo_t::vermicro

Definition at line 14 of file Ch.cs.

12.22.2.6 uint EmbeddedCh::ChInfo_t::verminor

Definition at line 13 of file Ch.cs.

12.22.2.7 String EmbeddedCh::ChInfo_t::version

Definition at line 11 of file Ch.cs.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/Ch.cs](#)

12.23 EmbeddedCh::ChInterp Class Reference

Public Member Functions

- delegate void [ChCallback](#) (IntPtr [interp](#), ref [ChBlock_t](#) calldata, IntPtr clientdata)
- [ChInterp](#) ()
- [int Initialize](#) ()
- [int RunScript](#) (String[] argv)
- [int AppendRunScript](#) (String argv)
- [int AppendRunScriptFile](#) (String filename)
- [int InitGlobalVar](#) (Int32 flag)
- [int ParseScript](#) (String[] argv)
- [int ExecScript](#) (String progname)
- [int ExecScriptM](#) (String progname)
- [int RunScriptM](#) (String[] argv)
- [int AppendParseScript](#) (String code)
- [int AppendParseScriptFile](#) (String filename)
- [int End](#) ()
- [int ExprCalc](#) (String expr, [ChType_t](#) datatype, IntPtr result)
- [int ExprEval](#) (String expr)
- [int ExprParse](#) (String expr)
- IntPtr [ExprValue](#) (String expr, IntPtr result)
- [int DeleteExprValue](#) (IntPtr vn)
- Int32 [SetVar](#) (String name, [ChType_t](#) atype)
- Int32 [Close](#) (Int32 fildes)
- Int32 [Reopen](#) (String filename, String mode, Int32 fildes)
- Int32 [Flush](#) (Int32 fildes)
- Int32 [Abort](#) ()
- Int32 [DeclareVar](#) (String declaration)
- Int32 [DeclareTypedef](#) (String name)
- Int32 [DeclareFunc](#) (String funcprototype, IntPtr funcptr)
- Int32 [SetGlobalUserData](#) (IntPtr userdata)
- IntPtr [GetGlobalUserData](#) ()
- Int32 [AddCallback](#) (Int32 _event, ChCallback callback, IntPtr clientdata, Int32 count)
- Int32 [ChangeStack](#) (Int32 level, ref [ChBlock_t](#) calldata)
- Int32 [StackLevel](#) (ref Int32 clevel, ref Int32 hlevel)
- String [StackName](#) (Int32 level, ref Int32 isfunc, String[] classname)
- Int32 [GlobalSymbolTotalNum](#) ()
- Int32 [GlobalSymbolIndexByName](#) (String name)
- IntPtr [GlobalSymbolAddrByIndex](#) (Int32 index)
- String [GlobalSymbolNameByIndex](#) (Int32 index)
- [ChType_t DataType](#) (String expr)
- Int32 [DataSize](#) (String expr)
- [ChType_t ArrayType](#) (String expr)
- Int32 [ArrayDim](#) (String expr)
- Int32 [ArrayExtent](#) (String expr, Int32 index)
- Int32 [ArrayNum](#) (String expr)
- [ChVarType_t VarType](#) (String name)
- [ChFuncType_t FuncType](#) (String name)
- Int32 [IsFuncVarArg](#) (String name)

- Int32 [FuncArgNum](#) (String name)
- IntPtr [SymbolAddrByName](#) (String name)
- Int32 [SymbolTotalNum](#) ()
- Int32 [SymbolIndexByName](#) (String name)
- IntPtr [SymbolAddrByIndex](#) (Int32 index)
- String [SymbolNameByIndex](#) (Int32 index)
- IntPtr [UserDefinedTag](#) (String expr)
- Int32 [UserDefinedInfo](#) (IntPtr udtag, ref [ChUserDefinedInfo_t](#) udtinfo)
- Int32 [UserDefinedMemInfoByName](#) (IntPtr udtag, String memname, ref [ChMemInfo_t](#) meminfo)
- Int32 [UserDefinedMemInfoByIndex](#) (IntPtr udtag, Int32 index, ref [ChMemInfo_t](#) meminfo)
- String [UserDefinedName](#) (String name)
- Int32 [UserDefinedSize](#) (String name)
- [ChType_t](#) [FuncArgDataType](#) (String funcname, Int32 argnum)
- [ChType_t](#) [FuncArgArrayType](#) (String funcname, Int32 argnum)
- Int32 [FuncArgArrayDim](#) (String funcname, Int32 argnum)
- Int32 [FuncArgArrayExtent](#) (String funcname, Int32 argnum, Int32 index)
- Int32 [FuncArgArrayNum](#) (String funcname, Int32 argnum)
- Int32 [FuncArgIsFunc](#) (String funcname, Int32 argnum)
- Int32 [FuncArgIsFuncVarArg](#) (String funcname, Int32 argnum)
- Int32 [FuncArgFuncArgNum](#) (String funcname, Int32 argnum)
- String [FuncArgUserDefinedName](#) (String funcname, Int32 argnum)
- Int32 [FuncArgUserDefinedSize](#) (String funcname, Int32 argnum)

Data Fields

- const String [chdll](#) = "embedchdll.dll"

Properties

- bool [UseOptions](#) [get, set]
- [ChShellType](#) [ShellType](#) [get, set]
- String [ChHome](#) [get, set]

Private Member Functions

- static Int32 [_Ch_Initialize](#) (ref IntPtr interpp, ref [ChOptions_t](#) option)
- static Int32 [_Ch_Initialize](#) (ref IntPtr interpp, IntPtr option)
- static Int32 [_Ch_InitGlobalVar](#) (IntPtr [interp](#), Int32 flag)
- static Int32 [_Ch_ParseScript](#) (IntPtr [interp](#), String[] argv)
- static Int32 [_Ch_ExecScript](#) (IntPtr [interp](#), String progname)
- static Int32 [_Ch_ExecScriptM](#) (IntPtr [interp](#), String progname)
- static Int32 [_Ch_RunScript](#) (IntPtr [interp](#), String[] argv)
- static Int32 [_Ch_RunScriptM](#) (IntPtr [interp](#), String[] argv)
- static Int32 [_Ch_AppendParseScript](#) (IntPtr [interp](#), String code)
- static Int32 [_Ch_AppendParseScriptFile](#) (IntPtr [interp](#), String filename)
- static Int32 [_Ch_AppendRunScript](#) (IntPtr [interp](#), String code)
- static Int32 [_Ch_AppendRunScriptFile](#) (IntPtr [interp](#), String filename)
- static Int32 [_Ch_ExprCalc](#) (IntPtr [interp](#), String expr, [ChType_t](#) datatype, IntPtr result)
- static Int32 [_Ch_ExprEval](#) (IntPtr [interp](#), String expr)

- static Int32 [_Ch_ExprParse](#) (IntPtr [interp](#), String expr)
- static IntPtr [_Ch_ExprValue](#) (IntPtr [interp](#), String expr, IntPtr result)
- static Int32 [_Ch_DeleteExprValue](#) (IntPtr [interp](#), IntPtr vn)
- static Int32 [_Ch_SetVar](#) (IntPtr [interp](#), String name, [ChType_t](#) atype)
- static Int32 [_Ch_Close](#) (IntPtr [interp](#), Int32 fildes)
- static Int32 [_Ch_Reopen](#) (IntPtr [interp](#), String filename, String mode, Int32 fildes)
- static Int32 [_Ch_Flush](#) (IntPtr [interp](#), Int32 fildes)
- static Int32 [_Ch_End](#) (IntPtr [interp](#))
- static Int32 [_Ch_Abort](#) (IntPtr [interp](#))
- static Int32 [_Ch_DeclareVar](#) (IntPtr [interp](#), String declaration)
- static Int32 [_Ch_DeclareTypedef](#) (IntPtr [interp](#), String name)
- static Int32 [_Ch_DeclareFunc](#) (IntPtr [interp](#), String funcprototype, IntPtr funcptr)
- static Int32 [_Ch_SetGlobalUserData](#) (IntPtr [interp](#), IntPtr userdata)
- static IntPtr [_Ch_GetGlobalUserData](#) (IntPtr [interp](#))
- static Int32 [_Ch_AddCallback](#) (IntPtr [interp](#), Int32 _event, ChCallback callback, IntPtr clientdata, Int32 count)
- static Int32 [_Ch_ChangeStack](#) (IntPtr [interp](#), Int32 level, ref [ChBlock_t](#) calldata)
- static Int32 [_Ch_StackLevel](#) (IntPtr [interp](#), ref Int32 clevel, ref Int32 hlevel)
- static String [_Ch_StackName](#) (IntPtr [interp](#), Int32 level, ref Int32 isfunc, String[] classname)
- static Int32 [_Ch_GlobalSymbolTotalNum](#) (IntPtr [interp](#))
- static Int32 [_Ch_GlobalSymbolIndexByName](#) (IntPtr [interp](#), String name)
- static IntPtr [_Ch_GlobalSymbolAddrByIndex](#) (IntPtr [interp](#), Int32 index)
- static String [_Ch_GlobalSymbolNameByIndex](#) (IntPtr [interp](#), Int32 index)
- static [ChType_t](#) [_Ch_DataType](#) (IntPtr [interp](#), String expr)
- static Int32 [_Ch_DataSize](#) (IntPtr [interp](#), String expr)
- static [ChType_t](#) [_Ch_ArrayType](#) (IntPtr [interp](#), String expr)
- static Int32 [_Ch_ArrayDim](#) (IntPtr [interp](#), String expr)
- static Int32 [_Ch_ArrayExtent](#) (IntPtr [interp](#), String expr, Int32 index)
- static Int32 [_Ch_ArrayNum](#) (IntPtr [interp](#), String expr)
- static [ChVarType_t](#) [_Ch_VarType](#) (IntPtr [interp](#), String name)
- static [ChFuncType_t](#) [_Ch_FuncType](#) (IntPtr [interp](#), String name)
- static Int32 [_Ch_IsFuncVarArg](#) (IntPtr [interp](#), String name)
- static Int32 [_Ch_FuncArgNum](#) (IntPtr [interp](#), String name)
- static IntPtr [_Ch_SymbolAddrByName](#) (IntPtr [interp](#), String name)
- static Int32 [_Ch_SymbolTotalNum](#) (IntPtr [interp](#))
- static Int32 [_Ch_SymbolIndexByName](#) (IntPtr [interp](#), String name)
- static IntPtr [_Ch_SymbolAddrByIndex](#) (IntPtr [interp](#), Int32 index)
- static String [_Ch_SymbolNameByIndex](#) (IntPtr [interp](#), Int32 index)
- static IntPtr [_Ch_UserDefinedTag](#) (IntPtr [interp](#), String expr)
- static Int32 [_Ch_UserDefinedInfo](#) (IntPtr [interp](#), IntPtr udtag, ref [ChUserDefinedInfo_t](#) udinfo)
- static Int32 [_Ch_UserDefinedMemInfoByName](#) (IntPtr [interp](#), IntPtr udtag, String memname, ref [ChMemInfo_t](#) meminfo)
- static Int32 [_Ch_UserDefinedMemInfoByIndex](#) (IntPtr [interp](#), IntPtr udtag, Int32 index, ref [ChMemInfo_t](#) meminfo)
- static String [_Ch_UserDefinedName](#) (IntPtr [interp](#), String name)
- static Int32 [_Ch_UserDefinedSize](#) (IntPtr [interp](#), String name)
- static [ChType_t](#) [_Ch_FuncArgDataType](#) (IntPtr [interp](#), String funcname, Int32 argnum)
- static [ChType_t](#) [_Ch_FuncArgArrayType](#) (IntPtr [interp](#), String funcname, Int32 argnum)
- static Int32 [_Ch_FuncArgArrayDim](#) (IntPtr [interp](#), String funcname, Int32 argnum)
- static Int32 [_Ch_FuncArgArrayExtent](#) (IntPtr [interp](#), String funcname, Int32 argnum, Int32 index)

- static Int32 `_Ch_FuncArgArrayNum` (IntPtr *interp*, String funcname, Int32 argnum)
- static Int32 `_Ch_FuncArgIsFunc` (IntPtr *interp*, String funcname, Int32 argnum)
- static Int32 `_Ch_FuncArgIsFuncVarArg` (IntPtr *interp*, String funcname, Int32 argnum)
- static Int32 `_Ch_FuncArgFuncArgNum` (IntPtr *interp*, String funcname, Int32 argnum)
- static String `_Ch_FuncArgUserDefinedName` (IntPtr *interp*, String funcname, Int32 argnum)
- static Int32 `_Ch_FuncArgUserDefinedSize` (IntPtr *interp*, String funcname, Int32 argnum)
- static String `_Ch_Home` (IntPtr *interp*)
- static Int32 `_Ch_Version` (IntPtr *interp*, ChInfo_t *info)
- static IntPtr `_Ch_GlobalSymbolAddrByName` (IntPtr *interp*, String name)
- static IntPtr `_Ch_SymbolAddrByName` (IntPtr *interp*, String name)
- static Int32 `_Ch_CallFuncByAddrv` (IntPtr *interp*, IntPtr fptr, IntPtr retval, va_list ap)
- static Int32 `_Ch_CallFuncByNamev` (IntPtr *interp*, String name, IntPtr retval, va_list ap)
- static Int32 `_Ch_CallFuncByNameVar` (IntPtr *interp*, String name, IntPtr retval, ChVaList_t arglist)

Private Attributes

- IntPtr *interp* = IntPtr.Zero
- ChOptions_t *options* = new ChOptions_t()
- bool *useOptions* = false

12.23.1 Detailed Description

Definition at line 9 of file ChInterp.cs.

12.23.2 Constructor & Destructor Documentation

12.23.2.1 EmbeddedCh::ChInterp::ChInterp () [inline]

Definition at line 25 of file ChInterp.cs.

12.23.3 Member Function Documentation

12.23.3.1 static Int32 EmbeddedCh::ChInterp::_Ch_Abort (IntPtr *interp*) [private]

Referenced by Abort().

12.23.3.2 static Int32 EmbeddedCh::ChInterp::_Ch_AddCallback (IntPtr *interp*, Int32 *_event*, ChCallback *callback*, IntPtr *clientdata*, Int32 *count*) [private]

Referenced by AddCallback().

12.23.3.3 static Int32 EmbeddedCh::ChInterp::_Ch_AppendParseScript (IntPtr *interp*, String *code*) [private]

Referenced by AppendParseScript().

12.23.3.4 `static Int32 EmbeddedCh::ChInterp::_Ch_AppendParseScriptFile (IntPtr interp, String filename) [private]`

Referenced by AppendParseScriptFile().

12.23.3.5 `static Int32 EmbeddedCh::ChInterp::_Ch_AppendRunScript (IntPtr interp, String code) [private]`

Referenced by AppendRunScript().

12.23.3.6 `static Int32 EmbeddedCh::ChInterp::_Ch_AppendRunScriptFile (IntPtr interp, String filename) [private]`

Referenced by AppendRunScriptFile().

12.23.3.7 `static Int32 EmbeddedCh::ChInterp::_Ch_ArrayDim (IntPtr interp, String expr) [private]`

Referenced by ArrayDim().

12.23.3.8 `static Int32 EmbeddedCh::ChInterp::_Ch_ArrayExtent (IntPtr interp, String expr, Int32 index) [private]`

Referenced by ArrayExtent().

12.23.3.9 `static Int32 EmbeddedCh::ChInterp::_Ch_ArrayNum (IntPtr interp, String expr) [private]`

Referenced by ArrayNum().

12.23.3.10 `static ChType_t EmbeddedCh::ChInterp::_Ch_ArrayType (IntPtr interp, String expr) [private]`

Referenced by ArrayType().

12.23.3.11 `static Int32 EmbeddedCh::ChInterp::_Ch_CallFuncByAddrv (IntPtr interp, IntPtr fptr, IntPtr retval, va_list ap) [private]`

12.23.3.12 `static Int32 EmbeddedCh::ChInterp::_Ch_CallFuncByNamev (IntPtr interp, String name, IntPtr retval, va_list ap) [private]`

12.23.3.13 `static Int32 EmbeddedCh::ChInterp::_Ch_CallFuncByNameVar (IntPtr interp, String name, IntPtr retval, ChVaList_t arglist) [private]`

12.23.3.14 `static Int32 EmbeddedCh::ChInterp::_Ch_ChangeStack (IntPtr interp, Int32 level, ref ChBlock_t calldata) [private]`

Referenced by ChangeStack().

12.23.3.15 `static Int32 EmbeddedCh::ChInterp::_Ch_Close (IntPtr interp, Int32 fildes) [private]`

Referenced by Close().

12.23.3.16 `static Int32 EmbeddedCh::ChInterp::_Ch_DataSize (IntPtr interp, String expr) [private]`

Referenced by DataSize().

12.23.3.17 `static ChType_t EmbeddedCh::ChInterp::_Ch_DataType (IntPtr interp, String expr) [private]`

Referenced by DataType().

12.23.3.18 `static Int32 EmbeddedCh::ChInterp::_Ch_DeclareFunc (IntPtr interp, String funcprototype, IntPtr funcptr) [private]`

Referenced by DeclareFunc().

12.23.3.19 `static Int32 EmbeddedCh::ChInterp::_Ch_DeclareTypedef (IntPtr interp, String name) [private]`

Referenced by DeclareTypedef().

12.23.3.20 `static Int32 EmbeddedCh::ChInterp::_Ch_DeclareVar (IntPtr interp, String declaration) [private]`

Referenced by DeclareVar().

12.23.3.21 `static Int32 EmbeddedCh::ChInterp::_Ch_DeleteExprValue (IntPtr interp, IntPtr vn) [private]`

Referenced by DeleteExprValue().

12.23.3.22 `static Int32 EmbeddedCh::ChInterp::_Ch_End (IntPtr interp) [private]`

Referenced by End().

12.23.3.23 `static Int32 EmbeddedCh::ChInterp::_Ch_ExecScript (IntPtr interp, String progname) [private]`

Referenced by ExecScript().

12.23.3.24 `static Int32 EmbeddedCh::ChInterp::_Ch_ExecScriptM (IntPtr interp, String progname) [private]`

Referenced by ExecScriptM().

12.23.3.25 `static Int32 EmbeddedCh::ChInterp::_Ch_ExprCalc (IntPtr interp, String expr, ChType_t datatype, IntPtr result) [private]`

Referenced by ExprCalc().

12.23.3.26 `static Int32 EmbeddedCh::ChInterp::_Ch_ExprEval (IntPtr interp, String expr) [private]`

Referenced by ExprEval().

12.23.3.27 `static Int32 EmbeddedCh::ChInterp::_Ch_ExprParse (IntPtr interp, String expr) [private]`

Referenced by ExprParse().

12.23.3.28 `static IntPtr EmbeddedCh::ChInterp::_Ch_ExprValue (IntPtr interp, String expr, IntPtr result) [private]`

Referenced by ExprValue().

12.23.3.29 `static Int32 EmbeddedCh::ChInterp::_Ch_Flush (IntPtr interp, Int32 fildes) [private]`

Referenced by Flush().

12.23.3.30 `static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgArrayDim (IntPtr interp, String funcname, Int32 argnum) [private]`

Referenced by FuncArgArrayDim().

12.23.3.31 `static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgArrayExtent (IntPtr interp, String funcname, Int32 argnum, Int32 index) [private]`

Referenced by FuncArgArrayExtent().

12.23.3.32 `static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgArrayNum (IntPtr interp, String funcname, Int32 argnum) [private]`

Referenced by FuncArgArrayNum().

12.23.3.33 `static ChType_t EmbeddedCh::ChInterp::_Ch_FuncArgArrayType (IntPtr interp, String funcname, Int32 argnum) [private]`

Referenced by FuncArgArrayType().

12.23.3.34 static ChType_t EmbeddedCh::ChInterp::_Ch_FuncArgDataType (IntPtr *interp*, String *funcname*, Int32 *argnum*) [private]

Referenced by FuncArgDataType().

12.23.3.35 static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgFuncArgNum (IntPtr *interp*, String *funcname*, Int32 *argnum*) [private]

Referenced by FuncArgFuncArgNum().

12.23.3.36 static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgIsFunc (IntPtr *interp*, String *funcname*, Int32 *argnum*) [private]

Referenced by FuncArgIsFunc().

12.23.3.37 static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgIsFuncVarArg (IntPtr *interp*, String *funcname*, Int32 *argnum*) [private]

Referenced by FuncArgIsFuncVarArg().

12.23.3.38 static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgNum (IntPtr *interp*, String *name*) [private]

Referenced by FuncArgNum().

12.23.3.39 static String EmbeddedCh::ChInterp::_Ch_FuncArgUserDefinedName (IntPtr *interp*, String *funcname*, Int32 *argnum*) [private]

Referenced by FuncArgUserDefinedName().

12.23.3.40 static Int32 EmbeddedCh::ChInterp::_Ch_FuncArgUserDefinedSize (IntPtr *interp*, String *funcname*, Int32 *argnum*) [private]

Referenced by FuncArgUserDefinedSize().

12.23.3.41 static ChFuncType_t EmbeddedCh::ChInterp::_Ch_FuncType (IntPtr *interp*, String *name*) [private]

Referenced by FuncType().

12.23.3.42 static IntPtr EmbeddedCh::ChInterp::_Ch_GetGlobalUserData (IntPtr *interp*) [private]

Referenced by GetGlobalUserData().

12.23.3.43 `static IntPtr EmbeddedCh::ChInterp::_Ch_GlobalSymbolAddrByIndex (IntPtr interp, Int32 index) [private]`

Referenced by GlobalSymbolAddrByIndex().

12.23.3.44 `static IntPtr EmbeddedCh::ChInterp::_Ch_GlobalSymbolAddrByName (IntPtr interp, String name) [private]`

12.23.3.45 `static Int32 EmbeddedCh::ChInterp::_Ch_GlobalSymbolIndexByName (IntPtr interp, String name) [private]`

Referenced by GlobalSymbolIndexByName().

12.23.3.46 `static String EmbeddedCh::ChInterp::_Ch_GlobalSymbolNameByIndex (IntPtr interp, Int32 index) [private]`

Referenced by GlobalSymbolNameByIndex().

12.23.3.47 `static Int32 EmbeddedCh::ChInterp::_Ch_GlobalSymbolTotalNum (IntPtr interp) [private]`

Referenced by GlobalSymbolTotalNum().

12.23.3.48 `static String EmbeddedCh::ChInterp::_Ch_Home (IntPtr interp) [private]`

12.23.3.49 `static Int32 EmbeddedCh::ChInterp::_Ch_InitGlobalVar (IntPtr interp, Int32 flag) [private]`

Referenced by InitGlobalVar().

12.23.3.50 `static Int32 EmbeddedCh::ChInterp::_Ch_Initialize (ref IntPtr interp, IntPtr option) [private]`

12.23.3.51 `static Int32 EmbeddedCh::ChInterp::_Ch_Initialize (ref IntPtr interp, ref ChOptions_t option) [private]`

Referenced by Initialize().

12.23.3.52 `static Int32 EmbeddedCh::ChInterp::_Ch_IsFuncVarArg (IntPtr interp, String name) [private]`

Referenced by IsFuncVarArg().

12.23.3.53 `static Int32 EmbeddedCh::ChInterp::_Ch_ParseScript (IntPtr interp, String[] argv) [private]`

Referenced by ParseScript().

12.23.3.54 static Int32 EmbeddedCh::ChInterp::_Ch_Reopen (IntPtr *interp*, String *filename*, String *mode*, Int32 *fildev*) [private]

Referenced by Reopen().

12.23.3.55 static Int32 EmbeddedCh::ChInterp::_Ch_RunScript (IntPtr *interp*, String[] *argv*) [private]

Referenced by RunScript().

12.23.3.56 static Int32 EmbeddedCh::ChInterp::_Ch_RunScriptM (IntPtr *interp*, String[] *argv*) [private]

Referenced by RunScriptM().

12.23.3.57 static Int32 EmbeddedCh::ChInterp::_Ch_SetGlobalUserData (IntPtr *interp*, IntPtr *userdata*) [private]

Referenced by SetGlobalUserData().

12.23.3.58 static Int32 EmbeddedCh::ChInterp::_Ch_SetVar (IntPtr *interp*, String *name*, ChType *tatype*) [private]

Referenced by SetVar().

12.23.3.59 static Int32 EmbeddedCh::ChInterp::_Ch_StackLevel (IntPtr *interp*, ref Int32 *clevel*, ref Int32 *hlevel*) [private]

Referenced by StackLevel().

12.23.3.60 static String EmbeddedCh::ChInterp::_Ch_StackName (IntPtr *interp*, Int32 *level*, ref Int32 *isfunc*, String[] *classname*) [private]

Referenced by StackName().

12.23.3.61 static IntPtr EmbeddedCh::ChInterp::_Ch_SymbolAddrByIndex (IntPtr *interp*, Int32 *index*) [private]

Referenced by SymbolAddrByIndex().

12.23.3.62 static IntPtr EmbeddedCh::ChInterp::_Ch_SymbolAddrByName (IntPtr *interp*, String *name*) [private]

12.23.3.63 static IntPtr EmbeddedCh::ChInterp::_Ch_SymbolAddrByName (IntPtr *interp*, String *name*) [private]

Referenced by SymbolAddrByName().

12.23.3.64 `static Int32 EmbeddedCh::ChInterp::_Ch_SymbolIndexByName (IntPtr interp, String name) [private]`

Referenced by `SymbolIndexByName()`.

12.23.3.65 `static String EmbeddedCh::ChInterp::_Ch_SymbolNameByIndex (IntPtr interp, Int32 index) [private]`

Referenced by `SymbolNameByIndex()`.

12.23.3.66 `static Int32 EmbeddedCh::ChInterp::_Ch_SymbolTotalNum (IntPtr interp) [private]`

Referenced by `SymbolTotalNum()`.

12.23.3.67 `static Int32 EmbeddedCh::ChInterp::_Ch_UserDefinedInfo (IntPtr interp, IntPtr udtag, ref ChUserDefinedInfo_t udinfo) [private]`

Referenced by `UserDefinedInfo()`.

12.23.3.68 `static Int32 EmbeddedCh::ChInterp::_Ch_UserDefinedMemInfoByIndex (IntPtr interp, IntPtr udtag, Int32 index, ref ChMemInfo_t meminfo) [private]`

Referenced by `UserDefinedMemInfoByIndex()`.

12.23.3.69 `static Int32 EmbeddedCh::ChInterp::_Ch_UserDefinedMemInfoByName (IntPtr interp, IntPtr udtag, String memname, ref ChMemInfo_t meminfo) [private]`

Referenced by `UserDefinedMemInfoByName()`.

12.23.3.70 `static String EmbeddedCh::ChInterp::_Ch_UserDefinedName (IntPtr interp, String name) [private]`

Referenced by `UserDefinedName()`.

12.23.3.71 `static Int32 EmbeddedCh::ChInterp::_Ch_UserDefinedSize (IntPtr interp, String name) [private]`

Referenced by `UserDefinedSize()`.

12.23.3.72 `static IntPtr EmbeddedCh::ChInterp::_Ch_UserDefinedTag (IntPtr interp, String expr) [private]`

Referenced by `UserDefinedTag()`.

12.23.3.73 `static ChVarType_t EmbeddedCh::ChInterp::_Ch_VarType (IntPtr interp, String name) [private]`

Referenced by VarType().

12.23.3.74 `static Int32 EmbeddedCh::ChInterp::_Ch_Version (IntPtr interp, ChInfo_t * info) [private]`

12.23.3.75 `Int32 EmbeddedCh::ChInterp::Abort () [inline]`

Definition at line 183 of file ChInterp.cs.

References _Ch_Abort(), and interp.

12.23.3.76 `Int32 EmbeddedCh::ChInterp::AddCallback (Int32 _event, ChCallback callback, IntPtr clientdata, Int32 count) [inline]`

Definition at line 213 of file ChInterp.cs.

References _Ch_AddCallback(), and interp.

12.23.3.77 `int EmbeddedCh::ChInterp::AppendParseScript (String code) [inline]`

Definition at line 113 of file ChInterp.cs.

References _Ch_AppendParseScript(), and interp.

12.23.3.78 `int EmbeddedCh::ChInterp::AppendParseScriptFile (String filename) [inline]`

Definition at line 118 of file ChInterp.cs.

References _Ch_AppendParseScriptFile(), and interp.

12.23.3.79 `int EmbeddedCh::ChInterp::AppendRunScript (String argv) [inline]`

Definition at line 78 of file ChInterp.cs.

References _Ch_AppendRunScript(), and interp.

Referenced by Program1::Program::Main().

12.23.3.80 `int EmbeddedCh::ChInterp::AppendRunScriptFile (String filename) [inline]`

Definition at line 83 of file ChInterp.cs.

References _Ch_AppendRunScriptFile(), and interp.

Referenced by Program1::Program::Main().

12.23.3.81 `Int32 EmbeddedCh::ChInterp::ArrayDim (String expr) [inline]`

Definition at line 269 of file ChInterp.cs.

References _Ch_ArrayDim(), and interp.

12.23.3.82 Int32 EmbeddedCh::ChInterp::ArrayExtent (String *expr*, Int32 *index*) [inline]

Definition at line 274 of file ChInterp.cs.

References `_Ch_ArrayExtent()`, and `interp`.

12.23.3.83 Int32 EmbeddedCh::ChInterp::ArrayNum (String *expr*) [inline]

Definition at line 279 of file ChInterp.cs.

References `_Ch_ArrayNum()`, and `interp`.

12.23.3.84 ChType_t EmbeddedCh::ChInterp::ArrayType (String *expr*) [inline]

Definition at line 264 of file ChInterp.cs.

References `_Ch_ArrayType()`, and `interp`.

12.23.3.85 Int32 EmbeddedCh::ChInterp::ChangeStack (Int32 *level*, ref ChBlock_t *calldata*) [inline]

Definition at line 219 of file ChInterp.cs.

References `_Ch_ChangeStack()`, and `interp`.

12.23.3.86 delegate void EmbeddedCh::ChInterp::ChCallback (IntPtr *interp*, ref ChBlock_t *calldata*, IntPtr *clientdata*)**12.23.3.87 Int32 EmbeddedCh::ChInterp::Close (Int32 *filides*) [inline]**

Definition at line 168 of file ChInterp.cs.

References `_Ch_Close()`, and `interp`.

12.23.3.88 Int32 EmbeddedCh::ChInterp::DataSize (String *expr*) [inline]

Definition at line 259 of file ChInterp.cs.

References `_Ch_DataSize()`, and `interp`.

12.23.3.89 ChType_t EmbeddedCh::ChInterp::DataType (String *expr*) [inline]

Definition at line 254 of file ChInterp.cs.

References `_Ch_DataType()`, and `interp`.

12.23.3.90 Int32 EmbeddedCh::ChInterp::DeclareFunc (String *funcprototype*, IntPtr *funcptr*) [inline]

Definition at line 198 of file ChInterp.cs.

References `_Ch_DeclareFunc()`, and `interp`.

12.23.3.91 Int32 EmbeddedCh::ChInterp::DeclareTypedef (String *name*) [inline]

Definition at line 193 of file ChInterp.cs.

References `_Ch_DeclareTypedef()`, and `interp`.

12.23.3.92 Int32 EmbeddedCh::ChInterp::DeclareVar (String *declaration*) [inline]

Definition at line 188 of file ChInterp.cs.

References `_Ch_DeclareVar()`, and `interp`.

12.23.3.93 int EmbeddedCh::ChInterp::DeleteExprValue (IntPtr *vn*) [inline]

Definition at line 153 of file ChInterp.cs.

References `_Ch_DeleteExprValue()`, and `interp`.

12.23.3.94 int EmbeddedCh::ChInterp::End () [inline]

Definition at line 123 of file ChInterp.cs.

References `_Ch_End()`, and `interp`.

Referenced by `Program1::Program::Main()`.

12.23.3.95 int EmbeddedCh::ChInterp::ExecScript (String *programe*) [inline]

Definition at line 98 of file ChInterp.cs.

References `_Ch_ExecScript()`, and `interp`.

12.23.3.96 int EmbeddedCh::ChInterp::ExecScriptM (String *programe*) [inline]

Definition at line 103 of file ChInterp.cs.

References `_Ch_ExecScriptM()`, and `interp`.

12.23.3.97 int EmbeddedCh::ChInterp::ExprCalc (String *expr*, ChType_t *datatype*, IntPtr *result*) [inline]

Definition at line 131 of file ChInterp.cs.

References `_Ch_ExprCalc()`, and `interp`.

12.23.3.98 int EmbeddedCh::ChInterp::ExprEval (String *expr*) [inline]

Definition at line 136 of file ChInterp.cs.

References `_Ch_ExprEval()`, and `interp`.

12.23.3.99 int EmbeddedCh::ChInterp::ExprParse (String *expr*) [inline]

Definition at line 141 of file ChInterp.cs.

References `_Ch_ExprParse()`, and `interp`.

12.23.3.100 IntPtr EmbeddedCh::ChInterp::ExprValue (String *expr*, IntPtr *result*) [inline]

Definition at line 147 of file ChInterp.cs.

References `_Ch_ExprValue()`, and `interp`.

12.23.3.101 Int32 EmbeddedCh::ChInterp::Flush (Int32 *fildev*) [inline]

Definition at line 178 of file ChInterp.cs.

References `_Ch_Flush()`, and `interp`.

12.23.3.102 Int32 EmbeddedCh::ChInterp::FuncArgArrayDim (String *funcname*, Int32 *argnum*) [inline]

Definition at line 369 of file ChInterp.cs.

References `_Ch_FuncArgArrayDim()`, and `interp`.

12.23.3.103 Int32 EmbeddedCh::ChInterp::FuncArgArrayExtent (String *funcname*, Int32 *argnum*, Int32 *index*) [inline]

Definition at line 374 of file ChInterp.cs.

References `_Ch_FuncArgArrayExtent()`, and `interp`.

12.23.3.104 Int32 EmbeddedCh::ChInterp::FuncArgArrayNum (String *funcname*, Int32 *argnum*) [inline]

Definition at line 379 of file ChInterp.cs.

References `_Ch_FuncArgArrayNum()`, and `interp`.

12.23.3.105 ChType_t EmbeddedCh::ChInterp::FuncArgArrayType (String *funcname*, Int32 *argnum*) [inline]

Definition at line 364 of file ChInterp.cs.

References `_Ch_FuncArgArrayType()`, and `interp`.

12.23.3.106 ChType_t EmbeddedCh::ChInterp::FuncArgDataType (String *funcname*, Int32 *argnum*) [inline]

Definition at line 359 of file ChInterp.cs.

References `_Ch_FuncArgDataType()`, and `interp`.

12.23.3.107 Int32 EmbeddedCh::ChInterp::FuncArgFuncArgNum (String *funcname*, Int32 *argnum*) [inline]

Definition at line 394 of file ChInterp.cs.

References `_Ch_FuncArgFuncArgNum()`, and `interp`.

12.23.3.108 Int32 EmbeddedCh::ChInterp::FuncArgIsFunc (String *funcname*, Int32 *argnum*) [inline]

Definition at line 384 of file ChInterp.cs.

References `_Ch_FuncArgIsFunc()`, and `interp`.

12.23.3.109 Int32 EmbeddedCh::ChInterp::FuncArgIsFuncVarArg (String *funcname*, Int32 *argnum*) [inline]

Definition at line 389 of file ChInterp.cs.

References `_Ch_FuncArgIsFuncVarArg()`, and `interp`.

12.23.3.110 Int32 EmbeddedCh::ChInterp::FuncArgNum (String *name*) [inline]

Definition at line 299 of file ChInterp.cs.

References `_Ch_FuncArgNum()`, and `interp`.

12.23.3.111 String EmbeddedCh::ChInterp::FuncArgUserDefinedName (String *funcname*, Int32 *argnum*) [inline]

Definition at line 399 of file ChInterp.cs.

References `_Ch_FuncArgUserDefinedName()`, and `interp`.

12.23.3.112 Int32 EmbeddedCh::ChInterp::FuncArgUserDefinedSize (String *funcname*, Int32 *argnum*) [inline]

Definition at line 404 of file ChInterp.cs.

References `_Ch_FuncArgUserDefinedSize()`, and `interp`.

12.23.3.113 ChFuncType_t EmbeddedCh::ChInterp::FuncType (String *name*) [inline]

Definition at line 289 of file ChInterp.cs.

References `_Ch_FuncType()`, and `interp`.

12.23.3.114 IntPtr EmbeddedCh::ChInterp::GetGlobalUserData () [inline]

Definition at line 208 of file ChInterp.cs.

References `_Ch_GetGlobalUserData()`, and `interp`.

**12.23.3.115 IntPtr EmbeddedCh::ChInterp::GlobalSymbolAddrByIndex (Int32 *index*)
[inline]**

Definition at line 244 of file ChInterp.cs.

References `_Ch_GlobalSymbolAddrByIndex()`, and `interp`.

**12.23.3.116 Int32 EmbeddedCh::ChInterp::GlobalSymbolIndexByName (String *name*)
[inline]**

Definition at line 239 of file ChInterp.cs.

References `_Ch_GlobalSymbolIndexByName()`, and `interp`.

**12.23.3.117 String EmbeddedCh::ChInterp::GlobalSymbolNameByIndex (Int32 *index*)
[inline]**

Definition at line 249 of file ChInterp.cs.

References `_Ch_GlobalSymbolNameByIndex()`, and `interp`.

12.23.3.118 Int32 EmbeddedCh::ChInterp::GlobalSymbolTotalNum () [inline]

Definition at line 234 of file ChInterp.cs.

References `_Ch_GlobalSymbolTotalNum()`, and `interp`.

12.23.3.119 int EmbeddedCh::ChInterp::InitGlobalVar (Int32 *flag*) [inline]

Definition at line 88 of file ChInterp.cs.

References `_Ch_InitGlobalVar()`, and `interp`.

12.23.3.120 int EmbeddedCh::ChInterp::Initialize () [inline]

Definition at line 65 of file ChInterp.cs.

References `_Ch_Initialize()`, `interp`, and `useOptions`.

Referenced by `Program1::Program::Main()`.

12.23.3.121 Int32 EmbeddedCh::ChInterp::IsFuncVarArg (String *name*) [inline]

Definition at line 294 of file ChInterp.cs.

References `_Ch_IsFuncVarArg()`, and `interp`.

12.23.3.122 int EmbeddedCh::ChInterp::ParseScript (String[] *argv*) [inline]

Definition at line 93 of file ChInterp.cs.

References `_Ch_ParseScript()`, and `interp`.

**12.23.3.123 Int32 EmbeddedCh::ChInterp::Reopen (String *filename*, String *mode*, Int32 *filde*)
[inline]**

Definition at line 173 of file ChInterp.cs.

References `_Ch_Reopen()`, and `interp`.

12.23.3.124 int EmbeddedCh::ChInterp::RunScript (String[] *argv*) [inline]

Definition at line 73 of file ChInterp.cs.

References `_Ch_RunScript()`, and `interp`.

Referenced by `Program1::Program::Main()`.

12.23.3.125 int EmbeddedCh::ChInterp::RunScriptM (String[] *argv*) [inline]

Definition at line 108 of file ChInterp.cs.

References `_Ch_RunScriptM()`, and `interp`.

12.23.3.126 Int32 EmbeddedCh::ChInterp::SetGlobalUserData (IntPtr *userdata*) [inline]

Definition at line 203 of file ChInterp.cs.

References `_Ch_SetGlobalUserData()`, and `interp`.

12.23.3.127 Int32 EmbeddedCh::ChInterp::SetVar (String *name*, ChType_t *atype*) [inline]

Definition at line 159 of file ChInterp.cs.

References `_Ch_SetVar()`, and `interp`.

**12.23.3.128 Int32 EmbeddedCh::ChInterp::StackLevel (ref Int32 *clevel*, ref Int32 *hlevel*)
[inline]**

Definition at line 224 of file ChInterp.cs.

References `_Ch_StackLevel()`, and `interp`.

**12.23.3.129 String EmbeddedCh::ChInterp::StackName (Int32 *level*, ref Int32 *isfunc*, String[]
classname) [inline]**

Definition at line 229 of file ChInterp.cs.

References `_Ch_StackName()`, and `interp`.

12.23.3.130 IntPtr EmbeddedCh::ChInterp::SymbolAddrByIndex (Int32 *index*) [inline]

Definition at line 319 of file ChInterp.cs.

References `_Ch_SymbolAddrByIndex()`, and `interp`.

12.23.3.131 IntPtr EmbeddedCh::ChInterp::SymbolAddrByName (String *name*) [inline]

Definition at line 304 of file ChInterp.cs.

References `_Ch_SymbolAddrByName()`, and `interp`.

12.23.3.132 Int32 EmbeddedCh::ChInterp::SymbolIndexByName (String *name*) [inline]

Definition at line 314 of file ChInterp.cs.

References `_Ch_SymbolIndexByName()`, and `interp`.

12.23.3.133 String EmbeddedCh::ChInterp::SymbolNameByIndex (Int32 *index*) [inline]

Definition at line 324 of file ChInterp.cs.

References `_Ch_SymbolNameByIndex()`, and `interp`.

12.23.3.134 Int32 EmbeddedCh::ChInterp::SymbolTotalNum () [inline]

Definition at line 309 of file ChInterp.cs.

References `_Ch_SymbolTotalNum()`, and `interp`.

12.23.3.135 Int32 EmbeddedCh::ChInterp::UserDefinedInfo (IntPtr *udtag*, ref ChUserDefinedInfo_t *udinfo*) [inline]

Definition at line 334 of file ChInterp.cs.

References `_Ch_UserDefinedInfo()`, and `interp`.

12.23.3.136 Int32 EmbeddedCh::ChInterp::UserDefinedMemInfoByIndex (IntPtr *udtag*, Int32 *index*, ref ChMemInfo_t *meminfo*) [inline]

Definition at line 344 of file ChInterp.cs.

References `_Ch_UserDefinedMemInfoByIndex()`, and `interp`.

12.23.3.137 Int32 EmbeddedCh::ChInterp::UserDefinedMemInfoByName (IntPtr *udtag*, String *memname*, ref ChMemInfo_t *meminfo*) [inline]

Definition at line 339 of file ChInterp.cs.

References `_Ch_UserDefinedMemInfoByName()`, and `interp`.

12.23.3.138 String EmbeddedCh::ChInterp::UserDefinedName (String *name*) [inline]

Definition at line 349 of file ChInterp.cs.

References `_Ch_UserDefinedName()`, and `interp`.

12.23.3.139 Int32 EmbeddedCh::ChInterp::UserDefinedSize (String *name*) [inline]

Definition at line 354 of file ChInterp.cs.

References `_Ch_UserDefinedSize()`, and `interp`.

12.23.3.140 IntPtr EmbeddedCh::ChInterp::UserDefinedTag (String *expr*) [inline]

Definition at line 329 of file ChInterp.cs.

References `_Ch_UserDefinedTag()`, and `interp`.

12.23.3.141 ChVarType_t EmbeddedCh::ChInterp::VarType (String *name*) [inline]

Definition at line 284 of file ChInterp.cs.

References `_Ch_VarType()`, and `interp`.

12.23.4 Field Documentation**12.23.4.1 const String EmbeddedCh::ChInterp::chdll = "embedchdll.dll"**

Definition at line 13 of file ChInterp.cs.

12.23.4.2 IntPtr EmbeddedCh::ChInterp::interp = IntPtr.Zero [private]

Definition at line 19 of file ChInterp.cs.

Referenced by `Abort()`, `AddCallback()`, `AppendParseScript()`, `AppendParseScriptFile()`, `AppendRunScript()`, `AppendRunScriptFile()`, `ArrayDim()`, `ArrayExtent()`, `ArrayNum()`, `ArrayType()`, `ChangeStack()`, `Close()`, `DataSize()`, `DataType()`, `DeclareFunc()`, `DeclareTypedef()`, `DeclareVar()`, `DeleteExprValue()`, `End()`, `ExecScript()`, `ExecScriptM()`, `ExprCalc()`, `ExprEval()`, `ExprParse()`, `ExprValue()`, `Flush()`, `FuncArgArrayDim()`, `FuncArgArrayExtent()`, `FuncArgArrayNum()`, `FuncArgArrayType()`, `FuncArgDataType()`, `FuncArgFuncArgNum()`, `FuncArgIsFunc()`, `FuncArgIsFuncVarArg()`, `FuncArgNum()`, `FuncArgUserDefinedName()`, `FuncArgUserDefinedSize()`, `FuncType()`, `GetGlobalUserData()`, `GlobalSymbolAddrByIndex()`, `GlobalSymbolIndexByName()`, `GlobalSymbolNameByIndex()`, `GlobalSymbolTotalNum()`, `InitGlobalVar()`, `Initialize()`, `IsFuncVarArg()`, `ParseScript()`, `Reopen()`, `RunScript()`, `RunScriptM()`, `SetGlobalUserData()`, `SetVar()`, `StackLevel()`, `StackName()`, `SymbolAddrByIndex()`, `SymbolAddrByName()`, `SymbolIndexByName()`, `SymbolNameByIndex()`, `SymbolTotalNum()`, `UserDefinedInfo()`, `UserDefinedMemInfoByIndex()`, `UserDefinedMemInfoByName()`, `UserDefinedName()`, `UserDefinedSize()`, `UserDefinedTag()`, and `VarType()`.

12.23.4.3 ChOptions_t EmbeddedCh::ChInterp::options = new ChOptions_t() [private]

Definition at line 22 of file ChInterp.cs.

12.23.4.4 bool EmbeddedCh::ChInterp::useOptions = false [private]

Definition at line 23 of file ChInterp.cs.

Referenced by `Initialize()`.

12.23.5 Property Documentation

12.23.5.1 String EmbeddedCh::ChInterp::ChHome [get, set]

Definition at line 54 of file ChInterp.cs.

12.23.5.2 ChShellType EmbeddedCh::ChInterp::ShellType [get, set]

Definition at line 42 of file ChInterp.cs.

12.23.5.3 bool EmbeddedCh::ChInterp::UseOptions [get, set]

Definition at line 30 of file ChInterp.cs.

The documentation for this class was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/[ChInterp.cs](#)

12.24 EmbeddedCh::ChMemInfo_t Struct Reference

Data Fields

- [int](#) `index`
- [String](#) `memname`
- [int](#) `offset`
- [ChType_t](#) `dtype`
- [int](#) `ispublic`
- [int](#) `isfunc`
- [int](#) `ismemberfunc`
- [int](#) `isconstructor`
- [int](#) `isdestructor`
- [int](#) `isvararg`
- [int](#) `arraytype`
- [int](#) `dim`
- [int\[\]](#) `extent`
- [int](#) `isbitfield`
- [int](#) `fieldsize`
- [int](#) `fieldoffset`
- [IntPtr](#) `udtag`

12.24.1 Detailed Description

Definition at line 72 of file EmbedCh.cs.

12.24.2 Field Documentation

12.24.2.1 `int EmbeddedCh::ChMemInfo_t::arraytype`

Definition at line 83 of file EmbedCh.cs.

12.24.2.2 `int EmbeddedCh::ChMemInfo_t::dim`

Definition at line 84 of file EmbedCh.cs.

12.24.2.3 `ChType_t EmbeddedCh::ChMemInfo_t::dtype`

Definition at line 76 of file EmbedCh.cs.

12.24.2.4 `int [] EmbeddedCh::ChMemInfo_t::extent`

Definition at line 86 of file EmbedCh.cs.

12.24.2.5 `int EmbeddedCh::ChMemInfo_t::fieldoffset`

Definition at line 89 of file EmbedCh.cs.

12.24.2.6 int EmbeddedCh::ChMemInfo_t::fieldsize

Definition at line 88 of file EmbedCh.cs.

12.24.2.7 int EmbeddedCh::ChMemInfo_t::index

Definition at line 73 of file EmbedCh.cs.

12.24.2.8 int EmbeddedCh::ChMemInfo_t::isbitfield

Definition at line 87 of file EmbedCh.cs.

12.24.2.9 int EmbeddedCh::ChMemInfo_t::isconstructor

Definition at line 80 of file EmbedCh.cs.

12.24.2.10 int EmbeddedCh::ChMemInfo_t::isdestructor

Definition at line 81 of file EmbedCh.cs.

12.24.2.11 int EmbeddedCh::ChMemInfo_t::isfunc

Definition at line 78 of file EmbedCh.cs.

12.24.2.12 int EmbeddedCh::ChMemInfo_t::ismemberfunc

Definition at line 79 of file EmbedCh.cs.

12.24.2.13 int EmbeddedCh::ChMemInfo_t::ispublic

Definition at line 77 of file EmbedCh.cs.

12.24.2.14 int EmbeddedCh::ChMemInfo_t::isvararg

Definition at line 82 of file EmbedCh.cs.

12.24.2.15 String EmbeddedCh::ChMemInfo_t::memname

Definition at line 74 of file EmbedCh.cs.

12.24.2.16 int EmbeddedCh::ChMemInfo_t::offset

Definition at line 75 of file EmbedCh.cs.

12.24.2.17 IntPtr EmbeddedCh::ChMemInfo_t::udtag

Definition at line 92 of file EmbedCh.cs.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/EmbedCh.cs](#)

12.25 EmbeddedCh::ChOptions_t Struct Reference

Data Fields

- [int shelltype](#)
- [String chhome](#)

12.25.1 Detailed Description

Definition at line 9 of file EmbedCh.cs.

12.25.2 Field Documentation

12.25.2.1 String EmbeddedCh::ChOptions_t::chhome

Embedded Ch home directory. if NULL, use Ch home directory for standard/professional edition as Embedded Ch home directory

Definition at line 16 of file EmbedCh.cs.

12.25.2.2 int EmbeddedCh::ChOptions_t::shelltype

shell type: CH_REGULARCH or CH_SAFECH

Definition at line 11 of file EmbedCh.cs.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/EmbedCh.cs](#)

12.26 LibMC::MCAgency::ChOptions_t Struct Reference

ChOptions structures.

Data Fields

- [int shelltype](#)
- [String chhome](#)

12.26.1 Detailed Description

ChOptions structures. Allows the user to set the shell type and the home directory.

Note:

This struct is pulled directly from the Mobile-C library.

Definition at line 61 of file MCEExports.cs.

12.26.2 Field Documentation

12.26.2.1 String LibMC::MCAgency::ChOptions_t::chhome

Embedded Ch home directory. if NULL, use Ch home directory for standard/professional edition as Embedded Ch home directory

Definition at line 66 of file MCEExports.cs.

Referenced by LibMC::MCAgency::ChInitializeOptions().

12.26.2.2 int LibMC::MCAgency::ChOptions_t::shelltype

shell type: CH_REGULARCH or CH_SAFECH

Definition at line 63 of file MCEExports.cs.

Referenced by LibMC::MCAgency::ChInitializeOptions().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs](#)

12.27 EmbeddedCh::ChUserDefinedInfo_t Struct Reference

Data Fields

- [ChType_t dtype](#)
- [String tagname](#)
- [int size](#)
- [int totnum](#)

12.27.1 Detailed Description

Definition at line 64 of file EmbedCh.cs.

12.27.2 Field Documentation

12.27.2.1 ChType_t EmbeddedCh::ChUserDefinedInfo_t::dtype

Definition at line 65 of file EmbedCh.cs.

12.27.2.2 int EmbeddedCh::ChUserDefinedInfo_t::size

Definition at line 67 of file EmbedCh.cs.

12.27.2.3 String EmbeddedCh::ChUserDefinedInfo_t::tagname

Definition at line 66 of file EmbedCh.cs.

12.27.2.4 int EmbeddedCh::ChUserDefinedInfo_t::totnum

Definition at line 68 of file EmbedCh.cs.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/EmbedCh.cs](#)

12.28 EmbeddedCh::ChUserDefinedTag Class Reference

Public Member Functions

- [ChUserDefinedTag \(\)](#)

Properties

- [IntPtr Tag](#) [get, set]

Private Attributes

- [IntPtr ptr](#)

12.28.1 Detailed Description

Definition at line 8 of file ChUserDefinedTag.cs.

12.28.2 Constructor & Destructor Documentation

12.28.2.1 EmbeddedCh::ChUserDefinedTag::ChUserDefinedTag () [inline]

Definition at line 12 of file ChUserDefinedTag.cs.

References [ptr](#).

12.28.3 Field Documentation

12.28.3.1 IntPtr EmbeddedCh::ChUserDefinedTag::ptr [private]

Definition at line 10 of file ChUserDefinedTag.cs.

Referenced by [ChUserDefinedTag\(\)](#).

12.28.4 Property Documentation

12.28.4.1 IntPtr EmbeddedCh::ChUserDefinedTag::Tag [get, set]

Definition at line 18 of file ChUserDefinedTag.cs.

The documentation for this class was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/ChUserDefinedTag.cs](#)

12.29 EmbeddedCh::ChVaList Class Reference

Public Member Functions

- [ChVaList \(\)](#)

Private Attributes

- IntPtr [ptr](#) = IntPtr.Zero

12.29.1 Detailed Description

Definition at line 8 of file ChVaList.cs.

12.29.2 Constructor & Destructor Documentation

12.29.2.1 EmbeddedCh::ChVaList::ChVaList () [inline]

Definition at line 12 of file ChVaList.cs.

12.29.3 Field Documentation

12.29.3.1 IntPtr EmbeddedCh::ChVaList::ptr = IntPtr.Zero [private]

Definition at line 10 of file ChVaList.cs.

The documentation for this class was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/[ChVaList.cs](#)

12.30 cmd_prompt_s Struct Reference

```
#include <cmd_prompt.h>
```

Data Fields

- `THREAD_T` [thread](#)

12.30.1 Detailed Description

Definition at line 41 of file `cmd_prompt.h`.

12.30.2 Field Documentation

12.30.2.1 `THREAD_T cmd_prompt_s::thread`

Definition at line 42 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Start()`, and `MC_End()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/cmd_prompt.h`

12.31 `command_s` Struct Reference

```
#include <cmd_prompt.h>
```

Data Fields

- [int index](#)
- [int num_args](#)
- [char ** args](#)

12.31.1 Detailed Description

Definition at line 46 of file `cmd_prompt.h`.

12.31.2 Field Documentation

12.31.2.1 `char** command_s::args`

Definition at line 49 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Thread()`, `dealloc_command()`, `exec_command()`, `handler_COMPOSE_SEND()`, `handler_HELP()`, `handler_SEND()`, and `process_command()`.

12.31.2.2 `int command_s::index`

Definition at line 47 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Thread()`, `exec_command()`, and `process_command()`.

12.31.2.3 `int command_s::num_args`

Definition at line 48 of file `cmd_prompt.h`.

Referenced by `cmd_prompt_Thread()`, `dealloc_command()`, `exec_command()`, `handler_COMPOSE_SEND()`, `handler_HELP()`, `handler_SEND()`, and `process_command()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/cmd_prompt.h`

12.32 connection_s Struct Reference

```
#include <connection.h>
```

Data Fields

- [int connect_id](#)
- [int nonce](#)
- [char * remote_hostname](#)
- [struct sockaddr_in addr](#)
- [u_long clientfd](#)
- [u_long serverfd](#)
- [unsigned char * AES_key](#)

12.32.1 Detailed Description

Definition at line 46 of file connection.h.

12.32.2 Field Documentation

12.32.2.1 struct sockaddr_in connection_s::addr [read]

Definition at line 56 of file connection.h.

Referenced by connection_Copy(), listen_Thread(), and message_InitializeFromConnection().

12.32.2.2 unsigned char* connection_s::AES_key

Definition at line 63 of file connection.h.

Referenced by listen_Thread(), and rece_de_msg().

12.32.2.3 u_long connection_s::clientfd

Definition at line 59 of file connection.h.

Referenced by connection_Copy(), connection_Destroy(), listen_Thread(), message_InitializeFromConnection(), mtp_http_InitializeFromConnection(), and rece_de_msg().

12.32.2.4 int connection_s::connect_id

Definition at line 49 of file connection.h.

Referenced by AP_QUEUE_SEARCH_TEMPLATE(), connection_Copy(), listen_Thread(), and message_InitializeFromConnection().

12.32.2.5 int connection_s::nonce

Definition at line 51 of file connection.h.

Referenced by listen_Thread(), and rece_de_msg().

12.32.2.6 char* connection_s::remote_hostname

Definition at line 54 of file connection.h.

Referenced by AP_QUEUE_SEARCH_TEMPLATE(), connection_Copy(), connection_Destroy(), and listen_Thread().

12.32.2.7 u_long connection_s::serverfd

Definition at line 60 of file connection.h.

Referenced by connection_Copy(), and listen_Thread().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[connection.h](#)

12.33 fipa_expression_s::content_u Union Reference

```
#include <fipa_acl.h>
```

Data Fields

- struct [fipa_word_s](#) * word
- struct [fipa_string_s](#) * string
- struct [fipa_number_s](#) * number
- struct [fipa_DateTime_s](#) * datetime
- struct [fipa_expression_s](#) ** expression

12.33.1 Detailed Description

Definition at line 196 of file [fipa_acl.h](#).

12.33.2 Field Documentation

12.33.2.1 struct [fipa_DateTime_s](#)* [fipa_expression_s::content_u::datetime](#) [read]

Definition at line 201 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

12.33.2.2 struct [fipa_expression_s](#)** [fipa_expression_s::content_u::expression](#) [read]

Definition at line 202 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

12.33.2.3 struct [fipa_number_s](#)* [fipa_expression_s::content_u::number](#) [read]

Definition at line 200 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

12.33.2.4 struct [fipa_string_s](#)* [fipa_expression_s::content_u::string](#) [read]

Definition at line 199 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), [fipa_expression_Destroy\(\)](#), and [MC_AclSetConversationID\(\)](#).

12.33.2.5 struct [fipa_word_s](#)* [fipa_expression_s::content_u::word](#) [read]

Definition at line 198 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), and [fipa_expression_Destroy\(\)](#).

The documentation for this union was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.34 des3_context Struct Reference

Triple-DES context structure.

```
#include <des.h>
```

Data Fields

- [int mode](#)
- unsigned long [sk](#) [96]

12.34.1 Detailed Description

Triple-DES context structure.

Definition at line 23 of file des.h.

12.34.2 Field Documentation

12.34.2.1 int des3_context::mode

encrypt/decrypt

Definition at line 25 of file des.h.

12.34.2.2 unsigned long des3_context::sk[96]

3DES subkeys

Definition at line 26 of file des.h.

Referenced by `des3_crypt_ecb()`, `des3_set2key_dec()`, `des3_set2key_enc()`, `des3_set3key_dec()`, and `des3_set3key_enc()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/des.h`

12.35 `des_context` Struct Reference

DES context structure.

```
#include <des.h>
```

Data Fields

- `int mode`
- unsigned long `sk` [32]

12.35.1 Detailed Description

DES context structure.

Definition at line 13 of file `des.h`.

12.35.2 Field Documentation

12.35.2.1 `int des_context::mode`

encrypt/decrypt

Definition at line 15 of file `des.h`.

12.35.2.2 `unsigned long des_context::sk[32]`

DES subkeys

Definition at line 16 of file `des.h`.

Referenced by `des_crypt_ecb()`, `des_setkey_dec()`, and `des_setkey_enc()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/des.h`

12.36 dhm_context Struct Reference

```
#include <dhm.h>
```

Data Fields

- [int len](#)
- [mpi P](#)
- [mpi G](#)
- [mpi X](#)
- [mpi GX](#)
- [mpi GY](#)
- [mpi K](#)
- [mpi RP](#)

12.36.1 Detailed Description

Definition at line 16 of file dhm.h.

12.36.2 Field Documentation

12.36.2.1 mpi dhm_context::G

generator

Definition at line 20 of file dhm.h.

Referenced by dhm_free(), dhm_make_params(), dhm_make_public(), dhm_read_params(), main(), ssl_parse_server_key_exchange(), ssl_set_dh_param(), and ssl_write_server_key_exchange().

12.36.2.2 mpi dhm_context::GX

$\text{self} = G^X \bmod P$

Definition at line 22 of file dhm.h.

Referenced by dhm_free(), dhm_make_params(), dhm_make_public(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

12.36.2.3 mpi dhm_context::GY

$\text{peer} = G^Y \bmod P$

Definition at line 23 of file dhm.h.

Referenced by dhm_calc_secret(), dhm_free(), dhm_read_params(), dhm_read_public(), ssl_parse_client_key_exchange(), and ssl_parse_server_key_exchange().

12.36.2.4 `mpi dhm_context::K`

$\text{key} = \text{GY}^X \bmod P$

Definition at line 24 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_free()`, `ssl_parse_client_key_exchange()`, and `ssl_write_client_key_exchange()`.

12.36.2.5 `int dhm_context::len`

size(P) in chars

Definition at line 18 of file dhm.h.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `dhm_read_params()`, `dhm_read_public()`, `main()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_key_exchange()`, and `ssl_write_client_key_exchange()`.

12.36.2.6 `mpi dhm_context::P`

prime modulus

Definition at line 19 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_free()`, `dhm_make_params()`, `dhm_make_public()`, `dhm_read_params()`, `main()`, `ssl_parse_server_key_exchange()`, `ssl_set_dh_param()`, and `ssl_write_server_key_exchange()`.

12.36.2.7 `mpi dhm_context::RP`

cached $R^2 \bmod P$

Definition at line 25 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_free()`, `dhm_make_params()`, and `dhm_make_public()`.

12.36.2.8 `mpi dhm_context::X`

secret value

Definition at line 21 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_free()`, `dhm_make_params()`, `dhm_make_public()`, `ssl_write_client_key_exchange()`, and `ssl_write_server_key_exchange()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/dhm.h`

12.37 dynstring_s Struct Reference

```
#include <dynstring.h>
```

Data Fields

- [int len](#)
- [int size](#)
- [char * message](#)

12.37.1 Detailed Description

Definition at line 9 of file dynstring.h.

12.37.2 Field Documentation

12.37.2.1 int dynstring_s::len

Definition at line 10 of file dynstring.h.

Referenced by `dynstring_Append()`, `dynstring_New()`, and `mtp_http_CreateMessage()`.

12.37.2.2 char* dynstring_s::message

Definition at line 12 of file dynstring.h.

Referenced by `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `fipa_envelope_Compose__from()`, `MC_AclSend()`, `message_send_Thread()`, and `mtp_http_CreateMessage()`.

12.37.2.3 int dynstring_s::size

Definition at line 11 of file dynstring.h.

Referenced by `dynstring_Append()`, and `dynstring_New()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/dynstring.h](#)

12.38 fipa_acl_envelope_Received_s Struct Reference

```
#include <fipa_acl_envelope.h>
```

Data Fields

- struct [fipa_url_s](#) * [received_by](#)
- struct [fipa_url_s](#) * [received_from](#)
- struct [fipa_DateTime_s](#) * [received_date](#)
- char * [received_id](#)
- struct [fipa_url_s](#) * [received_via](#)

12.38.1 Detailed Description

Definition at line 7 of file `fipa_acl_envelope.h`.

12.38.2 Field Documentation

12.38.2.1 struct `fipa_url_s`* `fipa_acl_envelope_Received_s::received_by` [read]

Definition at line 9 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

12.38.2.2 struct `fipa_DateTime_s`* `fipa_acl_envelope_Received_s::received_date` [read]

Definition at line 11 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

12.38.2.3 struct `fipa_url_s`* `fipa_acl_envelope_Received_s::received_from` [read]

Definition at line 10 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

12.38.2.4 char* `fipa_acl_envelope_Received_s::received_id`

Definition at line 12 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

12.38.2.5 struct `fipa_url_s`* `fipa_acl_envelope_Received_s::received_via` [read]

Definition at line 13 of file `fipa_acl_envelope.h`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_envelope_Received_Destroy()`, and `fipa_envelope_HandleReceived()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl_envelope.h](#)

12.39 fipa_acl_envelope_s Struct Reference

```
#include <fipa_acl_envelope.h>
```

Data Fields

- [int num_params](#)
- [struct fipa_acl_Param_s ** params](#)

12.39.1 Detailed Description

Definition at line 40 of file `fipa_acl_envelope.h`.

12.39.2 Field Documentation

12.39.2.1 `int fipa_acl_envelope_s::num_params`

Definition at line 42 of file `fipa_acl_envelope.h`.

Referenced by `acc_connection_Thread()`, `fipa_acl_envelope_Copy()`, `fipa_acl_envelope_Destroy()`, and `fipa_envelope_HandleParams()`.

12.39.2.2 `struct fipa_acl_Param_s** fipa_acl_envelope_s::params` **[read]**

Definition at line 43 of file `fipa_acl_envelope.h`.

Referenced by `acc_connection_Thread()`, `fipa_acl_envelope_Copy()`, `fipa_acl_envelope_Destroy()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleParams()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_HandleReceived()`, and `fipa_envelope_HandleTo()`.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl_envelope.h](#)

12.40 fipa_acl_message_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- enum [fipa_performative_e](#) performative
- struct [fipa_agent_identifier_s](#) * sender
- struct [fipa_agent_identifier_set_s](#) * receiver
- int receiver_num
- struct [fipa_agent_identifier_set_s](#) * reply_to
- struct [fipa_string_s](#) * content
- struct [fipa_expression_s](#) * language
- struct [fipa_expression_s](#) * encoding
- struct [fipa_expression_s](#) * ontology
- enum [fipa_protocol_e](#) protocol
- struct [fipa_expression_s](#) * conversation_id
- struct [fipa_expression_s](#) * reply_with
- struct [fipa_expression_s](#) * in_reply_to
- struct [fipa_DateTime_s](#) * reply_by

12.40.1 Detailed Description

Definition at line 105 of file [fipa_acl.h](#).

12.40.2 Field Documentation

12.40.2.1 struct [fipa_string_s](#)* [fipa_acl_message_s::content](#) [read]

Definition at line 117 of file [fipa_acl.h](#).

Referenced by [fipa_acl_Compose\(\)](#), [fipa_acl_message_Copy\(\)](#), [fipa_acl_message_Destroy\(\)](#), [fipa_message_parameter_Parse\(\)](#), [MC_AclSetContent\(\)](#), and [MC_AclSetContent_chdl\(\)](#).

12.40.2.2 struct [fipa_expression_s](#)* [fipa_acl_message_s::conversation_id](#) [read]

Definition at line 128 of file [fipa_acl.h](#).

Referenced by [fipa_acl_Compose\(\)](#), [fipa_acl_message_Copy\(\)](#), [fipa_acl_message_Destroy\(\)](#), [fipa_message_parameter_Parse\(\)](#), [fipa_Reply\(\)](#), and [MC_AclSetConversationID\(\)](#).

12.40.2.3 struct [fipa_expression_s](#)* [fipa_acl_message_s::encoding](#) [read]

Definition at line 121 of file [fipa_acl.h](#).

Referenced by [fipa_acl_Compose\(\)](#), [fipa_acl_message_Copy\(\)](#), [fipa_acl_message_Destroy\(\)](#), and [fipa_message_parameter_Parse\(\)](#).

12.40.2.4 struct fipa_expression_s* fipa_acl_message_s::in_reply_to [read]

Definition at line 132 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

12.40.2.5 struct fipa_expression_s* fipa_acl_message_s::language [read]

Definition at line 119 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

12.40.2.6 struct fipa_expression_s* fipa_acl_message_s::ontology [read]

Definition at line 123 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

12.40.2.7 enum fipa_performative_e fipa_acl_message_s::performative

Definition at line 107 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_Parse(), MC_AclSetPerformative(), and MC_AclSetPerformative_chdl().

12.40.2.8 enum fipa_protocol_e fipa_acl_message_s::protocol

Definition at line 126 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_message_parameter_Parse(), fipa_Reply(), MC_AclSetProtocol(), and MC_AclSetProtocol_chdl().

12.40.2.9 struct fipa_agent_identifier_set_s* fipa_acl_message_s::receiver [read]

Definition at line 111 of file fipa_acl.h.

Referenced by AP_QUEUE_STD_DEFN_TEMPLATE(), fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), fipa_envelope_Compose__intended_receiver(), fipa_envelope_Compose__to(), fipa_message_parameter_Parse(), fipa_Reply(), MC_AclAddReceiver(), and MC_AclSend().

12.40.2.10 int fipa_acl_message_s::receiver_num

Definition at line 112 of file fipa_acl.h.

Referenced by MC_AclAddReceiver().

12.40.2.11 struct fipa_DateTime_s* fipa_acl_message_s::reply_by [read]

Definition at line 134 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

12.40.2.12 struct fipa_agent_identifier_set_s* fipa_acl_message_s::reply_to [read]

Definition at line 114 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), fipa_message_parameter_Parse(), fipa_Reply(), and MC_AclAddReplyTo().

12.40.2.13 struct fipa_expression_s* fipa_acl_message_s::reply_with [read]

Definition at line 130 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), and fipa_message_parameter_Parse().

12.40.2.14 struct fipa_agent_identifier_s* fipa_acl_message_s::sender [read]

Definition at line 109 of file fipa_acl.h.

Referenced by fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_acl_message_Destroy(), fipa_envelope_Compose__from(), fipa_message_parameter_Parse(), fipa_Reply(), and MC_AclSetSender().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.41 fipa_acl_Param_s Struct Reference

```
#include <fipa_acl_envelope.h>
```

Data Fields

- struct [fipa_agent_identifier_set_s](#) * to
- struct [fipa_agent_identifier_s](#) * from
- char * [comments](#)
- char * [acl_representation](#)
- char * [payload_length](#)
- char * [payload_encoding](#)
- struct [fipa_DateTime_s](#) * date
- struct [fipa_agent_identifier_set_s](#) * intended_receiver
- struct [fipa_acl_envelope_Received_s](#) * received

12.41.1 Detailed Description

Definition at line 22 of file [fipa_acl_envelope.h](#).

12.41.2 Field Documentation

12.41.2.1 char* fipa_acl_Param_s::acl_representation

Definition at line 27 of file [fipa_acl_envelope.h](#).

Referenced by [fipa_acl_Param_Copy\(\)](#), [fipa_acl_Param_Destroy\(\)](#), and [fipa_envelope_HandleAclRepresentation\(\)](#).

12.41.2.2 char* fipa_acl_Param_s::comments

Definition at line 26 of file [fipa_acl_envelope.h](#).

Referenced by [fipa_acl_Param_Copy\(\)](#), [fipa_acl_Param_Destroy\(\)](#), and [fipa_envelope_HandleComments\(\)](#).

12.41.2.3 struct fipa_DateTime_s* fipa_acl_Param_s::date [read]

Definition at line 30 of file [fipa_acl_envelope.h](#).

Referenced by [fipa_acl_Param_Copy\(\)](#), [fipa_acl_Param_Destroy\(\)](#), and [fipa_envelope_HandleDate\(\)](#).

12.41.2.4 struct fipa_agent_identifier_s* fipa_acl_Param_s::from [read]

Definition at line 25 of file [fipa_acl_envelope.h](#).

Referenced by [fipa_acl_Param_Copy\(\)](#), [fipa_acl_Param_Destroy\(\)](#), and [fipa_envelope_HandleFrom\(\)](#).

12.41.2.5 struct fipa_agent_identifier_set_s* fipa_acl_Param_s::intended_receiver [read]

Definition at line 31 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandleIntendedReceiver().

12.41.2.6 char* fipa_acl_Param_s::payload_encoding

Definition at line 29 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandlePayloadEncoding().

12.41.2.7 char* fipa_acl_Param_s::payload_length

Definition at line 28 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandlePayloadLength().

12.41.2.8 struct fipa_acl_envelope_Received_s* fipa_acl_Param_s::received [read]

Definition at line 32 of file fipa_acl_envelope.h.

Referenced by fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandleReceived().

12.41.2.9 struct fipa_agent_identifier_set_s* fipa_acl_Param_s::to [read]

Definition at line 24 of file fipa_acl_envelope.h.

Referenced by acc_connection_Thread(), fipa_acl_Param_Copy(), fipa_acl_Param_Destroy(), and fipa_envelope_HandleTo().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/fipa_acl_envelope.h

12.42 fipa_agent_identifier_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- char * [name](#)
- struct [fipa_url_sequence_s](#) * [addresses](#)
- struct [fipa_agent_identifier_set_s](#) * [resolvers](#)

12.42.1 Detailed Description

Definition at line 180 of file [fipa_acl.h](#).

12.42.2 Field Documentation

12.42.2.1 struct [fipa_url_sequence_s](#)* [fipa_agent_identifier_s::addresses](#) [read]

Definition at line 183 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [fipa_agent_identifier_Compose\(\)](#), [fipa_agent_identifier_Copy\(\)](#), [fipa_agent_identifier_Destroy\(\)](#), [fipa_envelope_Compose__from\(\)](#), [fipa_envelope_Compose__intended_receiver\(\)](#), [fipa_envelope_Compose__to\(\)](#), [fipa_envelope_ParseAddresses\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), [MC_AclSend\(\)](#), and [MC_AclSetSender\(\)](#).

12.42.2.2 char* [fipa_agent_identifier_s::name](#)

Definition at line 182 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [AP_QUEUE_STD_DEFN_TEMPLATE\(\)](#), [fipa_agent_identifier_Compose\(\)](#), [fipa_agent_identifier_Copy\(\)](#), [fipa_agent_identifier_Destroy\(\)](#), [fipa_envelope_Compose__from\(\)](#), [fipa_envelope_Compose__intended_receiver\(\)](#), [fipa_envelope_Compose__to\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), [MC_AclSend\(\)](#), and [MC_AclSetSender\(\)](#).

12.42.2.3 struct [fipa_agent_identifier_set_s](#)* [fipa_agent_identifier_s::resolvers](#) [read]

Definition at line 184 of file [fipa_acl.h](#).

Referenced by [fipa_agent_identifier_Compose\(\)](#), [fipa_agent_identifier_Copy\(\)](#), [fipa_agent_identifier_Destroy\(\)](#), and [fipa_envelope_ParseResolvers\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.43 fipa_agent_identifier_set_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- [int num](#)
- [int retain_order](#)
- struct [fipa_agent_identifier_s](#) ** [fipa_agent_identifiers](#)

12.43.1 Detailed Description

Definition at line 166 of file [fipa_acl.h](#).

12.43.2 Field Documentation

12.43.2.1 struct [fipa_agent_identifier_s](#)** [fipa_agent_identifier_set_s::fipa_agent_identifiers](#) [read]

Definition at line 170 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [AP_QUEUE_STD_DEFN_TEMPLATE\(\)](#), [fipa_agent_identifier_set_Compose\(\)](#), [fipa_agent_identifier_set_Copy\(\)](#), [fipa_agent_identifier_set_Destroy\(\)](#), [fipa_envelope_Compose__intended_receiver\(\)](#), [fipa_envelope_Compose__to\(\)](#), [fipa_envelope_HandleIntendedReceiver\(\)](#), [fipa_envelope_HandleTo\(\)](#), [fipa_envelope_ParseResolvers\(\)](#), [fipa_Reply\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), and [MC_AclSend\(\)](#).

12.43.2.2 int [fipa_agent_identifier_set_s::num](#)

Definition at line 168 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [fipa_agent_identifier_Compose\(\)](#), [fipa_agent_identifier_set_Compose\(\)](#), [fipa_agent_identifier_set_Copy\(\)](#), [fipa_agent_identifier_set_Destroy\(\)](#), [fipa_envelope_Compose__to\(\)](#), [fipa_envelope_HandleIntendedReceiver\(\)](#), [fipa_envelope_HandleTo\(\)](#), [fipa_envelope_ParseResolvers\(\)](#), [fipa_Reply\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), and [MC_AclSend\(\)](#).

12.43.2.3 int [fipa_agent_identifier_set_s::retain_order](#)

Definition at line 169 of file [fipa_acl.h](#).

Referenced by [fipa_agent_identifier_set_Copy\(\)](#), [fipa_agent_identifier_set_Parse\(\)](#), [fipa_envelope_HandleIntendedReceiver\(\)](#), [fipa_envelope_HandleTo\(\)](#), [fipa_envelope_ParseResolvers\(\)](#), and [fipa_Reply\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.44 fipa_comm_action_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- enum [fipa_comm_action_type_e](#) type
- [fipa_comm_callback](#) callback
- [fipa_comm_message_check_p](#) check_list
- [fipa_comm_reply_p](#) reply
- struct [fipa_comm_action_t](#) * next

12.44.1 Detailed Description

Definition at line 99 of file [fipa_comm.h](#).

12.44.2 Field Documentation

12.44.2.1 fipa_comm_callback fipa_comm_action_t::callback

Definition at line 102 of file [fipa_comm.h](#).

12.44.2.2 fipa_comm_message_check_p fipa_comm_action_t::check_list

Definition at line 103 of file [fipa_comm.h](#).

12.44.2.3 struct fipa_comm_action_t* fipa_comm_action_t::next [read]

Definition at line 105 of file [fipa_comm.h](#).

12.44.2.4 fipa_comm_reply_p fipa_comm_action_t::reply

Definition at line 104 of file [fipa_comm.h](#).

12.44.2.5 enum fipa_comm_action_type_e fipa_comm_action_t::type

Definition at line 101 of file [fipa_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.45 fipa_comm_message_check_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- char * [macro](#)
- enum [fipa_comm_check_type_e](#) type
- struct [fipa_comm_message_check_t](#) * next

12.45.1 Detailed Description

Definition at line 72 of file fipa_comm.h.

12.45.2 Field Documentation

12.45.2.1 char* fipa_comm_message_check_t::macro

Definition at line 74 of file fipa_comm.h.

12.45.2.2 struct fipa_comm_message_check_t* fipa_comm_message_check_t::next [read]

Definition at line 76 of file fipa_comm.h.

12.45.2.3 enum fipa_comm_check_type_e fipa_comm_message_check_t::type

Definition at line 75 of file fipa_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.46 fipa_comm_performative_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- [fipa_comm_action_p action_list](#)
- [fipa_comm_reply_p default_reply](#)

12.46.1 Detailed Description

Definition at line 115 of file fipa_comm.h.

12.46.2 Field Documentation

12.46.2.1 fipa_comm_action_p fipa_comm_performative_t::action_list

Definition at line 117 of file fipa_comm.h.

12.46.2.2 fipa_comm_reply_p fipa_comm_performative_t::default_reply

Definition at line 118 of file fipa_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.47 fipa_comm_protocol_cn_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- [protocol_contract_net_callback](#) [contract_func](#)
- [fipa_comm_callback](#) [proposal_func](#)
- char [require_inform](#)
- int [time_out](#)
- char * [cfp_request](#)
- [fipa_list_p](#) [replies](#)
- [fipa_list_p](#) [queue](#)

12.47.1 Detailed Description

Definition at line 143 of file `fipa_comm.h`.

12.47.2 Field Documentation

12.47.2.1 char* fipa_comm_protocol_cn_t::cfp_request

Definition at line 149 of file `fipa_comm.h`.

12.47.2.2 protocol_contract_net_callback fipa_comm_protocol_cn_t::contract_func

Definition at line 145 of file `fipa_comm.h`.

12.47.2.3 fipa_comm_callback fipa_comm_protocol_cn_t::proposal_func

Definition at line 146 of file `fipa_comm.h`.

12.47.2.4 fipa_list_p fipa_comm_protocol_cn_t::queue

Definition at line 153 of file `fipa_comm.h`.

12.47.2.5 fipa_list_p fipa_comm_protocol_cn_t::replies

Definition at line 152 of file `fipa_comm.h`.

12.47.2.6 char fipa_comm_protocol_cn_t::require_inform

Definition at line 147 of file `fipa_comm.h`.

12.47.2.7 int fipa_comm_protocol_cn_t::time_out

Definition at line 148 of file fipa_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.48 fipa_comm_protocol_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- char [error](#)
- char [state](#)
- char * [conversation_id](#)
- char * [network](#)
- enum [fipa_protocol_e](#) type
- void * [info](#)
- struct [fipa_comm_protocol_t](#) * [next](#)

12.48.1 Detailed Description

Definition at line 163 of file fipa_comm.h.

12.48.2 Field Documentation

12.48.2.1 char* fipa_comm_protocol_t::conversation_id

Definition at line 166 of file fipa_comm.h.

12.48.2.2 char fipa_comm_protocol_t::error

Definition at line 164 of file fipa_comm.h.

12.48.2.3 void* fipa_comm_protocol_t::info

Definition at line 169 of file fipa_comm.h.

12.48.2.4 char* fipa_comm_protocol_t::network

Definition at line 167 of file fipa_comm.h.

12.48.2.5 struct fipa_comm_protocol_t* fipa_comm_protocol_t::next [\[read\]](#)

Definition at line 170 of file fipa_comm.h.

12.48.2.6 char fipa_comm_protocol_t::state

Definition at line 165 of file fipa_comm.h.

12.48.2.7 enum fipa_protocol_e fipa_comm_protocol_t::type

Definition at line 168 of file fipa_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.49 fipa_comm_reply_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- enum [fipa_performative_e](#) type
- char * [content](#)

12.49.1 Detailed Description

Definition at line 86 of file [fipa_comm.h](#).

12.49.2 Field Documentation

12.49.2.1 char* [fipa_comm_reply_t::content](#)

Definition at line 89 of file [fipa_comm.h](#).

12.49.2.2 enum [fipa_performative_e](#) [fipa_comm_reply_t::type](#)

Definition at line 88 of file [fipa_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.50 fipa_comm_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- [int error](#)
- [char * agent_name](#)
- [char * agent_address](#)
- [void * agent](#)
- [char debug](#)
- [fipa_comm_performative_p performative](#) [MC_NUM_PERFORMATIVES]
- [fipa_comm_reply_p default_reply](#)
- [fipa_comm_protocol_p protocol](#) [FIPA_PROTOCOL_END]
- [fipa_comm_callback pCallbacks](#) [FIPA_PROTOCOL_END]

12.50.1 Detailed Description

Definition at line 179 of file fipa_comm.h.

12.50.2 Field Documentation

12.50.2.1 void* fipa_comm_t::agent

Definition at line 184 of file fipa_comm.h.

12.50.2.2 char* fipa_comm_t::agent_address

Definition at line 183 of file fipa_comm.h.

12.50.2.3 char* fipa_comm_t::agent_name

Definition at line 182 of file fipa_comm.h.

12.50.2.4 char fipa_comm_t::debug

Definition at line 185 of file fipa_comm.h.

12.50.2.5 fipa_comm_reply_p fipa_comm_t::default_reply

Definition at line 187 of file fipa_comm.h.

12.50.2.6 int fipa_comm_t::error

Definition at line 181 of file fipa_comm.h.

12.50.2.7 fipa_comm_callback fipa_comm_t::pCallbacks[FIPA_PROTOCOL_END]

Definition at line 189 of file fipa_comm.h.

12.50.2.8 fipa_comm_performative_p fipa_comm_t::performative[MC_NUM_PERFORMATIVES]

Definition at line 186 of file fipa_comm.h.

12.50.2.9 fipa_comm_protocol_p fipa_comm_t::protocol[FIPA_PROTOCOL_END]

Definition at line 188 of file fipa_comm.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.51 fipa_DateTime_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- [int year](#)
- [int month](#)
- [int day](#)
- [int hour](#)
- [int minute](#)
- [int second](#)
- [int millisecond](#)
- [char sign](#)
- [int is_utc](#)

12.51.1 Detailed Description

Definition at line 234 of file fipa_acl.h.

12.51.2 Field Documentation

12.51.2.1 int fipa_DateTime_s::day

Definition at line 238 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

12.51.2.2 int fipa_DateTime_s::hour

Definition at line 239 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

12.51.2.3 int fipa_DateTime_s::is_utc

Definition at line 244 of file fipa_acl.h.

12.51.2.4 int fipa_DateTime_s::millisecond

Definition at line 242 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

12.51.2.5 int fipa_DateTime_s::minute

Definition at line 240 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

12.51.2.6 int fipa_DateTime_s::month

Definition at line 237 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

12.51.2.7 int fipa_DateTime_s::second

Definition at line 241 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

12.51.2.8 char fipa_DateTime_s::sign

Definition at line 243 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose(), and fipa_datetime_Parse().

12.51.2.9 int fipa_DateTime_s::year

Definition at line 236 of file fipa_acl.h.

Referenced by fipa_DateTime_Compose().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.52 fipa_expression_s Struct Reference

```
#include <fipa_acl.h>
```

Data Structures

- union [content_u](#)

Data Fields

- enum [fipa_expression_type_e](#) type
- union [fipa_expression_s::content_u](#) content

12.52.1 Detailed Description

Definition at line 193 of file [fipa_acl.h](#).

12.52.2 Field Documentation

12.52.2.1 union [fipa_expression_s::content_u](#) [fipa_expression_s::content](#)

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), [fipa_expression_Destroy\(\)](#), and [MC_AclSetConversationID\(\)](#).

12.52.2.2 enum [fipa_expression_type_e](#) [fipa_expression_s::type](#)

Definition at line 195 of file [fipa_acl.h](#).

Referenced by [fipa_expression_Compose\(\)](#), [fipa_expression_Copy\(\)](#), [fipa_expression_Destroy\(\)](#), [fipa_expression_Parse\(\)](#), and [MC_AclSetConversationID\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.53 fipa_list_t Struct Reference

```
#include <fipa_comm.h>
```

Data Fields

- void * [data](#)
- struct [fipa_list_t](#) * [next](#)

12.53.1 Detailed Description

Definition at line 136 of file [fipa_comm.h](#).

12.53.2 Field Documentation

12.53.2.1 void* fipa_list_t::data

Definition at line 137 of file [fipa_comm.h](#).

12.53.2.2 struct fipa_list_t* fipa_list_t::next [read]

Definition at line 138 of file [fipa_comm.h](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h](#)

12.54 fipa_message_string_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- char * [message](#)
- char * [parse](#)

12.54.1 Detailed Description

Definition at line 143 of file [fipa_acl.h](#).

12.54.2 Field Documentation

12.54.2.1 char* fipa_message_string_s::message

Definition at line 145 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [fipa_envelope_HandleDate\(\)](#), [fipa_envelope_HandleReceived\(\)](#), [fipa_message_string_Copy\(\)](#), and [fipa_message_string_Destroy\(\)](#).

12.54.2.2 char* fipa_message_string_s::parse

Definition at line 146 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [fipa_agent_identifier_Parse\(\)](#), [fipa_CheckNextToken\(\)](#), [fipa_datetime_Parse\(\)](#), [fipa_envelope_HandleDate\(\)](#), [fipa_envelope_HandleReceived\(\)](#), [fipa_GetAtom\(\)](#), [fipa_GetNextWord\(\)](#), [fipa_GetWholeToken\(\)](#), [fipa_message_string_Copy\(\)](#), [fipa_string_Parse\(\)](#), and [fipa_word_Parse\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.55 fipa_number_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- char * [str](#)

12.55.1 Detailed Description

Definition at line 264 of file fipa_acl.h.

12.55.2 Field Documentation

12.55.2.1 char* fipa_number_s::str

Definition at line 266 of file fipa_acl.h.

Referenced by fipa_number_Compose(), fipa_number_Copy(), and fipa_number_Destroy().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.56 fipa_string_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- char * [content](#)

12.56.1 Detailed Description

Definition at line 223 of file fipa_acl.h.

12.56.2 Field Documentation

12.56.2.1 char* fipa_string_s::content

Definition at line 225 of file fipa_acl.h.

Referenced by fipa_string_Compose(), fipa_string_Copy(), fipa_string_Destroy(), fipa_string_Parse(), MC_AclSetContent(), and MC_AclSetConversationID().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.57 fipa_url_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- char * [str](#)

12.57.1 Detailed Description

Definition at line 253 of file fipa_acl.h.

12.57.2 Field Documentation

12.57.2.1 char* fipa_url_s::str

Definition at line 255 of file fipa_acl.h.

Referenced by `acc_connection_Thread()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__to()`, `fipa_envelope_HandleReceived()`, `fipa_envelope_ParseAddresses()`, `fipa_url_Compose()`, `fipa_url_Copy()`, `fipa_url_Destroy()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, `MC_AclSend()`, and `MC_AclSetSender()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h`

12.58 fipa_url_sequence_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- [int num](#)
- [struct fipa_url_s ** urls](#)

12.58.1 Detailed Description

Definition at line 154 of file [fipa_acl.h](#).

12.58.2 Field Documentation

12.58.2.1 int fipa_url_sequence_s::num

Definition at line 156 of file [fipa_acl.h](#).

Referenced by [fipa_agent_identifier_Compose\(\)](#), [fipa_envelope_Compose__from\(\)](#), [fipa_envelope_Compose__intended_receiver\(\)](#), [fipa_envelope_Compose__to\(\)](#), [fipa_envelope_ParseAddresses\(\)](#), [fipa_url_sequence_Compose\(\)](#), [fipa_url_sequence_Copy\(\)](#), [fipa_url_sequence_Destroy\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), [MC_AclSend\(\)](#), and [MC_AclSetSender\(\)](#).

12.58.2.2 struct fipa_url_s** fipa_url_sequence_s::urls [read]

Definition at line 157 of file [fipa_acl.h](#).

Referenced by [acc_connection_Thread\(\)](#), [fipa_envelope_Compose__from\(\)](#), [fipa_envelope_Compose__intended_receiver\(\)](#), [fipa_envelope_Compose__to\(\)](#), [fipa_envelope_ParseAddresses\(\)](#), [fipa_url_sequence_Compose\(\)](#), [fipa_url_sequence_Copy\(\)](#), [fipa_url_sequence_Destroy\(\)](#), [MC_AclAddReceiver\(\)](#), [MC_AclAddReplyTo\(\)](#), [MC_AclSend\(\)](#), and [MC_AclSetSender\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.59 fipa_word_s Struct Reference

```
#include <fipa_acl.h>
```

Data Fields

- char * [content](#)

12.59.1 Detailed Description

Definition at line 212 of file fipa_acl.h.

12.59.2 Field Documentation

12.59.2.1 char* fipa_word_s::content

Definition at line 214 of file fipa_acl.h.

Referenced by fipa_agent_identifier_Parse(), fipa_agent_identifier_set_Parse(), fipa_message_parameter_Parse(), fipa_message_type_Parse(), fipa_protocol_type_Parse(), fipa_url_Parse(), fipa_url_sequence_Parse(), fipa_word_Compose(), fipa_word_Copy(), and fipa_word_Destroy().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h](#)

12.60 foo_c Class Reference

Public Member Functions

- [foo_c](#) (float *f*, int *b*)
- [~foo_c](#) ()
- [int get_bar](#) ()
- [float get_foo](#) ()
- [void set_bar](#) (int *b*)
- [void set_foo](#) (float *f*)
- [void set_foobar](#) (float *f*, int *b*=0)
- [int get_barfoo](#) ()

Protected Member Functions

- [int get_global](#) ()

Static Protected Attributes

- static [int global](#)

Private Attributes

- [float foo](#)
- [int bar](#)
- [int barfoo](#)

12.60.1 Detailed Description

Definition at line 1 of file class.cxx.

12.60.2 Constructor & Destructor Documentation

12.60.2.1 foo_c::foo_c (float *f*, int *b*)

12.60.2.2 foo_c::~~foo_c ()

Definition at line 81 of file class.cxx.

12.60.3 Member Function Documentation

12.60.3.1 int foo_c::get_bar () **[inline]**

Definition at line 13 of file class.cxx.

References [bar](#).

12.60.3.2 int foo_c::get_barfoo () [inline]

Definition at line 67 of file class.cxx.

References barfoo.

12.60.3.3 float foo_c::get_foo () [inline]

Definition at line 20 of file class.cxx.

References foo.

12.60.3.4 int foo_c::get_global () [inline, protected]

Definition at line 54 of file class.cxx.

References global.

12.60.3.5 void foo_c::set_bar (int *b*) [inline]

Definition at line 27 of file class.cxx.

References bar.

12.60.3.6 void foo_c::set_foo (float *f*) [inline]

Definition at line 34 of file class.cxx.

References foo.

12.60.3.7 void foo_c::set_foobar (float *f*, int *b* = 0) [inline]

Definition at line 41 of file class.cxx.

References bar, and foo.

12.60.4 Field Documentation**12.60.4.1 int foo_c::bar [private]**

Definition at line 4 of file class.cxx.

Referenced by get_bar(), set_bar(), and set_foobar().

12.60.4.2 int foo_c::barfoo [private]

Definition at line 61 of file class.cxx.

Referenced by get_barfoo().

12.60.4.3 float foo_c::foo [private]

Definition at line 3 of file class.cxx.

Referenced by `get_foo()`, `set_foo()`, and `set_foobar()`.

12.60.4.4 int foo_c::global [static, protected]

Definition at line 50 of file class.cxx.

Referenced by `get_global()`.

The documentation for this class was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx`

12.61 foo_s Struct Reference

Public Member Functions

- [foo_s](#) (float [f](#), int [b](#))
- [~foo_s](#) ()
- [int get_bar](#) ()
- [float get_foo](#) ()
- [void set_bar](#) (int [b](#))
- [void set_foo](#) (float [f](#))

Data Fields

- [float foo](#)
- [int bar](#)

12.61.1 Detailed Description

Definition at line 1 of file struct.cxx.

12.61.2 Constructor & Destructor Documentation

12.61.2.1 foo_s::foo_s (float *f*, int *b*)

Definition at line 39 of file struct.cxx.

References [bar](#), and [foo](#).

12.61.2.2 foo_s::~~foo_s ()

Definition at line 46 of file struct.cxx.

12.61.3 Member Function Documentation

12.61.3.1 int foo_s::get_bar () [[inline](#)]

Definition at line 11 of file struct.cxx.

References [bar](#).

12.61.3.2 float foo_s::get_foo () [[inline](#)]

Definition at line 18 of file struct.cxx.

References [foo](#).

12.61.3.3 void foo_s::set_bar (int *b*) [inline]

Definition at line 25 of file struct.cxx.

References bar.

12.61.3.4 void foo_s::set_foo (float *f*) [inline]

Definition at line 32 of file struct.cxx.

References foo.

12.61.4 Field Documentation**12.61.4.1 int foo_s::bar**

Definition at line 4 of file struct.cxx.

Referenced by foo_s(), get_bar(), and set_bar().

12.61.4.2 float foo_s::foo

Definition at line 3 of file struct.cxx.

Referenced by foo_s(), get_foo(), and set_foo().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/[struct.cxx](#)

12.62 havege_state Struct Reference

HAVEGE state structure.

```
#include <havege.h>
```

Data Fields

- [int PT1](#)
- [int PT2](#)
- [int offset](#) [2]
- [int pool](#) [COLLECT_SIZE]
- [int WALK](#) [8192]

12.62.1 Detailed Description

HAVEGE state structure.

Definition at line 12 of file havege.h.

12.62.2 Field Documentation

12.62.2.1 int havege_state::offset[2]

Definition at line 14 of file havege.h.

Referenced by `havege_fill()`, and `havege_rand()`.

12.62.2.2 int havege_state::pool[COLLECT_SIZE]

Definition at line 15 of file havege.h.

Referenced by `havege_rand()`.

12.62.2.3 int havege_state::PT1

Definition at line 14 of file havege.h.

Referenced by `havege_fill()`.

12.62.2.4 int havege_state::PT2

Definition at line 14 of file havege.h.

Referenced by `havege_fill()`.

12.62.2.5 int havege_state::WALK[8192]

Definition at line 16 of file havege.h.

Referenced by `havege_fill()`.

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/havege.h>

12.63 host_id_s Struct Reference

```
#include <host_id.h>
```

Data Fields

- char * [hostname](#)
- [int](#) port

12.63.1 Detailed Description

Definition at line 38 of file `host_id.h`.

12.63.2 Field Documentation

12.63.2.1 char* host_id_s::hostname

Definition at line 40 of file `host_id.h`.

12.63.2.2 int host_id_s::port

Definition at line 41 of file `host_id.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/include/host_id.h`

12.64 hr_time Struct Reference

timer structure

```
#include <timing.h>
```

Data Fields

- unsigned char [opaque](#) [32]

12.64.1 Detailed Description

timer structure

Definition at line 10 of file timing.h.

12.64.2 Field Documentation

12.64.2.1 unsigned char hr_time::opaque[32]

Definition at line 12 of file timing.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/timing.h](#)

12.65 interpreter_variable_data_s Struct Reference

```
#include <interpreter_variable_data.h>
```

Data Fields

- char * [name](#)
- int [size](#)
- ChType_t [data_type](#)
- int [array_dim](#)
- int * [array_extent](#)
- void * [data](#)

12.65.1 Detailed Description

Definition at line 41 of file `interpreter_variable_data.h`.

12.65.2 Field Documentation

12.65.2.1 int interpreter_variable_data_s::array_dim

Definition at line 45 of file `interpreter_variable_data.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__data()`, `agent_xml_compose__row()`, `agent_xml_parse__data()`, `agent_xml_parse__row()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, `MC_AgentVariableRetrieveInfo()`, and `MC_GetAgentReturnData()`.

12.65.2.2 int* interpreter_variable_data_s::array_extent

Definition at line 46 of file `interpreter_variable_data.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__row()`, `agent_xml_parse__row()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Destroy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, `MC_AgentVariableRetrieveInfo()`, and `MC_GetAgentReturnData()`.

12.65.2.3 void* interpreter_variable_data_s::data

Definition at line 47 of file `interpreter_variable_data.h`.

Referenced by `agent_AddPersistentVariable()`, `agent_xml_compose__data()`, `agent_xml_compose__row()`, `agent_xml_parse__data()`, `agent_xml_parse__row()`, `interpreter_variable_data_Copy()`, `interpreter_variable_data_Destroy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `interpreter_variable_data_New()`, `MC_AgentVariableRetrieveInfo()`, `MC_AgentVariableRetrieveInfo()`, and `MC_SaveData_chdl()`.

12.65.2.4 ChType_t interpreter_variable_data_s::data_type

Definition at line 44 of file interpreter_variable_data.h.

Referenced by agent_AddPersistentVariable(), agent_xml_compose__data(), agent_xml_compose__row(), agent_xml_parse__data(), agent_xml_parse__row(), interpreter_variable_data_Copy(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), interpreter_variable_data_New(), MC_GetAgentReturnData(), and MC_SaveData_chdl().

12.65.2.5 char* interpreter_variable_data_s::name

Definition at line 42 of file interpreter_variable_data.h.

Referenced by agent_AddPersistentVariable(), agent_xml_compose__data(), agent_xml_parse__data(), interpreter_variable_data_Copy(), interpreter_variable_data_Destroy(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), interpreter_variable_data_New(), and MC_SaveData_chdl().

12.65.2.6 int interpreter_variable_data_s::size

Definition at line 43 of file interpreter_variable_data.h.

Referenced by agent_AddPersistentVariable(), interpreter_variable_data_Copy(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), interpreter_variable_data_New(), and MC_SaveData_chdl().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/interpreter_variable_data.h](#)

12.66 LibMC::InvalidAgencyException Class Reference

Exception class for use with null agency pointers.

Public Member Functions

- [InvalidAgencyException \(\)](#)
Null agency pointer exception constructor.
- [InvalidAgencyException \(String exc\)](#)
Null agency pointer exception constructor.

Private Attributes

- const String [msg](#) = "Private agency pointer not valid."

12.66.1 Detailed Description

Exception class for use with null agency pointers. This class provides a way to inform the program that an agency was created or accessed that had an invalid internal pointer.

Definition at line 1074 of file MCAgency.cs.

12.66.2 Constructor & Destructor Documentation

12.66.2.1 LibMC::InvalidAgencyException::InvalidAgencyException () [inline]

Null agency pointer exception constructor. Constructor for the exception class. This exception simply defines a recognizable exception class and sets the exception message appropriately.

Definition at line 1085 of file MCAgency.cs.

12.66.2.2 LibMC::InvalidAgencyException::InvalidAgencyException (String exc) [inline]

Null agency pointer exception constructor. Constructor for the exception class. Allows the use of a custom error message.

Definition at line 1096 of file MCAgency.cs.

12.66.3 Field Documentation

12.66.3.1 const String LibMC::InvalidAgencyException::msg = "Private agency pointer not valid." [private]

Definition at line 1076 of file MCAgency.cs.

The documentation for this class was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency.cs>

12.67 LibMC::InvalidAgentException Class Reference

Exception class for use with null agent pointers.

Public Member Functions

- [InvalidAgentException \(\)](#)
Null agent pointer exception constructor.
- [InvalidAgentException \(String exc\)](#)
Null agent pointer exception constructor.

Private Attributes

- const String [msg](#) = "Private agent pointer not valid."

12.67.1 Detailed Description

Exception class for use with null agent pointers. This class provides a way to inform the program that an agent was created or accessed that had an invalid internal pointer.

Definition at line 568 of file MCAgent.cs.

12.67.2 Constructor & Destructor Documentation

12.67.2.1 LibMC::InvalidAgentException::InvalidAgentException () [inline]

Null agent pointer exception constructor. Constructor for the exception class. This exception simply defines a recognizable exception class and sets the exception message appropriately.

Definition at line 579 of file MCAgent.cs.

12.67.2.2 LibMC::InvalidAgentException::InvalidAgentException (String exc) [inline]

Null agent pointer exception constructor. Constructor for the exception class. Allows the use of a custom error message.

Definition at line 590 of file MCAgent.cs.

12.67.3 Field Documentation

12.67.3.1 const String LibMC::InvalidAgentException::msg = "Private agent pointer not valid." [private]

Definition at line 570 of file MCAgent.cs.

The documentation for this class was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs

12.68 list_s Struct Reference

```
#include <list.h>
```

Data Fields

- [listNode_p listhead](#)
- [int size](#)

12.68.1 Detailed Description

Definition at line 41 of file list.h.

12.68.2 Field Documentation

12.68.2.1 listNode_p list_s::listhead

Definition at line 43 of file list.h.

Referenced by `barrier_queue_Add()`, `barrier_queue_Get()`, `ListAdd()`, `ListDelete()`, `ListGetHead()`, `ListInitialize()`, `ListPop()`, `ListSearch()`, `ListTerminate()`, `syncListAddNode()`, and `syncListFind()`.

12.68.2.2 int list_s::size

Definition at line 44 of file list.h.

Referenced by `barrier_queue_Delete()`, `list_pGetSize()`, `ListAdd()`, `ListDelete()`, `ListInitialize()`, `ListPop()`, `ListSearch()`, `ListTerminate()`, `syncListDelete()`, and `syncListRemove()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mc_list/list.h`

12.69 listNode_s Struct Reference

```
#include <list.h>
```

Data Fields

- DATA [node_data](#)
- struct [listNode_s](#) * [next](#)

12.69.1 Detailed Description

Definition at line 32 of file list.h.

12.69.2 Field Documentation

12.69.2.1 struct listNode_s* listNode_s::next [read]

Definition at line 35 of file list.h.

Referenced by `AP_QUEUE_STD_DEFN_TEMPLATE()`, `barrier_queue_Add()`, `barrier_queue_Get()`, `df_SearchForService()`, `ListAdd()`, `ListDelete()`, `ListPop()`, `ListSearch()`, `request_handler_DEREGISTER()`, `syncListAddNode()`, and `syncListFind()`.

12.69.2.2 DATA listNode_s::node_data

Definition at line 34 of file list.h.

Referenced by `AP_QUEUE_STD_DEFN_TEMPLATE()`, `barrier_queue_Add()`, `barrier_queue_Get()`, `df_SearchForService()`, `ListAdd()`, `ListDelete()`, `ListGetHead()`, `ListPop()`, `ListSearch()`, `request_handler_DEREGISTER()`, `syncListAddNode()`, and `syncListFind()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mc_list/list.h`

12.70 mc_platform_s Struct Reference

```
#include <mc_platform.h>
```

Data Fields

- [int err](#)
- [char * hostname](#)
- [unsigned short port](#)
- [int initInterps](#)
- [message_queue_p asm_message_queue](#)
- [message_queue_p message_queue](#)
- [agent_queue_p agent_queue](#)
- [connection_queue_p connection_queue](#)
- [df_p df](#)
- [ams_p ams](#)
- [acc_p acc](#)
- [cmd_prompt_p cmd_prompt](#)
- [syncList_p syncList](#)
- [barrier_queue_p barrier_queue](#)
- [listen_thread_arg_p listen_thread_arg](#)
- [listen_thread_arg_p client_thread_arg](#)
- [int default_agentstatus](#)
- [int stack_size \[MC_THREAD_ALL\]](#)
- [ChOptions_t * interp_options](#)
- [COND_T * MC_signal_cond](#)
- [COND_T * MC_sync_cond](#)
- [MUTEX_T * MC_signal_lock](#)
- [MUTEX_T * MC_sync_lock](#)
- [enum MC_Signal_e MC_signal](#)
- [enum MC_SteerCommand_e MC_steer_command](#)
- [MUTEX_T * MC_steer_lock](#)
- [COND_T * MC_steer_cond](#)
- [int giant](#)
- [MUTEX_T * giant_lock](#)
- [COND_T * giant_cond](#)
- [int quit](#)
- [MUTEX_T * quit_lock](#)
- [COND_T * quit_cond](#)
- [int sockfd](#)
- [char private_key \[1210\]](#)
- [MCAgency_t agency](#)
- [interpreter_queue_p interpreter_queue](#)

12.70.1 Detailed Description

Definition at line 49 of file mc_platform.h.

12.70.2 Field Documentation

12.70.2.1 `acc_p mc_platform_s::acc`

Definition at line 73 of file `mc_platform.h`.

Referenced by `acc_connection_Thread()`, `acc_MessageHandlerThread()`, `acc_Start()`, `acc_Thread()`, `listen_Thread()`, `MC_End()`, `mc_platform_Destroy()`, and `message_Send()`.

12.70.2.2 `MCAgency_t mc_platform_s::agency`

Definition at line 115 of file `mc_platform.h`.

Referenced by `handler_COMPOSE_SEND()`, `listen_Thread()`, `MC_Initialize()`, and `message_send_Thread()`.

12.70.2.3 `agent_queue_p mc_platform_s::agent_queue`

Definition at line 68 of file `mc_platform.h`.

Referenced by `acc_connection_Thread()`, `acc_MessageHandlerThread()`, `handler_FLUSH_AGENTS()`, `handler_PRINTLIST_AGENTS()`, `MC_AddAgent()`, `MC_AddStationaryAgent()`, `MC_FindAgentByID()`, `MC_FindAgentByName()`, `MC_GetAllAgents()`, `mc_platform_Destroy()`, `MC_RetrieveAgent()`, `MC_SendAgentMigrationMessageFile()`, `MC_WaitAgent()`, and `MC_WaitRetrieveAgent()`.

12.70.2.4 `ams_p mc_platform_s::ams`

Definition at line 72 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `agent_RunChScriptThread()`, `ams_Start()`, `ams_Thread()`, `MC_AddAgent()`, `MC_End()`, `mc_platform_Destroy()`, `MC_SendAgentMigrationMessageFile()`, and `MC_SetAgentStatus()`.

12.70.2.5 `message_queue_p mc_platform_s::asm_message_queue`

Definition at line 63 of file `mc_platform.h`.

12.70.2.6 `barrier_queue_p mc_platform_s::barrier_queue`

Definition at line 80 of file `mc_platform.h`.

Referenced by `MC_Barrier()`, `MC_BarrierDelete()`, `MC_BarrierInit()`, and `mc_platform_Destroy()`.

12.70.2.7 `listen_thread_arg_p mc_platform_s::client_thread_arg`

Definition at line 83 of file `mc_platform.h`.

12.70.2.8 `cmd_prompt_p mc_platform_s::cmd_prompt`

Definition at line 74 of file `mc_platform.h`.

Referenced by `cmd_prompt_Start()`, `MC_End()`, and `mc_platform_Destroy()`.

12.70.2.9 connection_queue_p mc_platform_s::connection_queue

Definition at line 69 of file mc_platform.h.

Referenced by acc_Thread(), handler_PRINT_CONNECTLIST(), listen_Thread(), MC_End(), and mc_platform_Destroy().

12.70.2.10 int mc_platform_s::default_agentstatus

Definition at line 85 of file mc_platform.h.

Referenced by agent_Initialize(), and MC_SetDefaultAgentStatus().

12.70.2.11 df_p mc_platform_s::df

Definition at line 71 of file mc_platform.h.

Referenced by df_ProcessRequest(), df_Start(), df_Thread(), MC_DeregisterService(), MC_End(), mc_platform_Destroy(), MC_RegisterService(), MC_SearchForService(), request_handler_DEREGISTER(), request_handler_REGISTER(), and request_handler_SEARCH().

12.70.2.12 int mc_platform_s::err

Definition at line 53 of file mc_platform.h.

Referenced by agent_Initialize(), message_InitializeFromAgent(), and message_InitializeFromString().

12.70.2.13 int mc_platform_s::giant

Definition at line 105 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), MC_GetAllAgents(), MC_HaltAgency(), MC_ResetSignal(), MC_ResumeAgency(), and MC_WaitSignal().

12.70.2.14 COND_T* mc_platform_s::giant_cond

Definition at line 107 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), mc_platform_Destroy(), and MC_ResetSignal().

12.70.2.15 MUTEX_T* mc_platform_s::giant_lock

Definition at line 106 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), MC_GetAllAgents(), MC_HaltAgency(), mc_platform_Destroy(), MC_ResetSignal(), MC_ResumeAgency(), and MC_WaitSignal().

12.70.2.16 char* mc_platform_s::hostname

Definition at line 58 of file mc_platform.h.

Referenced by `agent_Initialize()`, `agent_NewBinary()`, `fipa_envelope_Compose__from()`, `message_InitializeFromAgent()`, `message_InitializeFromString()`, `message_queue_SendOutgoing()`, and `udplisten_Thread()`.

12.70.2.17 `int mc_platform_s::initInterps`

Definition at line 60 of file `mc_platform.h`.

12.70.2.18 `ChOptions_t* mc_platform_s::interp_options`

Definition at line 89 of file `mc_platform.h`.

Referenced by `agent_RunChScriptThread()`, and `mc_platform_Destroy()`.

12.70.2.19 `interpreter_queue_p mc_platform_s::interpreter_queue`

Definition at line 118 of file `mc_platform.h`.

Referenced by `agent_Destroy()`, and `agent_RunChScriptThread()`.

12.70.2.20 `listen_thread_arg_p mc_platform_s::listen_thread_arg`

Definition at line 82 of file `mc_platform.h`.

12.70.2.21 `enum MC_Signal_e mc_platform_s::MC_signal`

Definition at line 96 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `agent_RunChScriptThread()`, `MC_ResetSignal()`, and `MC_WaitSignal()`.

12.70.2.22 `COND_T* mc_platform_s::MC_signal_cond`

Definition at line 92 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `agent_RunChScriptThread()`, `mc_platform_Destroy()`, and `MC_WaitSignal()`.

12.70.2.23 `MUTEX_T* mc_platform_s::MC_signal_lock`

Definition at line 94 of file `mc_platform.h`.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `agent_RunChScriptThread()`, `mc_platform_Destroy()`, and `MC_WaitSignal()`.

12.70.2.24 `enum MC_SteerCommand_e mc_platform_s::MC_steer_command`

Definition at line 99 of file `mc_platform.h`.

Referenced by `MC_SendSteerCommand()`, `MC_Steer()`, and `MC_SteerControl()`.

12.70.2.25 COND_T* mc_platform_s::MC_steer_cond

Definition at line 101 of file mc_platform.h.

Referenced by mc_platform_Destroy(), MC_SendSteerCommand(), and MC_SteerControl().

12.70.2.26 MUTEX_T* mc_platform_s::MC_steer_lock

Definition at line 100 of file mc_platform.h.

Referenced by mc_platform_Destroy(), MC_SendSteerCommand(), MC_Steer(), and MC_SteerControl().

12.70.2.27 COND_T* mc_platform_s::MC_sync_cond

Definition at line 93 of file mc_platform.h.

Referenced by mc_platform_Destroy().

12.70.2.28 MUTEX_T* mc_platform_s::MC_sync_lock

Definition at line 95 of file mc_platform.h.

Referenced by mc_platform_Destroy().

12.70.2.29 message_queue_p mc_platform_s::message_queue

Definition at line 67 of file mc_platform.h.

Referenced by acc_connection_Thread(), acc_MessageHandlerThread(), handler_PRINTLIST_MESSAGE(), MC_End(), MC_LoadAgentFromFile(), mc_platform_Destroy(), and MC_SendAgentMigrationMessage().

12.70.2.30 unsigned short mc_platform_s::port

Definition at line 59 of file mc_platform.h.

Referenced by acc_connection_Thread(), agent_Initialize(), agent_NewBinary(), fipa_envelope_Compose__from(), listen_Thread(), message_InitializeFromAgent(), message_InitializeFromString(), message_queue_SendOutgoing(), and udplisten_Thread().

12.70.2.31 char mc_platform_s::private_key[1210]

Definition at line 113 of file mc_platform.h.

Referenced by acc_connection_Thread(), listen_Thread(), and MC_AclSend().

12.70.2.32 int mc_platform_s::quit

Definition at line 109 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), ams_ManageAgentList(), ams_Thread(), df_Thread(), handler_QUIT(), MC_End(), and MC_MainLoop().

12.70.2.33 COND_T* mc_platform_s::quit_cond

Definition at line 111 of file mc_platform.h.

Referenced by handler_QUIT(), MC_End(), and MC_MainLoop().

12.70.2.34 MUTEX_T* mc_platform_s::quit_lock

Definition at line 110 of file mc_platform.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), ams_ManageAgentList(), ams_Thread(), df_Thread(), handler_QUIT(), MC_End(), MC_MainLoop(), and mc_platform_Destroy().

12.70.2.35 int mc_platform_s::sockfd

Definition at line 112 of file mc_platform.h.

Referenced by listen_Thread(), and mc_platform_Destroy().

12.70.2.36 int mc_platform_s::stack_size[MC_THREAD_ALL]

Definition at line 87 of file mc_platform.h.

Referenced by acc_Start(), agent_RunChScript(), ams_Start(), cmd_prompt_Start(), and df_Start().

12.70.2.37 syncList_p mc_platform_s::syncList

Definition at line 79 of file mc_platform.h.

Referenced by MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), MC_MutexLock(), MC_MutexUnlock(), MC_SemaphorePost(), MC_SemaphoreWait(), MC_SyncDelete(), and MC_SyncInit().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[mc_platform.h](#)

12.71 mc_rwlock_s Struct Reference

```
#include <mc_rwlock.h>
```

Data Fields

- [int num_readers](#)
- [int write_flag](#)
- [int write_request](#)
- [MUTEX_T * lock](#)
- [COND_T * cond](#)

12.71.1 Detailed Description

Definition at line 39 of file mc_rwlock.h.

12.71.2 Field Documentation

12.71.2.1 COND_T* mc_rwlock_s::cond

Definition at line 44 of file mc_rwlock.h.

Referenced by [mc_rwlock_destroy\(\)](#), [mc_rwlock_init\(\)](#), [mc_rwlock_rdlock\(\)](#), [mc_rwlock_rdunlock\(\)](#), [mc_rwlock_wrlock\(\)](#), and [mc_rwlock_wrunlock\(\)](#).

12.71.2.2 MUTEX_T* mc_rwlock_s::lock

Definition at line 43 of file mc_rwlock.h.

Referenced by [mc_rwlock_destroy\(\)](#), [mc_rwlock_init\(\)](#), [mc_rwlock_rdlock\(\)](#), [mc_rwlock_rdunlock\(\)](#), [mc_rwlock_wrlock\(\)](#), and [mc_rwlock_wrunlock\(\)](#).

12.71.2.3 int mc_rwlock_s::num_readers

Definition at line 40 of file mc_rwlock.h.

Referenced by [mc_rwlock_init\(\)](#), [mc_rwlock_rdlock\(\)](#), [mc_rwlock_rdunlock\(\)](#), and [mc_rwlock_wrlock\(\)](#).

12.71.2.4 int mc_rwlock_s::write_flag

Definition at line 41 of file mc_rwlock.h.

Referenced by [mc_rwlock_init\(\)](#), [mc_rwlock_rdlock\(\)](#), [mc_rwlock_wrlock\(\)](#), and [mc_rwlock_wrunlock\(\)](#).

12.71.2.5 int mc_rwlock_s::write_request

Definition at line 42 of file mc_rwlock.h.

Referenced by [mc_rwlock_init\(\)](#), [mc_rwlock_rdlock\(\)](#), and [mc_rwlock_wrlock\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/mc_rwlock.h](#)

12.72 LibMC::MCAclMessage Class Reference

Encapsulates ACL messages in the Mobile-C library.

Public Types

- enum [MC_FipaPerformative_e](#) {
 [FIPA_ERROR](#) = -1, [FIPA_ZERO](#), [FIPA_ACCEPT_PROPOSAL](#), [FIPA_AGREE](#),
 [FIPA_CANCEL](#), [FIPA_CALL_FOR_PROPOSAL](#), [FIPA_CONFIRM](#), [FIPA_DISCONFIRM](#),
 [FIPA_FAILURE](#), [FIPA_INFORM](#), [FIPA_INFORM_IF](#), [FIPA_INFORM_REF](#),
 [FIPA_NOT_UNDERSTOOD](#), [FIPA_PROPOGATE](#), [FIPA_PROPOSE](#), [FIPA_PROXY](#),
 [FIPA_QUERY_IF](#), [FIPA_QUERY_REF](#), [FIPA_REFUSE](#), [FIPA_REJECT_PROPOSAL](#),
 [FIPA_REQUEST](#), [FIPA_REQUEST_WHEN](#), [FIPA_REQUEST_WHENEVER](#), [FIPA_-
SUBSCRIBE](#) }

Enum for describing the type of an ACL message.

Public Member Functions

- [MCAclMessage](#) ()
Default constructor.
- void [New](#) ()
Creates a new, blank ACL message.
- [MCAclMessage Reply](#) ([MCAclMessage](#) acl_message)
Creates an ACL message that is a response to the argument.
- int [SetPerformative](#) ([MC_FipaPerformative_e](#) performative)
Sets the performative field of the message.
- int [SetSender](#) (String name, String address)
Sets the sender field of the message.
- int [AddReceiver](#) (String name, String address)
Adds a receiver to the list of receivers.
- int [AddReplyTo](#) (String name, String address)
Adds a "reply-to" field to the message.
- int [SetContent](#) (String content)
Sets the content field of the message.
- int [Destroy](#) ()
Destroys a message.

Properties

- internal IntPtr [AclMsg](#) [get, set]

Private Member Functions

- internal [MCAclMessage](#) (IntPtr msg)

Private Attributes

- IntPtr [aclmsg_p](#)

12.72.1 Detailed Description

Encapsulates ACL messages in the Mobile-C library. This class contains a pointer to an ACL message in the Mobile-C library. Functions are provided to send the message, set its various fields, and destroy the message.

Definition at line 72 of file MCAclMessage.cs.

12.72.2 Member Enumeration Documentation

12.72.2.1 enum LibMC::MCAclMessage::MC_FipaPerformative_e

Enum for describing the type of an ACL message.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

FIPA_ERROR Fipa performative enum value
FIPA_ZERO Fipa performative enum value
FIPA_ACCEPT_PROPOSAL Fipa performative enum value
FIPA_AGREE Fipa performative enum value
FIPA_CANCEL Fipa performative enum value
FIPA_CALL_FOR_PROPOSAL Fipa performative enum value
FIPA_CONFIRM Fipa performative enum value
FIPA_DISCONFIRM Fipa performative enum value
FIPA_FAILURE Fipa performative enum value
FIPA_INFORM Fipa performative enum value
FIPA_INFORM_IF Fipa performative enum value
FIPA_INFORM_REF Fipa performative enum value
FIPA_NOT_UNDERSTOOD Fipa performative enum value
FIPA_PROPOGATE Fipa performative enum value
FIPA_PROPOSE Fipa performative enum value

FIPA_PROXY Fipa performative enum value
FIPA_QUERY_IF Fipa performative enum value
FIPA_QUERY_REF Fipa performative enum value
FIPA_REFUSE Fipa performative enum value
FIPA_REJECT_PROPOSAL Fipa performative enum value
FIPA_REQUEST Fipa performative enum value
FIPA_REQUEST_WHEN Fipa performative enum value
FIPA_REQUEST_WHENEVER Fipa performative enum value
FIPA_SUBSCRIBE Fipa performative enum value

Definition at line 79 of file MCAclMessage.cs.

12.72.3 Constructor & Destructor Documentation

12.72.3.1 LibMC::MCAclMessage::MCAclMessage () [inline]

Default constructor. Creates an empty ACL message object.

Definition at line 114 of file MCAclMessage.cs.

References `aclmsg_p`.

Referenced by `Reply()`.

12.72.3.2 internal LibMC::MCAclMessage::MCAclMessage (IntPtr msg) [inline, private]

Definition at line 120 of file MCAclMessage.cs.

References `AclMsg`.

12.72.4 Member Function Documentation

12.72.4.1 int LibMC::MCAclMessage::AddReceiver (String name, String address) [inline]

Adds a receiver to the list of receivers. Adds a receiver to the list of receivers for the message.

Parameters:

name The name of the receiver.
address The address of the receiver.

Returns:

The return value of the underlying `MC_AclAddReceiver` function.

Note:

The message must be a valid message or this function will fail.

Definition at line 218 of file MCAclMessage.cs.

References `AclMsg`.

12.72.4.2 int LibMC::MCAclMessage::AddReplyTo (String *name*, String *address*) [inline]

Adds a "reply-to" field to the message. Adds a "reply-to" field to the message. The reply-to field overrides the sender field when creating a reply.

Parameters:

name The name of the receiver.
address The address of the receiver.

Returns:

The return value of the underlying MC_AclAddAddReplyTo function.

Note:

The message must be a valid message or this function will fail.

Definition at line 235 of file MCAclMessage.cs.

References AclMsg.

12.72.4.3 int LibMC::MCAclMessage::Destroy () [inline]

Destroys a message. This function destroys a message in the Mobile-C library. It releases the underlying memory and must be called when the message is no longer needed.

Returns:

The return value of the underlying MC_AclDestroy function.

Note:

The message must be a valid message or this function will fail. In addition, messages are not automatically destroyed by the garbage collector. Use care when creating messages and ensure they are properly destroyed.

Definition at line 269 of file MCAclMessage.cs.

References AclMsg, and aclmsg_p.

12.72.4.4 void LibMC::MCAclMessage::New () [inline]

Creates a new, blank ACL message. Creates a new ACL message. The message is blank but valid.

Definition at line 153 of file MCAclMessage.cs.

References AclMsg.

12.72.4.5 MCAclMessage LibMC::MCAclMessage::Reply (MCAclMessage *acl_message*) [inline]

Creates an ACL message that is a response to the argument. Creates an ACL message to respond to the argument.

Parameters:

acl_message The message from which to create the reply.

Returns:

A new ACL message that is a response to the argument or an empty message if there is an error.

Definition at line 167 of file MCAclMessage.cs.

References AclMsg, and MCAclMessage().

12.72.4.6 int LibMC::MCAclMessage::SetContent (String *content*) [inline]

Sets the content field of the message. Sets the content field of the message.

Parameters:

content The string to copy to the content field.

Returns:

The return value of the underlying MC_AclSetContent function.

Note:

The message must be a valid message or this function will fail.

Definition at line 250 of file MCAclMessage.cs.

References AclMsg.

12.72.4.7 int LibMC::MCAclMessage::SetPerformative (MC_FipaPerformative_e *performative*) [inline]

Sets the performative field of the message. Sets the performative field of the message.

Parameters:

performative The fipa_performative_e enum describing the message.

Returns:

The return value of the underlying MC_AclSetPerformative function.

Note:

The message must be a valid message or this function will fail.

Definition at line 186 of file MCAclMessage.cs.

References AclMsg.

12.72.4.8 `int LibMC::MCAclMessage::SetSender (String name, String address) [inline]`

Sets the sender field of the message. Sets the performative field of the message.

Parameters:

- name* The name of the sending entity.
address The address of the sending entity.

Returns:

The return value of the underlying MC_AclSetSender function.

Note:

The message must be a valid message or this function will fail.

Definition at line 202 of file MCAclMessage.cs.

References AclMsg.

12.72.5 Field Documentation

12.72.5.1 `IntPtr LibMC::MCAclMessage::aclmsg_p [private]`

Definition at line 105 of file MCAclMessage.cs.

Referenced by Destroy(), and MCAclMessage().

12.72.6 Property Documentation

12.72.6.1 `internal IntPtr LibMC::MCAclMessage::AclMsg [get, set, private]`

Definition at line 129 of file MCAclMessage.cs.

Referenced by LibMC::MCAgent::AclPost(), LibMC::MCAgency::AclSend(), AddReceiver(), AddReplyTo(), Destroy(), MCAclMessage(), New(), Reply(), SetContent(), SetPerformative(), and SetSender().

The documentation for this class was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclMessage.cs>

12.73 LibMC::MCAgency Class Reference

Wrapper class for [MCAgency_t](#) structure.

Data Structures

- struct [ChOptions_t](#)
ChOptions structures.
- struct [MCAgency_t](#)
- struct [MCAgencyOptions_t](#)

Public Types

- enum [MCAgencyState](#) {
[NoState](#) = -1, [Initialized](#) = 0, [Running](#), [Halted](#),
[Ended](#) }
Enum for describing the state of the agency.
- enum [ChShellType](#) { [CH_REGULARCH](#) = 0, [CH_SAFECH](#) = 1 }
Ch shell type.
- enum [MC_ThreadIndex_e](#) {
[MC_THREAD_DF](#) = 0, [MC_THREAD_AMS](#), [MC_THREAD_ACC](#), [MC_THREAD_CP](#),
[MC_THREAD_AGENT](#), [MC_THREAD_ALL](#) }
Enum for describing the different threads that Mobile-C uses.
- enum [MC_SteerCommand_e](#) { [MC_RUN](#) = 0, [MC_SUSPEND](#), [MC_RESTART](#), [MC_STOP](#) }
Available commands for MC_Steer.
- enum [MC_Signal_e](#) {
[MC_NO_SIGNAL](#) = 0x00, [MC_RECV_CONNECTION](#) = 0x01, [MC_RECV_MESSAGE](#) = 0x02,
[MC_RECV_AGENT](#) = 0x04,
[MC_RECV_RETURN](#) = 0x08, [MC_EXEC_AGENT](#) = 0x10, [MC_ALL_SIGNALS](#) = 0x20 }
MobileC system signals.

Public Member Functions

- [MCAgency](#) ()
Default constructor.
- [int Initialize](#) ()
Starts the agency.
- [int End](#) ()
Stops and destroys the agency.

- [int ChInitializeOptions](#) ([ChShellType](#) shellType, String home)
Initializes Ch [options](#) for the agency.
- [int SetThreadsAllOn](#) ()
- [int SetThreadsAllOff](#) ()
Sets all threads for the agency to "off".
- [int SetThreadOn](#) ([MC_ThreadIndex_e](#) index)
Sets an individual thread for the agency to "on".
- [int SetThreadOff](#) ([MC_ThreadIndex_e](#) index)
Sets an individual thread for the agency to "off".
- [int HaltAgency](#) ()
Temporarily halts the agency.
- [int ResumeAgency](#) ()
Resumes a halted agency.
- [int SetDefaultAgentStatus](#) ([MCAgent.MC_AgentStatus_e](#) status)
Sets the default state of an agent in the agency.
- [MCAgent WaitRetrieveAgent](#) ()
Waits for an agent to arrive and returns the agent.
- [int WaitAgent](#) ()
Waits for an agent to arrive.
- [int SendAgentMigrationMessageFile](#) (String filename, String hostname, [int](#) port)
Sends an agent migration message file to an agency.
- [int LoadAgentMigrationMessageFile](#) (String filename)
Load an agent migration message.
- [int SendAgentMigrationMessage](#) (String message, String hostname, [int](#) port)
Sends an agent migration message to an agency.
- [int CondBroadcast](#) ([int](#) id)
Broadcast a condition signal.
- [int CondSignal](#) ([int](#) id)
Signal a condition.
- [int CondReset](#) ([int](#) id)
Reset a condition signal.
- [int CondWait](#) ([int](#) id)
Wait for a condition signal.

- [int MutexLock \(int id\)](#)
Lock a mutex.
- [int MutexUnlock \(int id\)](#)
Unlock a mutex.
- [int SemaphorePost \(int id\)](#)
Posts a semaphore.
- [int SemaphoreWait \(int id\)](#)
Wait for a semaphore to be posted.
- [int ResetSignal \(\)](#)
Reset an agency signal.
- [int SyncDelete \(int id\)](#)
Delete a synchronization variable.
- [int SyncInit \(int id\)](#)
Create a new synchronization variable.
- [int WaitSignal \(MC_Signal_esignals\)](#)
Wait for agency signals.
- [int BarrierDelete \(int id\)](#)
Delete a barrier object.
- [int BarrierInit \(int id, int num_procs\)](#)
Create a new barrier.
- [MC_SteerCommand_e SteerControl \(\)](#)
Steering control function.
- [int Steer \(IntPtr funcptr, IntPtr arg\)](#)
Steering control function.
- [int RegisterService \(MCAgent agent, int agentID, String agentName, String\[\] serviceNames, int numServices\)](#)
Registers services in the agency.
- [int SearchForService \(String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr numResults\)](#)
Searches for services in the agency.
- [int AddAgent \(MCAgent agent\)](#)
Add an agent to the agency.
- [MCAgent FindAgentByName \(String name\)](#)
Finds an agent by its name.

- [MCAgent FindAgentByID](#) (int id)
Find an agent by its ID.
- [MCAgent RetrieveAgent](#) ()
Retrieve an agent from the agency.
- [int AclSend](#) ([MCAclMessage](#) acl_message)
Send an ACL message to the agency.
- [int MainLoop](#) ()
Makes the agency wait indefinitely.

Properties

- [IntPtr Agency](#) [get, set]
- [int Port](#) [get, set]
Accessor for the port number of the agency.
- [MCAgencyState State](#) [get]
Accessor for the agency state.

Private Member Functions

- static [IntPtr _MC_Initialize](#) (int port, ref [MCAgencyOptions_t](#) options)
- static [int _MC_End](#) (IntPtr agency)
- static [int _MC_ChInitializeOptions](#) (IntPtr agency, [ChOptions_t](#) options)
- static [int _MC_InitializeAgencyOptions](#) (ref [MCAgencyOptions_t](#) options)
- static [int _MC_SetThreadsAllOn](#) (ref [MCAgencyOptions_t](#) options)
- static [int _MC_SetThreadsAllOff](#) (ref [MCAgencyOptions_t](#) options)
- static [int _MC_SetThreadOn](#) (ref [MCAgencyOptions_t](#) options, [MC_ThreadIndex_e](#) index)
- static [int _MC_SetThreadOff](#) (ref [MCAgencyOptions_t](#) options, [MC_ThreadIndex_e](#) index)
- static [int _MC_HaltAgency](#) (IntPtr agency)
- static [int _MC_ResumeAgency](#) (IntPtr agency)
- static [int _MC_SetDefaultAgentStatus](#) (IntPtr agency, [MCAgent.MC_AgentStatus_e](#) status)
- static [IntPtr _MC_WaitRetrieveAgent](#) (IntPtr agency)
- static [int _MC_WaitAgent](#) (IntPtr agency)
- static [int _MC_MainLoop](#) (IntPtr agency)
- static [int _MC_SendAgentMigrationMessageFile](#) (IntPtr agency, String filename, String hostname, int port)
- static [int _MC_SendAgentMigrationMessage](#) (IntPtr agency, String message, String hostname, int port)
- static [int _MC_CondBroadcast](#) (IntPtr agency, int id)
- static [int _MC_CondSignal](#) (IntPtr agency, int id)
- static [int _MC_CondReset](#) (IntPtr agency, int id)
- static [int _MC_CondWait](#) (IntPtr agency, int id)
- static [int _MC_MutexLock](#) (IntPtr agency, int id)
- static [int _MC_MutexUnlock](#) (IntPtr agency, int id)

- static [int _MC_SemaphorePost](#) (IntPtr agency, [int](#) id)
- static [int _MC_SemaphoreWait](#) (IntPtr agency, [int](#) id)
- static [int _MC_ResetSignal](#) (IntPtr agency)
- static [int _MC_SyncDelete](#) (IntPtr agency, [int](#) id)
- static [int _MC_SyncInit](#) (IntPtr agency, [int](#) id)
- static [int _MC_WaitSignal](#) (IntPtr agency, [int](#) signals)
- static [int _MC_BarrierDelete](#) (IntPtr agency, [int](#) id)
- static [int _MC_BarrierInit](#) (IntPtr agency, [int](#) id, [int](#) num_procs)
- static [MC_SteerCommand_e _MC_SteerControl](#) ()
- static [int _MC_Steer](#) (IntPtr agency, IntPtr funcptr, IntPtr arg)
- static [int _MC_RegisterService](#) (IntPtr agency, IntPtr agent, [int](#) agentID, String agentName, String[] serviceNames, [int](#) numServices)
- static [int _MC_SearchForService](#) (IntPtr agency, String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr numResults)
- static [int _MC_AddAgent](#) (IntPtr agency, IntPtr agent)
- static internal [int _MC_DeleteAgent](#) (IntPtr agent)
- static IntPtr [_MC_FindAgentByName](#) (IntPtr agency, String name)
- static IntPtr [_MC_FindAgentByID](#) (IntPtr agency, [int](#) ID)
- static IntPtr [_MC_RetrieveAgent](#) (IntPtr agency)
- static internal [int _MC_GetAgentID](#) (IntPtr agent)
- static internal String [_MC_GetAgentName](#) (IntPtr agent)
- static internal [int _MC_GetAgentNumTasks](#) (IntPtr agent)
- static internal [MCAgent.MC_AgentStatus_e _MC_GetAgentStatus](#) (IntPtr agent)
- static internal [MCAgent.MC_AgentType_e _MC_GetAgentType](#) (IntPtr agent)
- static internal String [_MC_GetAgentXMLString](#) (IntPtr agent)
- static internal [int _MC_PrintAgentCode](#) (IntPtr agent)
- static internal String [_MC_RetrieveAgentCode](#) (IntPtr agent)
- static internal [int _MC_SetAgentStatus](#) (IntPtr agent, [MCAgent.MC_AgentStatus_e](#) status)
- static internal [int _MC_TerminateAgent](#) (IntPtr agent)
- static internal [int _MC_CallAgentFunc](#) (IntPtr agent, String funcName, IntPtr returnVal, IntPtr varg)
- static internal IntPtr [_MC_GetAgentExecEngine](#) (IntPtr agent)
- static internal [int _MC_GetAgentReturnData](#) (IntPtr agent, [int](#) task_num, IntPtr data, IntPtr dim, IntPtr extent)
- static internal IntPtr [_MC_AclNew](#) ()
- static internal [int _MC_AclPost](#) (IntPtr agent, IntPtr message)
- static internal IntPtr [_MC_AclReply](#) (IntPtr acl_message)
- static internal IntPtr [_MC_AclRetrieve](#) (IntPtr agent)
- static internal [int _MC_AclSend](#) (IntPtr agency, IntPtr acl_message)
- static internal IntPtr [_MC_AclWaitRetrieve](#) (IntPtr agent)
- static internal [int _MC_AclSetPerformative](#) (IntPtr acl, [MCAclMessage.MC_FipaPerformative_e](#) performative)
- static internal [int _MC_AclSetSender](#) (IntPtr acl, String name, String address)
- static internal [int _MC_AclAddReceiver](#) (IntPtr acl, String name, String address)
- static internal [int _MC_AclAddReplyTo](#) (IntPtr acl, String name, String address)
- static internal [int _MC_AclSetContent](#) (IntPtr acl, String content)
- static internal [int _MC_AclDestroy](#) (IntPtr acl)

Private Attributes

- IntPtr [agency_p](#) = IntPtr.Zero
- [MCAgencyOptions_t options](#)
- int [port](#) = -1
- [MCAgencyState state](#) = MCAgencyState.NoState
- const String [mcdll](#)

12.73.1 Detailed Description

Wrapper class for [MCAgency_t](#) structure. This class provides an interface to the Mobile-C agency. Member functions for the class are generally overloaded versions of the respective functions in the Mobile-C library. The class maintains a pointer to the Mobile-C agency in unmanaged memory. The pointer is not accessible by the user.

Definition at line 341 of file MCAgency.cs.

12.73.2 Member Enumeration Documentation

12.73.2.1 enum LibMC::MCAgency::ChShellType

Ch shell type. Used to set the shell type for the Ch interpreter.

Enumerator:

CH_REGULARCH Default, regular shell
CH_SAFECH Safe shell

Definition at line 363 of file MCAgency.cs.

12.73.2.2 enum LibMC::MCAgency::MC_Signal_e

MobileC system signals. Each signal is activated after the corresponding action. i.e. The 'MC_RECV_MESSAGE' signal is activated after a message is received.

Note:

This enum is pulled directly from the Mobile-C library.

See also:

[MC_WaitSignal\(\)](#), [MC_ResetSignal\(\)](#)

Enumerator:

MC_NO_SIGNAL
MC_RECV_CONNECTION
MC_RECV_MESSAGE
MC_RECV_AGENT
MC_RECV_RETURN
MC_EXEC_AGENT
MC_ALL_SIGNALS

Definition at line 112 of file MCExports.cs.

12.73.2.3 enum LibMC::MCAgency::MC_SteerCommand_e

Available commands for MC_Steer.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

MC_RUN Continue the algorithm
MC_SUSPEND Suspend/pause the algorithm
MC_RESTART Restart the algorithm from the beginning
MC_STOP Stop the algorithm

Definition at line 94 of file MCExports.cs.

12.73.2.4 enum LibMC::MCAgency::MC_ThreadIndex_e

Enum for describing the different threads that Mobile-C uses. These enums can be used to turn threads on and off before an agency is initialized.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

MC_THREAD_DF Directory Facilitator
MC_THREAD_AMS Agent Management system
MC_THREAD_ACC Agency communications
MC_THREAD_CP Command Prompt
MC_THREAD_AGENT Agent threads
MC_THREAD_ALL

Definition at line 79 of file MCExports.cs.

12.73.2.5 enum LibMC::MCAgency::MCAgencyState

Enum for describing the state of the agency. This enum is used to determine whether or not certain actions should be permitted, such as halting, resuming, and ending an agency

Enumerator:

NoState Default, uninitialized state
Initialized Agency initialized, but not started
Running Agency is running
Halted Agency has been stopped (can be resumed)
Ended Agency is stopped (destroyed)

Definition at line 349 of file MCAgency.cs.

12.73.3 Constructor & Destructor Documentation

12.73.3.1 LibMC::MCAgency::MCAgency () [inline]

Default constructor. The default constructor for the [MCAgency](#) class. It creates a new agency, default [options](#) for the agency, and initializes the agency. It does not start the agency.

Definition at line 381 of file MCAgency.cs.

References `_MC_InitializeAgencyOptions()`, and `state`.

12.73.4 Member Function Documentation

12.73.4.1 static internal int LibMC::MCAgency::_MC_AclAddReceiver (IntPtr *acl*, String *name*, String *address*) [private]

12.73.4.2 static internal int LibMC::MCAgency::_MC_AclAddReplyTo (IntPtr *acl*, String *name*, String *address*) [private]

12.73.4.3 static internal int LibMC::MCAgency::_MC_AclDestroy (IntPtr *acl*) [private]

12.73.4.4 static internal IntPtr LibMC::MCAgency::_MC_AclNew () [private]

12.73.4.5 static internal int LibMC::MCAgency::_MC_AclPost (IntPtr *agent*, IntPtr *message*) [private]

12.73.4.6 static internal IntPtr LibMC::MCAgency::_MC_AclReply (IntPtr *acl_message*) [private]

12.73.4.7 static internal IntPtr LibMC::MCAgency::_MC_AclRetrieve (IntPtr *agent*) [private]

12.73.4.8 static internal int LibMC::MCAgency::_MC_AclSend (IntPtr *agency*, IntPtr *acl_message*) [private]

12.73.4.9 static internal int LibMC::MCAgency::_MC_AclSetContent (IntPtr *acl*, String *content*) [private]

12.73.4.10 static internal int LibMC::MCAgency::_MC_AclSetPerformative (IntPtr *acl*, MCAclMessage.MC_FipaPerformative *e_performative*) [private]

12.73.4.11 static internal int LibMC::MCAgency::_MC_AclSetSender (IntPtr *acl*, String *name*, String *address*) [private]

12.73.4.12 static internal IntPtr LibMC::MCAgency::_MC_AclWaitRetrieve (IntPtr *agent*) [private]

12.73.4.13 static int LibMC::MCAgency::_MC_AddAgent (IntPtr *agency*, IntPtr *agent*) [private]

Referenced by `AddAgent()`.

12.73.4.14 `static int LibMC::MCAgency::_MC_BarrierDelete (IntPtr agency, int id)
[private]`

Referenced by BarrierDelete().

12.73.4.15 `static int LibMC::MCAgency::_MC_BarrierInit (IntPtr agency, int id, int num_procs)
[private]`

Referenced by BarrierInit().

12.73.4.16 `static internal int LibMC::MCAgency::_MC_CallAgentFunc (IntPtr agent, String
funcName, IntPtr returnVal, IntPtr varg) [private]`

12.73.4.17 `static int LibMC::MCAgency::_MC_ChInitializeOptions (IntPtr agency, ChOptions_t
options) [private]`

Referenced by ChInitializeOptions().

12.73.4.18 `static int LibMC::MCAgency::_MC_CondBroadcast (IntPtr agency, int id)
[private]`

Referenced by CondBroadcast().

12.73.4.19 `static int LibMC::MCAgency::_MC_CondReset (IntPtr agency, int id) [private]`

Referenced by CondReset().

12.73.4.20 `static int LibMC::MCAgency::_MC_CondSignal (IntPtr agency, int id) [private]`

Referenced by CondSignal().

12.73.4.21 `static int LibMC::MCAgency::_MC_CondWait (IntPtr agency, int id) [private]`

Referenced by CondWait().

12.73.4.22 `static internal int LibMC::MCAgency::_MC_DeleteAgent (IntPtr agent) [private]`

12.73.4.23 `static int LibMC::MCAgency::_MC_End (IntPtr agency) [private]`

12.73.4.24 `static IntPtr LibMC::MCAgency::_MC_FindAgentByID (IntPtr agency, int ID)
[private]`

Referenced by FindAgentByID().

12.73.4.25 `static IntPtr LibMC::MCAgency::_MC_FindAgentByName (IntPtr agency, String
name) [private]`

Referenced by FindAgentByName().

- 12.73.4.26 **static internal IntPtr LibMC::MCAgency::_MC_GetAgentExecEngine (IntPtr *agent*)**
[private]
- 12.73.4.27 **static internal int LibMC::MCAgency::_MC_GetAgentID (IntPtr *agent*)** [private]
- 12.73.4.28 **static internal String LibMC::MCAgency::_MC_GetAgentName (IntPtr *agent*)**
[private]
- 12.73.4.29 **static internal int LibMC::MCAgency::_MC_GetAgentNumTasks (IntPtr *agent*)**
[private]
- 12.73.4.30 **static internal int LibMC::MCAgency::_MC_GetAgentReturnData (IntPtr *agent*, int**
***task_num*, IntPtr *data*, IntPtr *dim*, IntPtr *extent*)** [private]
- 12.73.4.31 **static internal MCAgent.MC_AgentStatus_e LibMC::MCAgency::_MC_-**
GetAgentStatus (IntPtr *agent*) [private]
- 12.73.4.32 **static internal MCAgent.MC_AgentType_e LibMC::MCAgency::_MC_GetAgentType**
(IntPtr *agent*) [private]
- 12.73.4.33 **static internal String LibMC::MCAgency::_MC_GetAgentXMLString (IntPtr *agent*)**
[private]
- 12.73.4.34 **static int LibMC::MCAgency::_MC_HaltAgency (IntPtr *agency*)** [private]

Referenced by HaltAgency().

- 12.73.4.35 **static IntPtr LibMC::MCAgency::_MC_Initialize (int *port*, ref MCAgencyOptions_t**
***options*)** [private]
- 12.73.4.36 **static int LibMC::MCAgency::_MC_InitializeAgencyOptions (ref**
MCAgencyOptions_t *options*) [private]

Referenced by MCAgency().

- 12.73.4.37 **static int LibMC::MCAgency::_MC_MainLoop (IntPtr *agency*)** [private]

Referenced by MainLoop().

- 12.73.4.38 **static int LibMC::MCAgency::_MC_MutexLock (IntPtr *agency*, int *id*)** [private]

Referenced by MutexLock().

- 12.73.4.39 **static int LibMC::MCAgency::_MC_MutexUnlock (IntPtr *agency*, int *id*)**
[private]

Referenced by MutexUnlock().

12.73.4.40 `static internal int LibMC::MCAgency::_MC_PrintAgentCode (IntPtr agent)
[private]`

12.73.4.41 `static int LibMC::MCAgency::_MC_RegisterService (IntPtr agency, IntPtr agent, int
agentID, String agentName, String[] serviceNames, int numServices) [private]`

Referenced by RegisterService().

12.73.4.42 `static int LibMC::MCAgency::_MC_ResetSignal (IntPtr agency) [private]`

Referenced by ResetSignal().

12.73.4.43 `static int LibMC::MCAgency::_MC_ResumeAgency (IntPtr agency) [private]`

Referenced by ResumeAgency().

12.73.4.44 `static IntPtr LibMC::MCAgency::_MC_RetrieveAgent (IntPtr agency) [private]`

Referenced by RetrieveAgent().

12.73.4.45 `static internal String LibMC::MCAgency::_MC_RetrieveAgentCode (IntPtr agent)
[private]`

12.73.4.46 `static int LibMC::MCAgency::_MC_SearchForService (IntPtr agency, String
searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr
numResults) [private]`

12.73.4.47 `static int LibMC::MCAgency::_MC_SemaphorePost (IntPtr agency, int id)
[private]`

Referenced by SemaphorePost().

12.73.4.48 `static int LibMC::MCAgency::_MC_SemaphoreWait (IntPtr agency, int id)
[private]`

Referenced by SemaphoreWait().

12.73.4.49 `static int LibMC::MCAgency::_MC_SendAgentMigrationMessage (IntPtr agency,
String message, String hostname, int port) [private]`

Referenced by SendAgentMigrationMessage().

12.73.4.50 `static int LibMC::MCAgency::_MC_SendAgentMigrationMessageFile (IntPtr agency, String filename, String hostname, int port) [private]`

12.73.4.51 `static internal int LibMC::MCAgency::_MC_SetAgentStatus (IntPtr agent, MCAgent.MC_AgentStatus_e status) [private]`

12.73.4.52 `static int LibMC::MCAgency::_MC_SetDefaultAgentStatus (IntPtr agency, MCAgent.MC_AgentStatus_e status) [private]`

Referenced by SetDefaultAgentStatus().

12.73.4.53 `static int LibMC::MCAgency::_MC_SetThreadOff (ref MCAgencyOptions_t options, MC_ThreadIndex_e index) [private]`

Referenced by SetThreadOff().

12.73.4.54 `static int LibMC::MCAgency::_MC_SetThreadOn (ref MCAgencyOptions_t options, MC_ThreadIndex_e index) [private]`

Referenced by SetThreadOn().

12.73.4.55 `static int LibMC::MCAgency::_MC_SetThreadsAllOff (ref MCAgencyOptions_t options) [private]`

Referenced by SetThreadsAllOff().

12.73.4.56 `static int LibMC::MCAgency::_MC_SetThreadsAllOn (ref MCAgencyOptions_t options) [private]`

Referenced by SetThreadsAllOn().

12.73.4.57 `static int LibMC::MCAgency::_MC_Steer (IntPtr agency, IntPtr funcptr, IntPtr arg) [private]`

12.73.4.58 `static MC_SteerCommand_e LibMC::MCAgency::_MC_SteerControl () [private]`

Referenced by SteerControl().

12.73.4.59 `static int LibMC::MCAgency::_MC_SyncDelete (IntPtr agency, int id) [private]`

Referenced by SyncDelete().

12.73.4.60 `static int LibMC::MCAgency::_MC_SyncInit (IntPtr agency, int id) [private]`

Referenced by SyncInit().

12.73.4.61 `static internal int LibMC::MCAgency::_MC_TerminateAgent (IntPtr agent)`
[private]

12.73.4.62 `static int LibMC::MCAgency::_MC_WaitAgent (IntPtr agency)` [private]

Referenced by WaitAgent().

12.73.4.63 `static IntPtr LibMC::MCAgency::_MC_WaitRetrieveAgent (IntPtr agency)`
[private]

Referenced by WaitRetrieveAgent().

12.73.4.64 `static int LibMC::MCAgency::_MC_WaitSignal (IntPtr agency, int signals)`
[private]

Referenced by WaitSignal().

12.73.4.65 `int LibMC::MCAgency::AclSend (MCAclMessage acl_message)` [inline]

Send an ACL message to the agency. Sends an ACL message to the agency. The message is delivered appropriately.

Parameters:

acl_message The message to send.

Returns:

The return value of the underlying MC_CondBroadcast function.

Definition at line 1048 of file MCAgency.cs.

References LibMC::MCAclMessage::AclMsg, and Agency.

12.73.4.66 `int LibMC::MCAgency::AddAgent (MCAgent agent)` [inline]

Add an agent to the agency. Adds an agent to the agency.

Parameters:

agent The agent to add.

Returns:

The return value of the underlying MC_AddAgent function.

Definition at line 993 of file MCAgency.cs.

References _MC_AddAgent(), Agency, and LibMC::MCAgent::Agent.

12.73.4.67 int LibMC::MCAgency::BarrierDelete (int *id*) [inline]

Delete a barrier object. Deletes a barrier object from the agency. The parameter "id" is the ID of the agency sync variable created with [BarrierInit\(\)](#).

Parameters:

id The ID number of the barrier to delete.

Returns:

The return value of the underlying MC_BarrierDelete function.

Definition at line 877 of file MCAgency.cs.

References `_MC_BarrierDelete()`, and `Agency`.

12.73.4.68 int LibMC::MCAgency::BarrierInit (int *id*, int *num_procs*) [inline]

Create a new barrier. Creates a new barrier object in the agency.

Parameters:

id The ID number of the condition to signal.

num_procs the number of process to block (?)

Returns:

The return value of the underlying MC_BarrierInit function.

Definition at line 891 of file MCAgency.cs.

References `_MC_BarrierInit()`, and `Agency`.

12.73.4.69 int LibMC::MCAgency::ChInitializeOptions (ChShellType *shellType*, String *home*) [inline]

Initializes Ch [options](#) for the agency. Can be used to set the home directory and shell mode for the Ch interpreter.

Parameters:

shellType The type of shell Ch should use: CH_REGULARARCH or CH_SAFECH.

home The home directory Ch should use.

Returns:

The return value of the underlying MC_ChInitializeOptions function.

Note:

This function must be called before the agency is started.

Definition at line 499 of file MCAgency.cs.

References `_MC_ChInitializeOptions()`, `Agency`, `LibMC::MCAgency::ChOptions_t::chhome`, `int`, and `LibMC::MCAgency::ChOptions_t::shelltype`.

12.73.4.70 int LibMC::MCAgency::CondBroadcast (int *id*) [inline]

Broadcast a condition signal. Broadcasts a signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying MC_CondBroadcast function.

Definition at line 705 of file MCAgency.cs.

References [_MC_CondBroadcast\(\)](#), and [Agency](#).

12.73.4.71 int LibMC::MCAgency::CondReset (int *id*) [inline]

Reset a condition signal. Resets a signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function must be called after a condition is received in order to clear it.

Parameters:

id The ID number of the condition to reset.

Returns:

The return value of the underlying MC_CondReset function.

Definition at line 734 of file MCAgency.cs.

References [_MC_CondReset\(\)](#), and [Agency](#).

12.73.4.72 int LibMC::MCAgency::CondSignal (int *id*) [inline]

Signal a condition. Signals a condition in the agency. The parameter "id" is the ID of the agency sync variable to signal that was created with [SyncInit\(\)](#).

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying MC_CondSignal function.

Definition at line 719 of file MCAgency.cs.

References [_MC_CondSignal\(\)](#), and [Agency](#).

12.73.4.73 int LibMC::MCAgency::CondWait (int *id*) [inline]

Wait for a condition signal. Waits for a condition signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function blocks until the signal is received.

Parameters:

id The ID number of the condition to wait for.

Returns:

The return value of the underlying MC_CondWait function.

Definition at line 749 of file MCAgency.cs.

References _MC_CondWait(), and Agency.

12.73.4.74 int LibMC::MCAgency::End () [inline]

Stops and destroys the agency. Stops the agency and sets the agency state appropriately.

Returns:

The return value of the underlying MC_End function.

Note:

This call will fail if the underlying Mobile-C agency is not in the correct state.

Definition at line 481 of file MCAgency.cs.

References Agency, and state.

12.73.4.75 MCAgent LibMC::MCAgency::FindAgentByID (int *id*) [inline]

Find an agent by its ID. Finds an agent in the agency by its ID number.

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying MC_FindAgentByID function.

Definition at line 1019 of file MCAgency.cs.

References _MC_FindAgentByID(), and Agency.

12.73.4.76 MCAgent LibMC::MCAgency::FindAgentByName (String *name*) [inline]

Finds an agent by its name. Finds an agent in the agency by its name.

Parameters:

name The name of the agent to search for.

Returns:

The return value of the underlying MC_FindAgentByName function.

Definition at line 1006 of file MCAgency.cs.

References _MC_FindAgentByName(), and Agency.

12.73.4.77 int LibMC::MCAgency::HaltAgency () [inline]

Temporarily halts the agency. Halts the agency until it is resumed or ended.

Returns:

The return value of the underlying MC_HaltAgency function.

Note:

The underlying Mobile-C agency must be in the correct state to call this function or it will fail.

Definition at line 576 of file MCAgency.cs.

References `_MC_HaltAgency()`, `Agency`, and `state`.

12.73.4.78 int LibMC::MCAgency::Initialize () [inline]

Starts the agency. Starts the agency and sets the agency state.

Returns:

0 on success, -1 on failure.

Note:

The agency port and any other [options](#) must be set before calling this function.

Definition at line 459 of file MCAgency.cs.

References `Agency`, `port`, and `state`.

12.73.4.79 int LibMC::MCAgency::LoadAgentMigrationMessageFile (String filename) [inline]

Load an agent migration message. Loads the specified XML file to this agency automatically. There is no need to specify a port or agency location.

Parameters:

filename The name of the file to send (fully qualified).

Returns:

The return value of the underlying MC_SendAgentMigrationMessageFile function.

Definition at line 672 of file MCAgency.cs.

References `Agency`, and `Port`.

12.73.4.80 int LibMC::MCAgency::MainLoop () [inline]

Makes the agency wait indefinitely. Makes the agency wait indefinitely until it receives a "quit" command or is otherwise terminated.

Returns:

The return value of the underlying MC_MainLoop function.

Definition at line 1061 of file MCAgency.cs.

References _MC_MainLoop(), and Agency.

12.73.4.81 int LibMC::MCAgency::MutexLock (int *id*) [inline]

Lock a mutex. Locks a mutex in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function blocks until the mutex is locked.

Parameters:

id The ID number of the mutex to lock.

Returns:

The return value of the underlying MC_MutexLock function.

Definition at line 764 of file MCAgency.cs.

References _MC_MutexLock(), and Agency.

12.73.4.82 int LibMC::MCAgency::MutexUnlock (int *id*) [inline]

Unlock a mutex. Locks a mutex in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the mutex to unlock.

Returns:

The return value of the underlying MC_MutexUnlock function.

Definition at line 778 of file MCAgency.cs.

References _MC_MutexUnlock(), and Agency.

12.73.4.83 int LibMC::MCAgency::RegisterService (MCAgent *agent*, int *agentID*, String *agentName*, String[] *serviceNames*, int *numServices*) [inline]

Registers services in the agency. Registers services provided by agents with the agency. Not really useful in binary space.

Parameters:

agent The agent providing the services.

agentID The agent ID number.

agentName The agent name.

serviceNames An array of service names.

numServices The number of services provided.

Returns:

The return value of the underlying MC_RegisterService function.

Todo

Test MC_RegisterService and MC_SearchForService.

Definition at line 952 of file MCAgency.cs.

References `_MC_RegisterService()`, `Agency`, and `LibMC::MCAgent::Agent`.

12.73.4.84 int LibMC::MCAgency::ResetSignal () [inline]

Reset an agency signal. Resets a signal in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Returns:

The return value of the underlying MC_ResetSignal function.

Definition at line 820 of file MCAgency.cs.

References `_MC_ResetSignal()`, and `Agency`.

12.73.4.85 int LibMC::MCAgency::ResumeAgency () [inline]

Resumes a halted agency. Resumes a halted agency. Cannot be used on ended agencies.

Returns:

The return value of the underlying MC_ResumeAgency function.

Note:

The underlying Mobile-C agency must be in the correct state to call this function or it will fail.

Definition at line 593 of file MCAgency.cs.

References `_MC_ResumeAgency()`, `Agency`, and `state`.

12.73.4.86 MCAgent LibMC::MCAgency::RetrieveAgent () [inline]

Retrieve an agent from the agency. Really not sure.

Returns:

The return value of the underlying MC_CondBroadcast function.

Definition at line 1031 of file MCAgency.cs.

References `_MC_RetrieveAgent()`, and `Agency`.

12.73.4.87 `int LibMC::MCAgency::SearchForService (String searchString, IntPtr agentNames, IntPtr serviceNames, IntPtr agentIDs, IntPtr numResults) [inline]`

Searches for services in the agency. Searches for services provided by agents with the agency. Not really useful in binary space.

Parameters:

searchString The agent providing the services.

agentNames The agent ID number.

serviceNames The agent name.

agentIDs An array of service names.

numResults The number of services provided.

Returns:

The return value of the underlying MC_SearchForService function.

Note:

This function does nothing but throw an exception right now.

Bug

MC_SearchForService is not yet implemented.

Todo

Implement SearchForService

Definition at line 975 of file MCAgency.cs.

12.73.4.88 `int LibMC::MCAgency::SemaphorePost (int id) [inline]`

Posts a semaphore. Posts a semaphore in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the semaphore to post.

Returns:

The return value of the underlying MC_SemaphorePost function.

Definition at line 792 of file MCAgency.cs.

References `_MC_SemaphorePost()`, and `Agency`.

12.73.4.89 `int LibMC::MCAgency::SemaphoreWait (int id) [inline]`

Wait for a semaphore to be posted. Wait for a semaphore in the agency to be posted. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#). This function blocks until the semaphore is posted.

Parameters:

id The ID number of the semaphore to wait for.

Returns:

The return value of the underlying MC_SemaphoreWait function.

Definition at line 807 of file MCAgency.cs.

References _MC_SemaphoreWait(), and Agency.

12.73.4.90 int LibMC::MCAgency::SendAgentMigrationMessage (String message, String hostname, int port) [inline]

Sends an agent migration message to an agency. Sends an agent migration message to another agency (local or remote).

Parameters:

message The agent migration message.

hostname The URL, IP address, or other identifier for the agency host.

port The port to send to.

Returns:

The return value of the underlying MC_SendAgentMigrationMessageFile function.

Definition at line 687 of file MCAgency.cs.

References _MC_SendAgentMigrationMessage(), and Agency.

12.73.4.91 int LibMC::MCAgency::SendAgentMigrationMessageFile (String filename, String hostname, int port) [inline]

Sends an agent migration message file to an agency. Sends the specified XML file to another agency (local or remote).

Parameters:

filename The name of the file to send (fully qualified).

hostname The URL, IP address, or other identifier for the agency host.

port The port to send to.

Returns:

The return value of the underlying MC_SendAgentMigrationMessageFile function.

Definition at line 658 of file MCAgency.cs.

References Agency.

12.73.4.92 `int LibMC::MCAgency::SetDefaultAgentStatus (MCAgent.MC_AgentStatus_e status) [inline]`

Sets the default state of an agent in the agency. Can be used to set the default status of agents, but most agents managed their state on their own.

Parameters:

status The enum that identifies the desired agent state.

Returns:

The return value of the underlying MC_SetDefaultAgentStatus function.

Definition at line 609 of file MCAgency.cs.

References `_MC_SetDefaultAgentStatus()`, and `Agency`.

12.73.4.93 `int LibMC::MCAgency::SetThreadOff (MC_ThreadIndex_e index) [inline]`

Sets an individual thread for the agency to "off.". Most commonly used to turn the command prompt thread off.

Parameters:

index The enum that identifies the thread to be turned off.

Returns:

The return value of the underlying MC_SetThreadOff function.

Note:

This function must be called before the agency is started.

Definition at line 561 of file MCAgency.cs.

References `_MC_SetThreadOff()`.

12.73.4.94 `int LibMC::MCAgency::SetThreadOn (MC_ThreadIndex_e index) [inline]`

Sets an individual thread for the agency to "on.". Threads are on by default. If they have been turned off, this function turns them on again.

Parameters:

index The enum that identifies the thread to be turned on.

Returns:

The return value of the underlying MC_SetThreadOn function.

Note:

This function must be called before the agency is started.

Definition at line 546 of file MCAgency.cs.

References `_MC_SetThreadOn()`.

12.73.4.95 int LibMC::MCAgency::SetThreadsAllOff () [inline]

Sets all threads for the agency to "off.". Sets all threads for the agency to "off." Not recommended for use.

Returns:

The return value of the underlying MC_SetThreadsAllOff function.

Note:

This function must be called before the agency is started.

Definition at line 530 of file MCAgency.cs.

References `_MC_SetThreadsAllOff()`.

12.73.4.96 int LibMC::MCAgency::SetThreadsAllOn () [inline]

Definition at line 516 of file MCAgency.cs.

References `_MC_SetThreadsAllOn()`.

12.73.4.97 int LibMC::MCAgency::Steer (IntPtr funcptr, IntPtr arg) [inline]

Steering control function. Really not sure.

Parameters:

funcptr Pointer to the steering function

arg Argument to function

Returns:

The return value of the underlying `_MC_Steer` function.

Note:

This function does nothing but throw an exception right now.

Bug

`MC_Steer` is not yet implemented.

Todo

Implement `MC_Steer`

Definition at line 927 of file MCAgency.cs.

12.73.4.98 MC_SteerCommand_e LibMC::MCAgency::SteerControl () [inline]

Steering control function. Really not sure.

Returns:

The return value of the underlying `MC_SteerControl` function.

Todo

Test MC_SteerControl, MC_Steer.

Definition at line 909 of file MCAgency.cs.

References _MC_SteerControl().

12.73.4.99 int LibMC::MCAgency::SyncDelete (int *id*) [inline]

Delete a synchronization variable. Deletes a synchronization variable in the agency. The parameter "id" is the ID of the agency sync variable created with [SyncInit\(\)](#).

Parameters:

id The ID number of the variable to delete.

Returns:

The return value of the underlying MC_SyncDelete function.

Definition at line 834 of file MCAgency.cs.

References _MC_SyncDelete(), and Agency.

12.73.4.100 int LibMC::MCAgency::SyncInit (int *id*) [inline]

Create a new synchronization variable. Creates a new synchronization variable in the agency. The parameter "id" is desired ID of the variable. A random ID is returned if "id" is already in use.

Parameters:

id The ID number of the condition to signal.

Returns:

The return value of the underlying MC_CondBroadcast function- either a random ID or the desired ID if the desired ID is already in use.

Definition at line 850 of file MCAgency.cs.

References _MC_SyncInit(), and Agency.

12.73.4.101 int LibMC::MCAgency::WaitAgent () [inline]

Waits for an agent to arrive. Waits for an agent to arrive in the agency. The agent is allowed to execute normally.

Returns:

The return value of the underlying MC_WaitAgent function.

Definition at line 639 of file MCAgency.cs.

References _MC_WaitAgent(), and Agency.

12.73.4.102 MCAgent LibMC::MCAgency::WaitRetrieveAgent () [inline]

Waits for an agent to arrive and returns the agent. Waits for an agent to arrive in the agency, then returns that agent. The agent is not allowed to execute.

Returns:

The agent that was retrieved or an empty agent if it fails.

Definition at line 622 of file MCAgency.cs.

References `_MC_WaitRetrieveAgent()`, and `Agency`.

12.73.4.103 int LibMC::MCAgency::WaitSignal (MC_Signal_e signals) [inline]

Wait for agency signals. Waits for signals to occur in the agency.

Parameters:

signals The ID number of the condition to signal.

Returns:

The return value of the underlying `MC_WaitSignal` function.

Definition at line 863 of file MCAgency.cs.

References `_MC_WaitSignal()`, and `Agency`.

12.73.5 Field Documentation

12.73.5.1 IntPtr LibMC::MCAgency::agency_p = IntPtr.Zero [private]

Definition at line 369 of file MCAgency.cs.

12.73.5.2 const String LibMC::MCAgency::mcdll [private]

Initial value:

```
"libmc.dll"
```

Definition at line 15 of file MCEExports.cs.

12.73.5.3 MCAgencyOptions_t LibMC::MCAgency::options [private]

Definition at line 370 of file MCAgency.cs.

12.73.5.4 `int LibMC::MCAgency::port = -1` `[private]`

Definition at line 371 of file MCAgency.cs.

Referenced by Initialize().

12.73.5.5 `MCAgencyState LibMC::MCAgency::state = MCAgencyState.NoState` `[private]`

Definition at line 372 of file MCAgency.cs.

Referenced by End(), HaltAgency(), Initialize(), MCAgency(), and ResumeAgency().

12.73.6 Property Documentation

12.73.6.1 `IntPtr LibMC::MCAgency::Agency` `[get, set, private]`

Definition at line 389 of file MCAgency.cs.

Referenced by AclSend(), AddAgent(), BarrierDelete(), BarrierInit(), ChInitializeOptions(), CondBroadcast(), CondReset(), CondSignal(), CondWait(), End(), FindAgentByID(), FindAgentByName(), HaltAgency(), Initialize(), LoadAgentMigrationMessageFile(), MainLoop(), MutexLock(), MutexUnlock(), RegisterService(), ResetSignal(), ResumeAgency(), RetrieveAgent(), SemaphorePost(), SemaphoreWait(), SendAgentMigrationMessage(), SendAgentMigrationMessageFile(), SetDefaultAgentStatus(), SyncDelete(), SyncInit(), WaitAgent(), WaitRetrieveAgent(), and WaitSignal().

12.73.6.2 `int LibMC::MCAgency::Port` `[get, set]`

Accessor for the port number of the agency. Allows the user to set the agency port or get the port number while it is running.

Note:

The port must be set before the agency is started. Once the agency is started, the port cannot be changed.

Definition at line 418 of file MCAgency.cs.

Referenced by LoadAgentMigrationMessageFile().

12.73.6.3 `MCAgencyState LibMC::MCAgency::State` `[get]`

Accessor for the agency state. Allows the user to query the state of the agency.

Note:

The state cannot be set by the user. It is controlled internally.

Definition at line 438 of file MCAgency.cs.

The documentation for this class was generated from the following files:

- [/home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgency.cs](#)
- [/home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExports.cs](#)

12.74 LibMC::MCAgency::MCAgency_t Struct Reference

Data Fields

- [int client](#)
- [int server](#)
- [String hostName](#)
- [String filename](#)
- [int portno](#)
- [int portnoc](#)
- [IntPtr mc_platform](#)
- [int default_agentstatus](#)
- [int threads](#)
- [int enable_security](#)
- [int\[\] stack_size](#)
- [int last_error](#)

12.74.1 Detailed Description

Definition at line 23 of file MCExports.cs.

12.74.2 Field Documentation

12.74.2.1 `int LibMC::MCAgency::MCAgency_t::client`

Definition at line 25 of file MCExports.cs.

12.74.2.2 `int LibMC::MCAgency::MCAgency_t::default_agentstatus`

Agency default agent status

Definition at line 32 of file MCExports.cs.

12.74.2.3 `int LibMC::MCAgency::MCAgency_t::enable_security`

Security flag

Definition at line 34 of file MCExports.cs.

12.74.2.4 `String LibMC::MCAgency::MCAgency_t::filename`

Definition at line 28 of file MCExports.cs.

12.74.2.5 `String LibMC::MCAgency::MCAgency_t::hostName`

Local Hostname

Definition at line 27 of file MCExports.cs.

12.74.2.6 int LibMC::MCAgency::MCAgency_t::last_error

Definition at line 38 of file MCEExports.cs.

12.74.2.7 IntPtr LibMC::MCAgency::MCAgency_t::mc_platform

Local MobileC Platform

Definition at line 31 of file MCEExports.cs.

12.74.2.8 int LibMC::MCAgency::MCAgency_t::portno

Local port number

Definition at line 29 of file MCEExports.cs.

12.74.2.9 int LibMC::MCAgency::MCAgency_t::portnoc

Definition at line 30 of file MCEExports.cs.

12.74.2.10 int LibMC::MCAgency::MCAgency_t::server

Definition at line 26 of file MCEExports.cs.

12.74.2.11 int [] LibMC::MCAgency::MCAgency_t::stack_size

Definition at line 37 of file MCEExports.cs.

12.74.2.12 int LibMC::MCAgency::MCAgency_t::threads

flag which determines which threads to start

Definition at line 33 of file MCEExports.cs.

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs>

12.75 MCAgencyOptions_s Struct Reference

User modifiable agency [options](#).

```
#include <libmc.h>
```

Data Fields

- [int threads](#)
- [int default_agent_status](#)
- [int modified](#)
- [int enable_security](#)
- unsigned char [passphrase](#) [32]
- [int stack_size](#) [MC_THREAD_ALL]
- char * [known_host_filename](#)
- char * [priv_key_filename](#)
- [int initInterps](#)
- ChOptions_t * [ch_options](#)

12.75.1 Detailed Description

User modifiable agency [options](#).

Definition at line 248 of file libmc.h.

12.75.2 Field Documentation

12.75.2.1 ChOptions_t* MCAgencyOptions_s::ch_options

Definition at line 261 of file libmc.h.

Referenced by MC_Initialize().

12.75.2.2 int MCAgencyOptions_s::default_agent_status

Default agent status

Definition at line 250 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

12.75.2.3 int MCAgencyOptions_s::enable_security

security enable flag

Definition at line 252 of file libmc.h.

12.75.2.4 int MCAgencyOptions_s::initInterps

Definition at line 260 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

12.75.2.5 char* MCAgencyOptions_s::known_host_filename

Definition at line 258 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

12.75.2.6 int MCAgencyOptions_s::modified

unused

Definition at line 251 of file libmc.h.

Referenced by MC_InitializeAgencyOptions().

12.75.2.7 unsigned char MCAgencyOptions_s::passphrase[32]

security enable flag

Definition at line 253 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

12.75.2.8 char* MCAgencyOptions_s::priv_key_filename

Definition at line 259 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

12.75.2.9 int MCAgencyOptions_s::stack_size[MC_THREAD_ALL]

If the stack size is -1, use system def.

Definition at line 256 of file libmc.h.

Referenced by MC_Initialize(), and MC_InitializeAgencyOptions().

12.75.2.10 int MCAgencyOptions_s::threads

Threads to start

Definition at line 249 of file libmc.h.

Referenced by MC_Initialize(), MC_InitializeAgencyOptions(), MC_SetThreadOff(), MC_SetThreadOn(), MC_SetThreadsAllOff(), and MC_SetThreadsAllOn().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[libmc.h](#)

12.76 LibMC::MCAgency::MCAgencyOptions_t Struct Reference

Data Fields

- [int threads](#)
- [int default_agent_status](#)
- [int modified](#)
- [int enable_security](#)
- [int\[\] stack_size](#)

12.76.1 Detailed Description

Definition at line 42 of file MCEExports.cs.

12.76.2 Field Documentation

12.76.2.1 int LibMC::MCAgency::MCAgencyOptions_t::default_agent_status

Default agent status

Definition at line 45 of file MCEExports.cs.

12.76.2.2 int LibMC::MCAgency::MCAgencyOptions_t::enable_security

security enable flag

Definition at line 47 of file MCEExports.cs.

12.76.2.3 int LibMC::MCAgency::MCAgencyOptions_t::modified

unused

Definition at line 46 of file MCEExports.cs.

12.76.2.4 int [] LibMC::MCAgency::MCAgencyOptions_t::stack_size

If the stack size is -1, use system def.

Definition at line 50 of file MCEExports.cs.

12.76.2.5 int LibMC::MCAgency::MCAgencyOptions_t::threads

Threads to start

Definition at line 44 of file MCEExports.cs.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCEExports.cs](#)

12.77 LibMC::MCAgent Class Reference

Wrapper class for MCAgent_t structure.

Public Types

- enum [MC_AgentType_e](#) { [MC_NONE](#) = -1, [MC_REMOTE_AGENT](#) = 0, [MC_LOCAL_AGENT](#), [MC_RETURN_AGENT](#) }
Enum for describing the type of an agent.
- enum [MC_AgentStatus_e](#) {
[MC_NO_STATUS](#) = -1, [MC_WAIT_CH](#) = 0, [MC_WAIT_MESSGSEND](#), [MC_AGENT_ACTIVE](#),
[MC_AGENT_NEUTRAL](#), [MC_AGENT_SUSPENDED](#), [MC_WAIT_FINISHED](#) }
Enum for describing the status of an agent.

Public Member Functions

- [MCAgent](#) ()
Default constructor.
- override string [ToString](#) ()
Display the agent's fields.
- int [DeleteAgent](#) ()
Deletes an agent.
- String [GetAgentXMLString](#) ()
Gets the agent's XML string.
- int [PrintAgentCode](#) ()
Gets the agent's C code string.
- String [RetrieveAgentCode](#) ()
Gets the agent's C code string.
- int [TerminateAgent](#) ()
Terminates an agent.
- int [AclPost](#) ([MCAclMessage](#) message)
Posts an ACL message to the agent.
- [MCAclMessage](#) [AclRetrieve](#) ()
Retrieve an ACL message from the agent.
- [MCAclMessage](#) [AclWaitRetrieve](#) ()
Wait for and retrieve an ACL message from the agent.

- [int CallAgentFunc](#) (String funcName, IntPtr retval, IntPtr varg)
Calls a function in an agent script.
- [int CallAgentFunc](#) (String funcName, ref object retval, ref object varg)
Calls a function in an agent script.
- IntPtr [GetAgentExecEngine](#) ()
Gets an agent's Ch interpreter.
- [int GetAgentReturnData](#) (int task_num, IntPtr data, IntPtr dim, IntPtr extent)
Calls a function in an agent script.

Static Public Member Functions

- static implicit [operator IntPtr](#) (MCAGENT agent)
- static implicit [operator MCAGENT](#) (IntPtr ip)

Properties

- internal IntPtr [Agent](#) [get, set]
- [int AgentID](#) [get]
Gets the agent's ID number.
- String [AgentName](#) [get]
Gets the agent's name.
- [int AgentNumTasks](#) [get]
Gets the agent's number of tasks.
- [MC_AgentStatus_e AgentStatus](#) [get, set]
Gets or sets the agent's status.
- [MC_AgentType_e AgentType](#) [get]
Gets the agent's type.
- bool [Valid](#) [get]
Checks whether the agent is valid.

Private Member Functions

- internal [MCAGENT](#) (IntPtr ip)
- void [GetAgentFields](#) ()

Private Attributes

- IntPtr `agent_p`
- String `name` = ""
- int `id` = -1
- int `numTasks` = -1
- `MC_AgentStatus_e status` = `MC_AgentStatus_e.MC_NO_STATUS`
- `MC_AgentType_e type` = `MC_AgentType_e.MC_NONE`

12.77.1 Detailed Description

Wrapper class for `MCAgent_t` structure. This class provides an interface to the Mobile-C agent structure. Member functions for the class are generally overloaded versions of the respective functions in the Mobile-C library. The class maintains a pointer to a Mobile-C agent in unmanaged memory. The pointer is not accessible by the user.

Definition at line 61 of file `MCAgent.cs`.

12.77.2 Member Enumeration Documentation

12.77.2.1 enum `LibMC::MCAgent::MC_AgentStatus_e`

Enum for describing the status of an agent.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

`MC_NO_STATUS` Default value for uninitialized agent
`MC_WAIT_CH` Waiting to be started
`MC_WAIT_MESSGSEND` Finished, waiting to migrate
`MC_AGENT_ACTIVE` Running
`MC_AGENT_NEUTRAL` Not running, but do not flush
`MC_AGENT_SUSPENDED` Unused
`MC_WAIT_FINISHED` Finished, waiting to be flushed

Definition at line 88 of file `MCAgent.cs`.

12.77.2.2 enum `LibMC::MCAgent::MC_AgentType_e`

Enum for describing the type of an agent.

Note:

This enum is pulled directly from the Mobile-C library.

Enumerator:

`MC_NONE` Default value to describe unininitialized agent.

MC_REMOTE_AGENT A remote agent.

MC_LOCAL_AGENT A local agent.

MC_RETURN_AGENT A returning agent.

Definition at line 75 of file MCAgent.cs.

12.77.3 Constructor & Destructor Documentation

12.77.3.1 LibMC::MCAgent::MCAgent () [inline]

Default constructor. Creates an empty agent.

Definition at line 104 of file MCAgent.cs.

Referenced by operator MCAgent().

12.77.3.2 internal LibMC::MCAgent::MCAgent (IntPtr ip) [inline, private]

Definition at line 110 of file MCAgent.cs.

References Agent.

12.77.4 Member Function Documentation

12.77.4.1 int LibMC::MCAgent::AclPost (MCAclMessage message) [inline]

Posts an ACL message to the agent. Delivers an ACL message to the agent.

Parameters:

message The ACL message object to deliver.

Returns:

The return value of the underlying MC_AclPost function call.

Note:

The message must be a valid message or this function call will fail.

Definition at line 397 of file MCAgent.cs.

References LibMC::MCAclMessage::AclMsg, and Agent.

12.77.4.2 MCAclMessage LibMC::MCAgent::AclRetrieve () [inline]

Retrieve an ACL message from the agent. Retrieves an ACL message from the agent if one is available.

Returns:

The ACL message or a blank ACL message if one was not available.

Note:

The message must be a valid message or this function call will fail.

Definition at line 413 of file MCAgent.cs.

References Agent.

12.77.4.3 MCAclMessage LibMC::MCAgent::AclWaitRetrieve () [inline]

Wait for and retrieve an ACL message from the agent. Retrieves an ACL message from the agent when one becomes available.

Returns:

The ACL message or a blank ACL message if the call fails.

Note:

This function call blocks.

Definition at line 432 of file MCAgent.cs.

References Agent.

12.77.4.4 int LibMC::MCAgent::CallAgentFunc (String *funcName*, ref object *retval*, ref object *varg*) [inline]

Calls a function in an agent script. Calls a function in an agent's script file. This function requires boxing of parameters, but no marshaling.

Parameters:

funcName The name of the function to call
retval A boxed object to hold the return value
varg The boxed argument to the agent function

Returns:

The return value of the underlying MC_CallAgentFunc function call.

Note:

This function handles marshaling of the argument and return value. The memory provided to the agent function for both *retval* and *varg* is not preserved after this function call! If the memory is to be kept by the agent, use the manually marshaled version of this function. Also note that even though structures can be marshaled automatically, in this function, the type of the structure is unknown and therefore it must be handled manually, even though the marshaling is transparent to the user.

See also:

Overloaded [CallAgentFunc](#), LibMCConsole example

Definition at line 488 of file MCAgent.cs.

References CallAgentFunc().

12.77.4.5 int LibMC::MCAgent::CallAgentFunc (String *funcName*, IntPtr *retval*, IntPtr *varg*) [inline]

Calls a function in an agent script. Calls a function in an agent's script file. This function requires manual marshaling by the user.

Parameters:

funcName The name of the function to call
retval A pointer to memory for the return value
varg A pointer to the argument for the function

Returns:

The return value of the underlying MC_CallAgentFunc function call.

Note:

BE VERY CAREFUL! You must marshal your arguments! If possible, use the other CallAgentFunc that handles marshaling automatically.

See also:

Overloaded [CallAgentFunc](#), LibMCConsole example

Definition at line 458 of file MCAgent.cs.

References Agent.

Referenced by CallAgentFunc().

12.77.4.6 int LibMC::MCAgent::DeleteAgent () [inline]

Deletes an agent. Deletes an agent from the agency.

Returns:

The return value of the underlying MC_DeleteAgent function call.

Definition at line 321 of file MCAgent.cs.

References Agent.

12.77.4.7 IntPtr LibMC::MCAgent::GetAgentExecEngine () [inline]

Gets an agent's Ch interpreter. Gets a pointer to the agent's Ch interpreter. Will be improved shortly.

Returns:

A pointer to the Ch interpreter.

Note:

Nothing in the LibMC.NET library can make use of the Ch interpreter yet.

Todo

Wrap MC_GetAgentExecEngine with an object for the void* pointer return type (Ch interpreter).

Definition at line 533 of file MCAgent.cs.

References Agent.

12.77.4.8 void LibMC::MCAgent::GetAgentFields () [inline, private]

Definition at line 163 of file MCAgent.cs.

References Agent, name, numTasks, status, and type.

12.77.4.9 int LibMC::MCAgent::GetAgentReturnData (int task_num, IntPtr data, IntPtr dim, IntPtr extent) [inline]

Calls a function in an agent script. Calls a function in an agent's script file. This function requires manual marshaling by the user.

Parameters:

task_num Task number to get data from

data A pointer to memory for the data

dim A pointer to hold the dimensions of the data

extent A pointer to hold the dimensions of the data

Returns:

The return value of the underlying MC_GetAgentReturnData function call.

Note:

This function does nothing but throw an exception right now.

Todo

Implement GetAgentReturnData

Definition at line 554 of file MCAgent.cs.

12.77.4.10 String LibMC::MCAgent::GetAgentXMLString () [inline]

Gets the agent's XML string. Returns the full XML string associated with the agent.

Returns:

The return value of the underlying MC_GetAgentXMLString function call.

Definition at line 334 of file MCAgent.cs.

References Agent.

12.77.4.11 static implicit LibMC::MCAgent::operator IntPtr (MCAgent *agent*) [inline, static]

Definition at line 299 of file MCAgent.cs.

References Agent.

12.77.4.12 static implicit LibMC::MCAgent::operator MCAgent (IntPtr *ip*) [inline, static]

Definition at line 304 of file MCAgent.cs.

References MCAgent().

12.77.4.13 int LibMC::MCAgent::PrintAgentCode () [inline]

Gets the agent's C code string. Prints the C code associated with the agent to stdout.

Returns:

The return value of the underlying MC_PrintAgentCode function call.

Definition at line 347 of file MCAgent.cs.

References Agent.

12.77.4.14 String LibMC::MCAgent::RetrieveAgentCode () [inline]

Gets the agent's C code string. Returns the C code associated with the agent.

Returns:

A string containing the agent's C code.

Definition at line 359 of file MCAgent.cs.

References Agent.

12.77.4.15 int LibMC::MCAgent::TerminateAgent () [inline]

Terminates an agent. Terminates an agent regardless of the agent's state.

Returns:

The return value of the underlying MC_TerminateAgent function call.

Definition at line 372 of file MCAgent.cs.

References Agent.

12.77.4.16 override string LibMC::MCAgent::ToString () [inline]

Display the agent's fields. Formats and returns a string with all of the agent's properties.

Returns:

A string containing a formatted representation of the agent's properties.

Note:

The agency port and any other [options](#) must be set before calling this function.

Definition at line 126 of file MCAgent.cs.

References AgentID, AgentName, AgentNumTasks, AgentStatus, and AgentType.

12.77.5 Field Documentation

12.77.5.1 IntPtr LibMC::MCAgent::agent_p [private]

Definition at line 63 of file MCAgent.cs.

12.77.5.2 int LibMC::MCAgent::id = -1 [private]

Definition at line 65 of file MCAgent.cs.

12.77.5.3 String LibMC::MCAgent::name = "" [private]

Definition at line 64 of file MCAgent.cs.

Referenced by GetAgentFields().

12.77.5.4 int LibMC::MCAgent::numTasks = -1 [private]

Definition at line 66 of file MCAgent.cs.

Referenced by GetAgentFields().

12.77.5.5 MC_AgentStatus_e LibMC::MCAgent::status = MC_AgentStatus_e.MC_NO_STATUS [private]

Definition at line 67 of file MCAgent.cs.

Referenced by GetAgentFields().

12.77.5.6 MC_AgentType_e LibMC::MCAgent::type = MC_AgentType_e.MC_NONE [private]

Definition at line 68 of file MCAgent.cs.

Referenced by GetAgentFields().

12.77.6 Property Documentation

12.77.6.1 internal IntPtr LibMC::MCAgent::Agent [get, set, private]

Definition at line 143 of file MCAgent.cs.

Referenced by AclPost(), AclRetrieve(), AclWaitRetrieve(), LibMC::MCAgency::AddAgent(), CallAgentFunc(), DeleteAgent(), GetAgentExecEngine(), GetAgentFields(), GetAgentXMLString(), MCAgent(), operator IntPtr(), PrintAgentCode(), LibMC::MCAgency::RegisterService(), RetrieveAgentCode(), and TerminateAgent().

12.77.6.2 int LibMC::MCAgent::AgentID [get]

Gets the agent's ID number. Gets the agent's ID number as assigned by Mobile-C if the agent is a valid agent.

Returns:

The agent's ID number or -1 for an empty agent.

Definition at line 181 of file MCAgent.cs.

Referenced by ToString().

12.77.6.3 String LibMC::MCAgent::AgentName [get]

Gets the agent's name. Gets the agent's name as assigned by Mobile-C or the agent script if the agent is a valid agent.

Returns:

The agent's name or an empty string for an empty agent.

Definition at line 201 of file MCAgent.cs.

Referenced by ToString().

12.77.6.4 int LibMC::MCAgent::AgentNumTasks [get]

Gets the agent's number of tasks. Gets the agent's ID number of tasks if the agent is a valid agent.

Returns:

The agent's ID number of tasks or -1 for an empty agent.

Definition at line 220 of file MCAgent.cs.

Referenced by ToString().

12.77.6.5 MC_AgentStatus_e LibMC::MCAgent::AgentStatus [get, set]

Gets or sets the agent's status. Gets or sets the agent's status. When setting the status, the status is double-checked after setting it and may not be set depending on the state of the agent and the agency.

Returns:

The agent's status or MC_NO_STATUS for an empty agent.

Definition at line 242 of file MCAgent.cs.

Referenced by ToString().

12.77.6.6 MC_AgentType_e LibMC::MCAgent::AgentType [get]

Gets the agent's type. Gets the agent's type.

Returns:

The agent's type or MC_NONE for an empty agent.

Definition at line 264 of file MCAgent.cs.

Referenced by ToString().

12.77.6.7 bool LibMC::MCAgent::Valid [get]

Checks whether the agent is valid. Checks the internal agent pointer to see if it is non-zero.

Returns:

True if the pointer is valid, false otherwise

Definition at line 285 of file MCAgent.cs.

The documentation for this class was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent.cs>

12.78 md2_context Struct Reference

MD2 context structure.

```
#include <md2.h>
```

Data Fields

- unsigned char [cksum](#) [16]
- unsigned char [state](#) [48]
- unsigned char [buffer](#) [16]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]
- [int](#) [left](#)

12.78.1 Detailed Description

MD2 context structure.

Definition at line 10 of file md2.h.

12.78.2 Field Documentation

12.78.2.1 unsigned char md2_context::buffer[16]

data block being processed

Definition at line 14 of file md2.h.

12.78.2.2 unsigned char md2_context::cksum[16]

checksum of the data block

Definition at line 12 of file md2.h.

12.78.2.3 unsigned char md2_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file md2.h.

12.78.2.4 int md2_context::left

amount of data in buffer

Definition at line 18 of file md2.h.

12.78.2.5 unsigned char md2_context::opad[64]

HMAC: outer padding

Definition at line 17 of file md2.h.

12.78.2.6 unsigned char md2_context::state[48]

intermediate digest state

Definition at line 13 of file md2.h.

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md2.h](#)

12.79 md4_context Struct Reference

MD4 context structure.

```
#include <md4.h>
```

Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [4]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]

12.79.1 Detailed Description

MD4 context structure.

Definition at line 10 of file md4.h.

12.79.2 Field Documentation

12.79.2.1 unsigned char md4_context::buffer[64]

data block being processed

Definition at line 14 of file md4.h.

12.79.2.2 unsigned char md4_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file md4.h.

12.79.2.3 unsigned char md4_context::opad[64]

HMAC: outer padding

Definition at line 17 of file md4.h.

12.79.2.4 unsigned long md4_context::state[4]

intermediate digest state

Definition at line 13 of file md4.h.

12.79.2.5 unsigned long md4_context::total[2]

number of bytes processed

Definition at line 12 of file md4.h.

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md4.h>

12.80 md5_context Struct Reference

MD5 context structure.

```
#include <md5.h>
```

Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [4]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]

12.80.1 Detailed Description

MD5 context structure.

Definition at line 10 of file md5.h.

12.80.2 Field Documentation

12.80.2.1 unsigned char md5_context::buffer[64]

data block being processed

Definition at line 14 of file md5.h.

Referenced by md5_update().

12.80.2.2 unsigned char md5_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file md5.h.

Referenced by md5_hmac_starts().

12.80.2.3 unsigned char md5_context::opad[64]

HMAC: outer padding

Definition at line 17 of file md5.h.

Referenced by md5_hmac_finish(), and md5_hmac_starts().

12.80.2.4 unsigned long md5_context::state[4]

intermediate digest state

Definition at line 13 of file md5.h.

Referenced by md5_finish(), md5_process(), md5_starts(), and ssl_calc_finished().

12.80.2.5 unsigned long md5_context::total[2]

number of bytes processed

Definition at line 12 of file md5.h.

Referenced by md5_finish(), md5_starts(), and md5_update().

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md5.h>

12.81 message_s Struct Reference

```
#include <message.h>
```

Data Fields

- struct sockaddr_in * [addr](#)
- [int connect_id](#)
- [int message_id](#)
- [int isHTTP](#)
- enum [message_type_e](#) [message_type](#)
- enum [http_performative_e](#) [http_type](#)
- [mxml_node_t](#) * [xml_root](#)
- [mxml_node_t](#) * [xml_payload](#)
- char * [message_body](#)
- char * [update_name](#)
- [int update_num](#)
- char * [from_address](#)
- char * [to_address](#)
- char * [target](#)
- [int agent_xml_flag](#)
- char * [sending_agent_name](#)

12.81.1 Detailed Description

Definition at line 77 of file message.h.

12.81.2 Field Documentation

12.81.2.1 struct sockaddr_in* message_s::addr [read]

Definition at line 79 of file message.h.

Referenced by [message_Destroy\(\)](#), [message_InitializeFromAgent\(\)](#), [message_InitializeFromConnection\(\)](#), [message_InitializeFromString\(\)](#), [message_New\(\)](#), and [message_xml_parse__message\(\)](#).

12.81.2.2 int message_s::agent_xml_flag

Definition at line 111 of file message.h.

Referenced by [agent_Initialize\(\)](#), [message_Destroy\(\)](#), [message_InitializeFromAgent\(\)](#), and [message_New\(\)](#).

12.81.2.3 int message_s::connect_id

Definition at line 82 of file message.h.

Referenced by [message_InitializeFromConnection\(\)](#), [message_InitializeFromString\(\)](#), and [message_New\(\)](#).

12.81.2.4 `char* message_s::from_address`

Definition at line 101 of file message.h.

Referenced by `AP_QUEUE_SEARCH_TEMPLATE()`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, `message_New()`, and `message_xml_parse__message()`.

12.81.2.5 `enum http_performative_e message_s::http_type`

Definition at line 90 of file message.h.

Referenced by `message_New()`.

12.81.2.6 `int message_s::isHTTP`

Definition at line 86 of file message.h.

Referenced by `message_New()`, `mtp_http_ComposeMessage()`, and `mtp_http_CreateMessage()`.

12.81.2.7 `char* message_s::message_body`

Definition at line 96 of file message.h.

Referenced by `acc_connection_Thread()`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, `message_New()`, `message_send_Thread()`, `mtp_http_ComposeMessage()`, and `mtp_http_CreateMessage()`.

12.81.2.8 `int message_s::message_id`

Definition at line 83 of file message.h.

Referenced by `AP_QUEUE_SEARCH_TEMPLATE()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, `message_InitializeFromString()`, and `message_New()`.

12.81.2.9 `enum message_type_e message_s::message_type`

Definition at line 89 of file message.h.

Referenced by `acc_connection_Thread()`, `acc_MessageHandlerThread()`, `agent_Initialize()`, `MC_AclSend()`, `message_InitializeFromAgent()`, `message_InitializeFromString()`, `message_New()`, and `message_xml_parse__message()`.

12.81.2.10 `char* message_s::sending_agent_name`

Definition at line 112 of file message.h.

Referenced by `message_InitializeFromAgent()`.

12.81.2.11 `char* message_s::target`

Definition at line 104 of file message.h.

Referenced by MC_AclSend(), message_Destroy(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), message_New(), and mtp_http_ComposeMessage().

12.81.2.12 char* message_s::to_address

Definition at line 102 of file message.h.

Referenced by acc_MessageHandlerThread(), AP_QUEUE_SEARCH_TEMPLATE(), MC_LoadAgentFromFile(), message_Destroy(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), message_New(), message_queue_SendOutgoing(), message_send_Thread(), mtp_http_ComposeMessage(), and mtp_http_CreateMessage().

12.81.2.13 char* message_s::update_name

Definition at line 98 of file message.h.

Referenced by message_Destroy(), message_InitializeFromAgent(), message_InitializeFromString(), and message_New().

12.81.2.14 int message_s::update_num

Definition at line 99 of file message.h.

Referenced by message_New().

12.81.2.15 mxml_node_t* message_s::xml_payload

Definition at line 93 of file message.h.

Referenced by agent_Initialize(), MC_LoadAgentFromFile(), message_New(), and message_xml_parse_message().

12.81.2.16 mxml_node_t* message_s::xml_root

Definition at line 92 of file message.h.

Referenced by acc_connection_Thread(), agent_Initialize(), MC_LoadAgentFromFile(), message_Destroy(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), message_New(), and message_xml_parse().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[message.h](#)

12.82 message_send_arg_s Struct Reference

```
#include <message.h>
```

Data Fields

- struct [mc_platform_s](#) * [mc_platform](#)
- [message_p](#) [message](#)
- char * [privatekey](#)

12.82.1 Detailed Description

Definition at line 116 of file [message.h](#).

12.82.2 Field Documentation

12.82.2.1 struct [mc_platform_s](#)* [message_send_arg_s::mc_platform](#) [read]

Definition at line 118 of file [message.h](#).

Referenced by [message_Send\(\)](#).

12.82.2.2 [message_p](#) [message_send_arg_s::message](#)

Definition at line 119 of file [message.h](#).

Referenced by [message_Send\(\)](#).

12.82.2.3 char* [message_send_arg_s::privatekey](#)

Definition at line 120 of file [message.h](#).

Referenced by [message_Send\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/include/message.h](#)

12.83 mpi Struct Reference

MPI structure.

```
#include <bignum.h>
```

Data Fields

- `int s`
- `int n`
- `t_int * p`

12.83.1 Detailed Description

MPI structure.

Definition at line 48 of file `bignum.h`.

12.83.2 Field Documentation

12.83.2.1 `int mpi::n`

total # of limbs

Definition at line 51 of file `bignum.h`.

Referenced by `debug_print_mpi()`, `mpi_add_abs()`, `mpi_add_int()`, `mpi_cmp_abs()`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_div_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_is_prime()`, `mpi_lsb()`, `mpi_lset()`, `mpi_mod_int()`, `mpi_montmul()`, `mpi_montred()`, `mpi_msb()`, `mpi_mul_int()`, `mpi_mul_mpi()`, `mpi_shift_l()`, `mpi_shift_r()`, `mpi_sub_abs()`, `mpi_sub_int()`, `mpi_write_string()`, and `x509parse_cert_info()`.

12.83.2.2 `t_int* mpi::p`

pointer to limbs

Definition at line 52 of file `bignum.h`.

Referenced by `debug_print_mpi()`, `dhm_make_params()`, `dhm_make_public()`, `mpi_add_abs()`, `mpi_add_int()`, `mpi_cmp_abs()`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_div_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_grow()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_lsb()`, `mpi_lset()`, `mpi_mod_int()`, `mpi_montg_init()`, `mpi_montmul()`, `mpi_montred()`, `mpi_msb()`, `mpi_mul_int()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_read_string()`, `mpi_shift_l()`, `mpi_shift_r()`, `mpi_sub_abs()`, `mpi_sub_int()`, `mpi_write_binary()`, `mpi_write_string()`, and `rsa_check_pubkey()`.

12.83.2.3 `int mpi::s`

integer sign

Definition at line 50 of file `bignum.h`.

Referenced by `mpi_add_int()`, `mpi_add_mpi()`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_div_int()`, `mpi_div_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_is_prime()`, `mpi_lset()`, `mpi_montred()`,

`mpi_mul_int()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_sub_int()`, `mpi_sub_mpi()`, and `mpi_write_string()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/bignum.h`

12.84 mtp_http_content_s Struct Reference

```
#include <mtp_http.h>
```

Data Fields

- char * [content_type](#)
- void * [data](#)

12.84.1 Detailed Description

Definition at line 106 of file mtp_http.h.

12.84.2 Field Documentation

12.84.2.1 char* mtp_http_content_s::content_type

Definition at line 108 of file mtp_http.h.

Referenced by MC_AclSend(), mtp_http_CreateMessage(), mtp_http_Destroy(), and mtp_http_Parse().

12.84.2.2 void* mtp_http_content_s::data

Definition at line 109 of file mtp_http.h.

Referenced by acc_connection_Thread(), MC_AclSend(), mtp_http_CreateMessage(), mtp_http_Destroy(), and mtp_http_Parse().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[mtp_http.h](#)

12.85 mtp_http_s Struct Reference

```
#include <mtp_http.h>
```

Data Fields

- enum [http_status_code_e](#) [http_status_code](#)
- enum [http_performative_e](#) [http_performative](#)
- char * [http_version](#)
- char * [host](#)
- char * [return_code](#)
- char * [target](#)
- char * [date](#)
- char * [server](#)
- char * [accept_ranges](#)
- char * [content_length](#)
- char * [connection](#)
- char * [content_type](#)
- char * [user_agent](#)
- char * [cache_control](#)
- char * [mime_version](#)
- int [response_code](#)
- char * [response_string](#)
- int [message_parts](#)
- char * [boundary](#)
- struct [mtp_http_content_s](#) * [content](#)
- int [header_length](#)

12.85.1 Detailed Description

Definition at line 112 of file `mtp_http.h`.

12.85.2 Field Documentation

12.85.2.1 char* mtp_http_s::accept_ranges

Definition at line 123 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`.

12.85.2.2 char* mtp_http_s::boundary

Definition at line 139 of file `mtp_http.h`.

Referenced by `mtp_http_Destroy()`, and `mtp_http_Parse()`.

12.85.2.3 char* mtp_http_s::cache_control

Definition at line 129 of file `mtp_http.h`.

12.85.2.4 char* mtp_http_s::connection

Definition at line 125 of file mtp_http.h.

Referenced by mtp_http_Destroy().

12.85.2.5 struct mtp_http_content_s* mtp_http_s::content [read]

Definition at line 140 of file mtp_http.h.

Referenced by acc_connection_Thread(), MC_AclSend(), mtp_http_CreateMessage(), mtp_http_Destroy(), mtp_http_New(), and mtp_http_Parse().

12.85.2.6 char* mtp_http_s::content_length

Definition at line 124 of file mtp_http.h.

Referenced by mtp_http_Destroy(), mtp_http_InitializeFromConnection(), and mtp_http_Parse().

12.85.2.7 char* mtp_http_s::content_type

Definition at line 126 of file mtp_http.h.

Referenced by mtp_http_Destroy(), and mtp_http_Parse().

12.85.2.8 char* mtp_http_s::date

Definition at line 121 of file mtp_http.h.

Referenced by mtp_http_Destroy().

12.85.2.9 int mtp_http_s::header_length

Definition at line 143 of file mtp_http.h.

Referenced by mtp_http_InitializeFromConnection(), and mtp_http_ParseHeader().

12.85.2.10 char* mtp_http_s::host

Definition at line 117 of file mtp_http.h.

Referenced by MC_AclSend(), mtp_http_CreateMessage(), and mtp_http_Destroy().

12.85.2.11 enum http_performative_e mtp_http_s::http_performative

Definition at line 115 of file mtp_http.h.

Referenced by acc_connection_Thread(), http_ParseRequest(), mtp_http_InitializeFromConnection(), and mtp_http_Parse().

12.85.2.12 enum http_status_code_e mtp_http_s::http_status_code

Definition at line 114 of file mtp_http.h.

12.85.2.13 char* mtp_http_s::http_version

Definition at line 116 of file mtp_http.h.

Referenced by mtp_http_Destroy().

12.85.2.14 int mtp_http_s::message_parts

Definition at line 138 of file mtp_http.h.

Referenced by acc_connection_Thread(), MC_AclSend(), mtp_http_CreateMessage(), mtp_http_Destroy(), and mtp_http_Parse().

12.85.2.15 char* mtp_http_s::mime_version

Definition at line 130 of file mtp_http.h.

12.85.2.16 int mtp_http_s::response_code

Definition at line 133 of file mtp_http.h.

Referenced by http_ParseRequest().

12.85.2.17 char* mtp_http_s::response_string

Definition at line 134 of file mtp_http.h.

Referenced by http_ParseRequest(), and mtp_http_Destroy().

12.85.2.18 char* mtp_http_s::return_code

Definition at line 118 of file mtp_http.h.

Referenced by mtp_http_Destroy().

12.85.2.19 char* mtp_http_s::server

Definition at line 122 of file mtp_http.h.

Referenced by mtp_http_Destroy().

12.85.2.20 char* mtp_http_s::target

Definition at line 119 of file mtp_http.h.

Referenced by acc_connection_Thread(), http_ParseRequest(), MC_AclSend(), mtp_http_CreateMessage(), and mtp_http_Destroy().

12.85.2.21 char* mtp_http_s::user_agent

Definition at line 127 of file mtp_http.h.

Referenced by mtp_http_Destroy().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/include/[mtp_http.h](#)

12.86 mxml_attr_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- char * [name](#)
- char * [value](#)

12.86.1 Detailed Description

Definition at line 86 of file mxml.h.

12.86.2 Field Documentation

12.86.2.1 char* mxml_attr_s::name

Definition at line 88 of file mxml.h.

Referenced by [mxml_write_node\(\)](#), [mxmlDelete\(\)](#), [mxmlElementGetAttr\(\)](#), and [mxmlElementSetAttr\(\)](#).

12.86.2.2 char* mxml_attr_s::value

Definition at line 89 of file mxml.h.

Referenced by [mxml_write_node\(\)](#), [mxmlDelete\(\)](#), [mxmlElementGetAttr\(\)](#), and [mxmlElementSetAttr\(\)](#).

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml.h](#)

12.87 mxml_custom_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- void * [data](#)
- void(* [destroy](#))(void *)

12.87.1 Detailed Description

Definition at line 105 of file mxml.h.

12.87.2 Field Documentation

12.87.2.1 void* mxml_custom_s::data

Definition at line 107 of file mxml.h.

Referenced by mxmlDelete(), mxmlNewCustom(), and mxmlSetCustom().

12.87.2.2 void(* mxml_custom_s::destroy)(void *)

Referenced by mxmlDelete(), mxmlNewCustom(), and mxmlSetCustom().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/[mxml.h](#)

12.88 mxml_fdbuf_s Struct Reference

Data Fields

- `int fd`
- `unsigned char * current`
- `unsigned char * end`
- `unsigned char buffer [8192]`

12.88.1 Detailed Description

Definition at line 83 of file mxml-file.c.

12.88.2 Field Documentation

12.88.2.1 `unsigned char mxml_fdbuf_s::buffer[8192]`

Definition at line 86 of file mxml-file.c.

Referenced by `mxml_fd_read()`, `mxml_fd_write()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

12.88.2.2 `unsigned char* mxml_fdbuf_s::current`

Definition at line 86 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_fd_putc()`, `mxml_fd_read()`, `mxml_fd_write()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

12.88.2.3 `unsigned char * mxml_fdbuf_s::end`

Definition at line 86 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_fd_putc()`, `mxml_fd_read()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

12.88.2.4 `int mxml_fdbuf_s::fd`

Definition at line 85 of file mxml-file.c.

Referenced by `mxml_fd_read()`, `mxml_fd_write()`, `mxmlLoadFd()`, and `mxmlSaveFd()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c`

12.89 mxml_index_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- char * [attr](#)
- int [num_nodes](#)
- int [alloc_nodes](#)
- int [cur_node](#)
- [mxml_node_t](#) ** [nodes](#)

12.89.1 Detailed Description

Definition at line 133 of file mxml.h.

12.89.2 Field Documentation

12.89.2.1 int mxml_index_s::alloc_nodes

Definition at line 137 of file mxml.h.

Referenced by [mxmlIndexDelete\(\)](#), and [mxmlIndexNew\(\)](#).

12.89.2.2 char* mxml_index_s::attr

Definition at line 135 of file mxml.h.

Referenced by [index_compare\(\)](#), [index_find\(\)](#), [mxmlIndexDelete\(\)](#), [mxmlIndexFind\(\)](#), and [mxmlIndexNew\(\)](#).

12.89.2.3 int mxml_index_s::cur_node

Definition at line 138 of file mxml.h.

Referenced by [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), and [mxmlIndexReset\(\)](#).

12.89.2.4 mxml_node_t** mxml_index_s::nodes

Definition at line 139 of file mxml.h.

Referenced by [index_sort\(\)](#), [mxmlIndexDelete\(\)](#), [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), [mxmlIndexNew\(\)](#), and [mxmlIndexReset\(\)](#).

12.89.2.5 int mxml_index_s::num_nodes

Definition at line 136 of file mxml.h.

Referenced by [main\(\)](#), [mxmlIndexEnum\(\)](#), [mxmlIndexFind\(\)](#), [mxmlIndexNew\(\)](#), and [mxmlIndexReset\(\)](#).

The documentation for this struct was generated from the following file:

- </home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml.h>

12.90 mxml_node_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- [mxml_type_t](#) type
- struct [mxml_node_s](#) * next
- struct [mxml_node_s](#) * prev
- struct [mxml_node_s](#) * parent
- struct [mxml_node_s](#) * child
- struct [mxml_node_s](#) * last_child
- [mxml_value_t](#) value

12.90.1 Detailed Description

Definition at line 122 of file mxml.h.

12.90.2 Field Documentation

12.90.2.1 struct mxml_node_s* mxml_node_s::child [read]

Definition at line 128 of file mxml.h.

Referenced by `add_variable()`, `agent_xml_parse__fill_row_data()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `main()`, `mxml_write_node()`, `mxmlAdd()`, `mxmlDelete()`, `mxmlRemove()`, `mxmlWalkNext()`, `scan_file()`, `sort_node()`, `whitespace_cb()`, `write_documentation()`, `write_element()`, and `xml_get_text()`.

12.90.2.2 struct mxml_node_s* mxml_node_s::last_child [read]

Definition at line 129 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxmlAdd()`, `mxmlRemove()`, `mxmlWalkPrev()`, and `scan_file()`.

12.90.2.3 struct mxml_node_s* mxml_node_s::next [read]

Definition at line 125 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxml_write_node()`, `mxmlAdd()`, `mxmlFindElement()`, `mxmlRemove()`, `mxmlWalkNext()`, `scan_file()`, `sort_node()`, `xml_get_next_element()`, and `xml_get_text()`.

12.90.2.4 struct mxml_node_s* mxml_node_s::parent [read]

Definition at line 127 of file mxml.h.

Referenced by `agent_xml_parse__data()`, `mxml_load_data()`, `mxmlAdd()`, `mxmlRemove()`, `mxmlWalkNext()`, `mxmlWalkPrev()`, `sort_node()`, `whitespace_cb()`, `ws_cb()`, and `xml_find_sibling()`.

12.90.2.5 struct mxml_node_s* mxml_node_s::prev [read]

Definition at line 126 of file mxml.h.

Referenced by mxml_write_node(), mxmlAdd(), mxmlRemove(), and mxmlWalkPrev().

12.90.2.6 mxml_type_t mxml_node_s::type

Definition at line 124 of file mxml.h.

Referenced by agent_xml_parse__fill_row_data(), fipa_envelope_HandleAclRepresentation(), fipa_envelope_HandleComments(), fipa_envelope_HandleDate(), fipa_envelope_HandlePayloadEncoding(), fipa_envelope_HandlePayloadLength(), fipa_envelope_ParseAddresses(), fipa_envelope_ParseAgentIdentifier(), main(), mxml_new(), mxml_write_node(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxmlSetCustom(), mxmlSetElement(), mxmlSetInteger(), mxmlSetOpaque(), mxmlSetReal(), mxmlSetText(), mxmlSetTextf(), write_element(), xml_get_cdata(), xml_get_element_name(), xml_get_next_element(), and xml_get_text().

12.90.2.7 mxml_value_t mxml_node_s::value

Definition at line 130 of file mxml.h.

Referenced by add_variable(), agent_xml_parse__fill_row_data(), fipa_envelope_HandleAclRepresentation(), fipa_envelope_HandleComments(), fipa_envelope_HandleDate(), fipa_envelope_HandlePayloadEncoding(), fipa_envelope_HandlePayloadLength(), fipa_envelope_ParseAddresses(), fipa_envelope_ParseAgentIdentifier(), index_compare(), index_find(), main(), mxml_get_entity(), mxml_load_data(), mxml_parse_element(), mxml_write_node(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxmlIndexNew(), mxmlNewCustom(), mxmlNewElement(), mxmlNewInteger(), mxmlNewOpaque(), mxmlNewReal(), mxmlNewText(), mxmlNewTextf(), mxmlSetCustom(), mxmlSetElement(), mxmlSetInteger(), mxmlSetOpaque(), mxmlSetReal(), mxmlSetText(), mxmlSetTextf(), scan_file(), sort_node(), type_cb(), update_comment(), whitespace_cb(), write_element(), ws_cb(), xml_get_element_name(), and xml_get_text().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/[mxml.h](#)

12.91 mxml_text_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- [int whitespace](#)
- [char * string](#)

12.91.1 Detailed Description

Definition at line 99 of file mxml.h.

12.91.2 Field Documentation

12.91.2.1 char* mxml_text_s::string

Definition at line 102 of file mxml.h.

Referenced by `add_variable()`, `agent_xml_parse__fill_row_data()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `main()`, `mxml_write_node()`, `mxmlDelete()`, `mxmlNewText()`, `mxmlNewTextf()`, `mxmlSetText()`, `mxmlSetTextf()`, `scan_file()`, `update_comment()`, `write_element()`, and `xml_get_text()`.

12.91.2.2 int mxml_text_s::whitespace

Definition at line 101 of file mxml.h.

Referenced by `add_variable()`, `main()`, `mxml_write_node()`, `mxmlNewText()`, `mxmlNewTextf()`, `mxmlSetText()`, `mxmlSetTextf()`, `scan_file()`, and `write_element()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml.h`

12.92 mxml_value_s Struct Reference

```
#include <mxml.h>
```

Data Fields

- `char * name`
- `int num_attrs`
- `mxml_attr_t * attrs`

12.92.1 Detailed Description

Definition at line 92 of file `mxml.h`.

12.92.2 Field Documentation

12.92.2.1 `mxml_attr_t* mxml_value_s::attrs`

Definition at line 96 of file `mxml.h`.

Referenced by `mxml_write_node()`, `mxmlDelete()`, `mxmlElementGetAttr()`, and `mxmlElementSetAttr()`.

12.92.2.2 `char* mxml_value_s::name`

Definition at line 94 of file `mxml.h`.

Referenced by `index_compare()`, `index_find()`, `main()`, `mxml_get_entity()`, `mxml_load_data()`, `mxml_parse_element()`, `mxml_write_node()`, `mxmlDelete()`, `mxmlElementSetAttr()`, `mxmlFindElement()`, `mxmlIndexNew()`, `mxmlNewElement()`, `mxmlSetElement()`, `scan_file()`, `sort_node()`, `type_cb()`, `update_comment()`, `whitespace_cb()`, `ws_cb()`, `xml_get_element_name()`, and `xml_get_text()`.

12.92.2.3 `int mxml_value_s::num_attrs`

Definition at line 95 of file `mxml.h`.

Referenced by `mxml_write_node()`, `mxmlDelete()`, `mxmlElementGetAttr()`, and `mxmlElementSetAttr()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml.h`

12.93 mxml_value_u Union Reference

```
#include <mxml.h>
```

Data Fields

- [mxml_element_t](#) element
- [int](#) integer
- [char *](#) opaque
- [double](#) real
- [mxml_text_t](#) text
- [mxml_custom_t](#) custom

12.93.1 Detailed Description

Definition at line 112 of file mxml.h.

12.93.2 Field Documentation

12.93.2.1 mxml_custom_t mxml_value_u::custom

Definition at line 119 of file mxml.h.

Referenced by [mxmlDelete\(\)](#), [mxmlNewCustom\(\)](#), and [mxmlSetCustom\(\)](#).

12.93.2.2 mxml_element_t mxml_value_u::element

Definition at line 114 of file mxml.h.

Referenced by [index_compare\(\)](#), [index_find\(\)](#), [main\(\)](#), [mxml_get_entity\(\)](#), [mxml_load_data\(\)](#), [mxml_parse_element\(\)](#), [mxml_write_node\(\)](#), [mxmlDelete\(\)](#), [mxmlElementGetAttr\(\)](#), [mxmlElementSetAttr\(\)](#), [mxmlFindElement\(\)](#), [mxmlIndexNew\(\)](#), [mxmlNewElement\(\)](#), [mxmlSetElement\(\)](#), [scan_file\(\)](#), [sort_node\(\)](#), [type_cb\(\)](#), [update_comment\(\)](#), [whitespace_cb\(\)](#), [ws_cb\(\)](#), [xml_get_element_name\(\)](#), and [xml_get_text\(\)](#).

12.93.2.3 int mxml_value_u::integer

Definition at line 115 of file mxml.h.

Referenced by [main\(\)](#), [mxml_write_node\(\)](#), [mxmlNewInteger\(\)](#), and [mxmlSetInteger\(\)](#).

12.93.2.4 char* mxml_value_u::opaque

Definition at line 116 of file mxml.h.

Referenced by [main\(\)](#), [mxml_write_node\(\)](#), [mxmlDelete\(\)](#), [mxmlNewOpaque\(\)](#), and [mxmlSetOpaque\(\)](#).

12.93.2.5 double mxml_value_u::real

Definition at line 117 of file mxml.h.

Referenced by [main\(\)](#), [mxml_write_node\(\)](#), [mxmlNewReal\(\)](#), and [mxmlSetReal\(\)](#).

12.93.2.6 mxml_text_t mxml_value_u::text

Definition at line 118 of file mxml.h.

Referenced by add_variable(), agent_xml_parse__fill_row_data(), fipa_envelope_HandleAclRepresentation(), fipa_envelope_HandleComments(), fipa_envelope_HandleDate(), fipa_envelope_HandlePayloadEncoding(), fipa_envelope_HandlePayloadLength(), fipa_envelope_ParseAddresses(), fipa_envelope_ParseAgentIdentifier(), main(), mxml_write_node(), mxmlDelete(), mxmlNewText(), mxmlNewTextf(), mxmlSetText(), mxmlSetTextf(), scan_file(), update_comment(), write_element(), and xml_get_text().

The documentation for this union was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/[mxml.h](#)

12.94 options Struct Reference

Data Fields

- `int opmode`
- `int iomode`
- `char * server_name`
- `int server_port`
- `int command`
- `int buffer_size`
- `int max_bytes`
- `int debug_level`
- `int conn_timeout`
- `int max_connections`
- `int session_reuse`
- `int session_lifetime`
- `int force_cipher` [2]

12.94.1 Detailed Description

Definition at line 75 of file `ssl_test.c`.

12.94.2 Field Documentation

12.94.2.1 `int options::buffer_size`

Definition at line 82 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

12.94.2.2 `int options::command`

Definition at line 81 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

12.94.2.3 `int options::conn_timeout`

Definition at line 85 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

12.94.2.4 `int options::debug_level`

Definition at line 84 of file `ssl_test.c`.

Referenced by `main()`.

12.94.2.5 int options::force_cipher[2]

Definition at line 89 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.6 int options::iomode

Definition at line 78 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.7 int options::max_bytes

Definition at line 83 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.8 int options::max_connections

Definition at line 86 of file ssl_test.c.

Referenced by main().

12.94.2.9 int options::opmode

Definition at line 77 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.10 char* options::server_name

Definition at line 79 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.11 int options::server_port

Definition at line 80 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.12 int options::session_lifetime

Definition at line 88 of file ssl_test.c.

Referenced by main(), and ssl_test().

12.94.2.13 int options::session_reuse

Definition at line 87 of file ssl_test.c.

Referenced by main(), and ssl_test().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/ssl_test.c](#)

12.95 Program1::Program Class Reference

Static Private Member Functions

- static void [Main](#) (string[] args)

12.95.1 Detailed Description

Definition at line 8 of file Program.cs.

12.95.2 Member Function Documentation

12.95.2.1 static void Program1::Program::Main (string[] args) [inline, static, private]

Definition at line 10 of file Program.cs.

References [EmbeddedCh::ChInterp::AppendRunScript\(\)](#), [EmbeddedCh::ChInterp::AppendRunScriptFile\(\)](#), [EmbeddedCh::ChInterp::End\(\)](#), [EmbeddedCh::ChInterp::Initialize\(\)](#), and [EmbeddedCh::ChInterp::RunScript\(\)](#).

The documentation for this class was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/Program1/Program.cs](#)

12.96 rsa_context Struct Reference

RSA context structure.

```
#include <rsa.h>
```

Data Fields

- [int ver](#)
- [int len](#)
- [mpi N](#)
- [mpi E](#)
- [mpi D](#)
- [mpi P](#)
- [mpi Q](#)
- [mpi DP](#)
- [mpi DQ](#)
- [mpi QP](#)
- [mpi RN](#)
- [mpi RP](#)
- [mpi RQ](#)
- [int padding](#)
- [int hash_id](#)
- [int\(* f_rng\)\(void *\)](#)
- [void * p_rng](#)

12.96.1 Detailed Description

RSA context structure.

Definition at line 56 of file rsa.h.

12.96.2 Field Documentation

12.96.2.1 mpi rsa_context::D

private exponent

Definition at line 64 of file rsa.h.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_check_privkey()`, `rsa_decryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_self_test()`, and `x509parse_key()`.

12.96.2.2 mpi rsa_context::DP

$D \% (P - 1)$

Definition at line 67 of file rsa.h.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_self_test()`, and `x509parse_key()`.

12.96.2.3 `mpi rsa_context::DQ`

$D \% (Q - 1)$

Definition at line 68 of file `rsa.h`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_self_test()`, and `x509parse_key()`.

12.96.2.4 `mpi rsa_context::E`

public exponent

Definition at line 62 of file `rsa.h`.

Referenced by `d2i_RSA_PUBKEY()`, `debug_print_crt()`, `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `rsa_decryption()`, `rsa_encryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_public()`, `rsa_self_test()`, `x509parse_crt()`, and `x509parse_key()`.

12.96.2.5 `int(* rsa_context::f_rng)(void *)`

RNG function

Referenced by `rsa_gen_key()`, and `rsa_init()`.

12.96.2.6 `int rsa_context::hash_id`

hash identifier

Definition at line 76 of file `rsa.h`.

Referenced by `rsa_init()`.

12.96.2.7 `int rsa_context::len`

size(N) in chars

Definition at line 59 of file `rsa.h`.

Referenced by `d2i_RSA_PUBKEY()`, `main()`, `rsa_decryption()`, `rsa_encryption()`, `rsa_gen_key()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, `rsa_pkcs1_verify()`, `rsa_private()`, `rsa_public()`, `rsa_self_test()`, `ssl_parse_certificate_verify()`, `ssl_parse_client_key_exchange()`, `ssl_parse_server_key_exchange()`, `ssl_write_certificate_verify()`, `ssl_write_client_key_exchange()`, `ssl_write_server_key_exchange()`, `x509parse_crt()`, and `x509parse_key()`.

12.96.2.8 `mpi rsa_context::N`

public modulus

Definition at line 61 of file `rsa.h`.

Referenced by `d2i_RSA_PUBKEY()`, `debug_print_crt()`, `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `rsa_decryption()`, `rsa_encryption()`, `rsa_free()`, `rsa_gen_key()`, `rsa_private()`, `rsa_public()`, `rsa_self_test()`, `x509parse_cert_info()`, `x509parse_crt()`, and `x509parse_key()`.

12.96.2.9 mpi rsa_context::P

1st prime factor

Definition at line 65 of file rsa.h.

Referenced by generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), main(), rsa_check_privkey(), rsa_decryption(), rsa_free(), rsa_gen_key(), rsa_private(), rsa_self_test(), and x509parse_key().

12.96.2.10 void* rsa_context::p_rng

RNG parameter

Definition at line 78 of file rsa.h.

Referenced by rsa_gen_key(), and rsa_init().

12.96.2.11 int rsa_context::padding

1.5 or OAEP/PSS

Definition at line 75 of file rsa.h.

Referenced by rsa_init(), rsa_pkcs1_decrypt(), rsa_pkcs1_encrypt(), rsa_pkcs1_sign(), and rsa_pkcs1_verify().

12.96.2.12 mpi rsa_context::Q

2nd prime factor

Definition at line 66 of file rsa.h.

Referenced by generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), main(), rsa_check_privkey(), rsa_decryption(), rsa_free(), rsa_gen_key(), rsa_private(), rsa_self_test(), and x509parse_key().

12.96.2.13 mpi rsa_context::QP

$1 / (Q \% P)$

Definition at line 69 of file rsa.h.

Referenced by generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), main(), rsa_decryption(), rsa_free(), rsa_gen_key(), rsa_private(), rsa_self_test(), and x509parse_key().

12.96.2.14 mpi rsa_context::RN

cached $R^2 \bmod N$

Definition at line 71 of file rsa.h.

Referenced by rsa_free(), rsa_private(), and rsa_public().

12.96.2.15 mpi rsa_context::RP

cached $R^2 \bmod P$

Definition at line 72 of file rsa.h.

Referenced by `rsa_free()`, and `rsa_private()`.

12.96.2.16 `mpi rsa_context::RQ`

cached $R^2 \bmod Q$

Definition at line 73 of file rsa.h.

Referenced by `rsa_free()`, and `rsa_private()`.

12.96.2.17 `int rsa_context::ver`

always 0

Definition at line 58 of file rsa.h.

Referenced by `x509parse_key()`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/rsa.h`

12.97 LibMC::Properties::Settings Class Reference

Public Member Functions

- [Settings](#) ()

Properties

- static [Settings Default](#) [get]

Private Member Functions

- void [SettingChangingEventHandler](#) (object sender, System.Configuration.SettingChangingEventArgs e)
- void [SettingsSavingEventHandler](#) (object sender, System.ComponentModel.CancelEventArgs e)

Static Private Attributes

- static [Settings defaultInstance](#) = (([Settings](#))(global::System.Configuration.ApplicationSettingsBase.Synchronized(new [Settings](#)())))

12.97.1 Detailed Description

Definition at line 16 of file Settings.Designer.cs.

12.97.2 Constructor & Destructor Documentation

12.97.2.1 LibMC::Properties::Settings::Settings () [inline]

Definition at line 19 of file Settings.cs.

12.97.3 Member Function Documentation

12.97.3.1 void LibMC::Properties::Settings::SettingChangingEventHandler (object sender, System.Configuration.SettingChangingEventArgs e) [inline, private]

Definition at line 28 of file Settings.cs.

12.97.3.2 void LibMC::Properties::Settings::SettingsSavingEventHandler (object sender, System.ComponentModel.CancelEventArgs e) [inline, private]

Definition at line 32 of file Settings.cs.

12.97.4 Field Documentation

12.97.4.1 Settings LibMC::Properties::Settings::defaultInstance =
((Settings)(global::System.Configuration.ApplicationSettingsBase.Synchronized(new
Settings())) [static, private]

Definition at line 18 of file Settings.Designer.cs.

12.97.5 Property Documentation

12.97.5.1 Settings LibMC::Properties::Settings::Default [static, get]

Definition at line 20 of file Settings.Designer.cs.

The documentation for this class was generated from the following files:

- [/home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties/Settings.Designer.cs](#)
- [/home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Settings.cs](#)

12.98 sha1_context Struct Reference

SHA-1 context structure.

```
#include <sha1.h>
```

Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [5]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]

12.98.1 Detailed Description

SHA-1 context structure.

Definition at line 10 of file `sha1.h`.

12.98.2 Field Documentation

12.98.2.1 unsigned char sha1_context::buffer[64]

data block being processed

Definition at line 14 of file `sha1.h`.

Referenced by `sha1_update()`.

12.98.2.2 unsigned char sha1_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file `sha1.h`.

Referenced by `sha1_hmac_starts()`.

12.98.2.3 unsigned char sha1_context::opad[64]

HMAC: outer padding

Definition at line 17 of file `sha1.h`.

Referenced by `sha1_hmac_finish()`, and `sha1_hmac_starts()`.

12.98.2.4 unsigned long sha1_context::state[5]

intermediate digest state

Definition at line 13 of file `sha1.h`.

Referenced by `sha1_finish()`, `sha1_process()`, `sha1_starts()`, and `ssl_calc_finished()`.

12.98.2.5 unsigned long sha1_context::total[2]

number of bytes processed

Definition at line 12 of file sha1.h.

Referenced by sha1_finish(), sha1_starts(), and sha1_update().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha1.h](#)

12.99 sha2_context Struct Reference

SHA-256 context structure.

```
#include <sha2.h>
```

Data Fields

- unsigned long [total](#) [2]
- unsigned long [state](#) [8]
- unsigned char [buffer](#) [64]
- unsigned char [ipad](#) [64]
- unsigned char [opad](#) [64]
- [int is224](#)

12.99.1 Detailed Description

SHA-256 context structure.

Definition at line 10 of file sha2.h.

12.99.2 Field Documentation

12.99.2.1 unsigned char sha2_context::buffer[64]

data block being processed

Definition at line 14 of file sha2.h.

Referenced by sha2_update().

12.99.2.2 unsigned char sha2_context::ipad[64]

HMAC: inner padding

Definition at line 16 of file sha2.h.

Referenced by sha2_hmac_starts().

12.99.2.3 int sha2_context::is224

0 => SHA-256, else SHA-224

Definition at line 18 of file sha2.h.

Referenced by sha2_finish(), sha2_hmac_finish(), and sha2_starts().

12.99.2.4 unsigned char sha2_context::opad[64]

HMAC: outer padding

Definition at line 17 of file sha2.h.

Referenced by sha2_hmac_finish(), and sha2_hmac_starts().

12.99.2.5 unsigned long sha2_context::state[8]

intermediate digest state

Definition at line 13 of file sha2.h.

Referenced by sha2_finish(), sha2_process(), and sha2_starts().

12.99.2.6 unsigned long sha2_context::total[2]

number of bytes processed

Definition at line 12 of file sha2.h.

Referenced by sha2_finish(), sha2_starts(), and sha2_update().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha2.h](#)

12.100 sha4_context Struct Reference

SHA-512 context structure.

```
#include <sha4.h>
```

Data Fields

- unsigned int64 [total](#) [2]
- unsigned int64 [state](#) [8]
- unsigned char [buffer](#) [128]
- unsigned char [ipad](#) [128]
- unsigned char [opad](#) [128]
- [int is384](#)

12.100.1 Detailed Description

SHA-512 context structure.

Definition at line 18 of file sha4.h.

12.100.2 Field Documentation

12.100.2.1 unsigned char sha4_context::buffer[128]

data block being processed

Definition at line 22 of file sha4.h.

Referenced by sha4_update().

12.100.2.2 unsigned char sha4_context::ipad[128]

HMAC: inner padding

Definition at line 24 of file sha4.h.

Referenced by sha4_hmac_starts().

12.100.2.3 int sha4_context::is384

0 => SHA-512, else SHA-384

Definition at line 26 of file sha4.h.

Referenced by sha4_finish(), sha4_hmac_finish(), and sha4_starts().

12.100.2.4 unsigned char sha4_context::opad[128]

HMAC: outer padding

Definition at line 25 of file sha4.h.

Referenced by sha4_hmac_finish(), and sha4_hmac_starts().

12.100.2.5 unsigned int64 sha4_context::state[8]

intermediate digest state

Definition at line 21 of file sha4.h.

Referenced by sha4_finish(), sha4_process(), and sha4_starts().

12.100.2.6 unsigned int64 sha4_context::total[2]

number of bytes processed

Definition at line 20 of file sha4.h.

Referenced by sha4_finish(), sha4_starts(), and sha4_update().

The documentation for this struct was generated from the following file:

- [/home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha4.h](#)

12.101 syncList_s Struct Reference

```
#include <sync_list.h>
```

Data Fields

- `RWLOCK_T * lock`
- `MUTEX_T * giant_lock`
- `list_p list`
- `int size`

12.101.1 Detailed Description

Definition at line 26 of file `sync_list.h`.

12.101.2 Field Documentation

12.101.2.1 `MUTEX_T* syncList_s::giant_lock`

Definition at line 28 of file `sync_list.h`.

Referenced by `MC_SyncDelete()`, and `MC_SyncInit()`.

12.101.2.2 `list_p syncList_s::list`

Definition at line 30 of file `sync_list.h`.

Referenced by `syncListAddNode()`, `syncListDelete()`, `syncListFind()`, and `syncListRemove()`.

12.101.2.3 `RWLOCK_T* syncList_s::lock`

Definition at line 27 of file `sync_list.h`.

Referenced by `syncListAddNode()`, `syncListDelete()`, `syncListFind()`, `syncListInit()`, and `syncListRemove()`.

12.101.2.4 `int syncList_s::size`

Definition at line 31 of file `sync_list.h`.

The documentation for this struct was generated from the following file:

- `/home/dko/Projects/mobilec/trunk/src/mc_sync/sync_list.h`

12.102 syncListNode_s Struct Reference

```
#include <sync_list.h>
```

Data Fields

- `MUTEX_T * lock`
- `COND_T * cond`
- `SEMAPHORE_T * sem`
- `int id`
- `int signalled`

12.102.1 Detailed Description

Definition at line 16 of file `sync_list.h`.

12.102.2 Field Documentation

12.102.2.1 `COND_T* syncListNode_s::cond`

Definition at line 18 of file `sync_list.h`.

Referenced by `MC_CondBroadcast()`, `MC_CondSignal()`, `MC_CondWait()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

12.102.2.2 `int syncListNode_s::id`

Definition at line 20 of file `sync_list.h`.

Referenced by `MC_SyncInit()`, `syncListAddNode()`, `syncListDelete()`, and `syncListRemove()`.

12.102.2.3 `MUTEX_T* syncListNode_s::lock`

Definition at line 17 of file `sync_list.h`.

Referenced by `MC_CondBroadcast()`, `MC_CondReset()`, `MC_CondSignal()`, `MC_CondWait()`, `MC_MutexLock()`, `MC_MutexUnlock()`, `MC_SyncDelete()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

12.102.2.4 `SEMAPHORE_T* syncListNode_s::sem`

Definition at line 19 of file `sync_list.h`.

Referenced by `MC_SemaphorePost()`, `MC_SemaphoreWait()`, `syncListNodeDestroy()`, `syncListNodeInit()`, and `syncListNodeNew()`.

12.102.2.5 `int syncListNode_s::signalled`

Definition at line 21 of file `sync_list.h`.

Referenced by MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), and syncListNodeNew().

The documentation for this struct was generated from the following file:

- /home/dko/Projects/mobilec/trunk/src/mc_sync/[sync_list.h](#)

Chapter 13

File Documentation

13.1 /home/dko/Projects/mobilec/trunk/src/acc.c File Reference

```
#include <sys/socket.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <netdb.h>
#include <sys/un.h>
#include <unistd.h>
#include <sys/time.h>
#include <pthread.h>
#include "config.h"
#include <stdlib.h>
#include "include/acc.h"
#include "include/connection.h"
#include "include/data_structures.h"
#include "include/macros.h"
#include "include/mc_error.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/mtp_http.h"
#include "include/xml_parser.h"
#include "include/fipa_acl_envelope.h"
```

Defines

- #define [BACKLOG](#) 200
- #define [CONN_THREADS](#) 40

- #define [CONNECT_THREAD_EXIT\(\)](#)
- #define [BUFLLEN](#) 512
- #define [UDPPORT](#) 8866

Functions

- [acc_p acc_Initialize](#) (struct [mc_platform_s](#) *[mc_platform](#))
- [int acc_Destroy](#) ([acc_p](#) [acc](#))
- [void * acc_MessageHandlerThread](#) ([void](#) *[arg](#))
- [void * acc_Thread](#) ([void](#) *[arg](#))
- [void * acc_connection_Thread](#) ([void](#) *[arg](#))
- [void acc_Start](#) ([mc_platform_p](#) [mc_platform](#))
- [int auth_conn_rece_key](#) ([int](#) [sockfd](#), [char](#) *[peer_name](#), [int](#) *[nonce](#), [unsigned char](#) *[aes_key](#), [char](#) *[privkey](#), [char](#) *[known_host_filename](#))
- [void * listen_Thread](#) ([void](#) *[arg](#))
- [void * udplisten_Thread](#) ([void](#) *[arg](#))

13.1.1 Define Documentation

13.1.1.1 #define BACKLOG 200

Definition at line 65 of file [acc.c](#).

Referenced by [listen_Thread\(\)](#).

13.1.1.2 #define BUFLLEN 512

Definition at line 788 of file [acc.c](#).

Referenced by [udplisten_Thread\(\)](#).

13.1.1.3 #define CONN_THREADS 40

Definition at line 284 of file [acc.c](#).

Referenced by [acc_Thread\(\)](#).

13.1.1.4 #define CONNECT_THREAD_EXIT()

Value:

```
free(arg); \
    MUTEX_LOCK(&acc->conn_thread_lock); \
    acc->num_conn_threads--; \
    COND_SIGNAL(&acc->conn_thread_cond); \
    MUTEX_UNLOCK(&acc->conn_thread_lock); \
    THREAD_EXIT();
```

Definition at line 378 of file [acc.c](#).

Referenced by [acc_connection_Thread\(\)](#).

13.1.1.5 #define UDPPORT 8866

Definition at line 789 of file acc.c.

Referenced by udplisten_Thread().

13.1.2 Function Documentation**13.1.2.1 void* acc_connection_Thread (void * arg)**

Definition at line 388 of file acc.c.

References mc_platform_s::acc, fipa_agent_identifier_s::addresses, agent_mailbox_Post(), mc_platform_s::agent_queue, AGENT_UPDATE, CANCEL, CONNECT_THREAD_EXIT, connection_Destroy(), mtp_http_s::content, mtp_http_content_s::data, FIPA_ACL, fipa_acl_envelope_Destroy(), fipa_acl_envelope_New(), fipa_acl_message_Destroy(), fipa_acl_message_New(), fipa_acl_Parse(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_Parse(), fipa_message_string_Destroy(), fipa_message_string_New(), mtp_http_s::http_performative, HTTP_POST, HTTP_PUT, agent_s::mailbox, agent_s::mc_platform, mc_platform, fipa_message_string_s::message, message_s::message_body, message_Destroy(), message_New(), mtp_http_s::message_parts, mc_platform_s::message_queue, message_s::message_type, message_xml_parse(), MOBILE_AGENT, mtp_http_Destroy(), mtp_http_InitializeFromConnection(), mtp_http_New(), MXML_NO_CALLBACK, mxmlloadString(), N_UNDRSTD, fipa_agent_identifier_s::name, fipa_agent_identifier_set_s::num, fipa_acl_envelope_s::num_params, fipa_acl_envelope_s::params, fipa_message_string_s::parse, mc_platform_s::port, mc_platform_s::private_key, QUER_IF, QUER_REF, RELAY, REQUEST, RETURN_MSG, fipa_url_s::str, SUBSCRIBE, mtp_http_s::target, fipa_acl_Param_s::to, fipa_url_sequence_s::urls, and message_s::xml_root.

Referenced by acc_Thread().

13.1.2.2 int acc_Destroy (acc_p acc)

Definition at line 93 of file acc.c.

References MC_SUCCESS.

Referenced by mc_platform_Destroy().

13.1.2.3 acc_p acc_Initialize (struct mc_platform_s * mc_platform)

Definition at line 68 of file acc.c.

References COND_INIT, COND_T, MUTEX_INIT, and MUTEX_T.

Referenced by mc_platform_Initialize().

13.1.2.4 void* acc_MessageHandlerThread (void * arg)

Definition at line 105 of file acc.c.

References mc_platform_s::acc, agent_Initialize(), mc_platform_s::agent_queue, agent_s::agent_status, AGENT_UPDATE, mc_platform_s::ams, CANCEL, COND_BROADCAST, COND_WAIT, agent_s::datastate, FIPA_ACL, mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, agent_s::lock, MC_AGENT_NEUTRAL, mc_platform, MC_RECV_AGENT, MC_RECV_MESSAGE,

MC_RECV_RETURN, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, message_Destroy(), mc_platform_s::message_queue, message_Send(), message_s::message_type, MOBILE_AGENT, MUTEX_LOCK, MUTEX_UNLOCK, N_UNDRSTD, agent_s::name, agent_datastate_s::persistent, QUER_IF, QUER_REF, mc_platform_s::quit, mc_platform_s::quit_lock, RELAY, REQUEST, RETURN_MSG, SUBSCRIBE, THREAD_EXIT, and message_s::to_address.

Referenced by acc_Start().

13.1.2.5 void acc_Start (mc_platform_p mc_platform)

Definition at line 564 of file acc.c.

References mc_platform_s::acc, acc_MessageHandlerThread(), acc_Thread(), listen_Thread(), MC_THREAD_ACC, mc_platform_s::stack_size, THREAD_CREATE, and udplisten_Thread().

Referenced by mc_platform_Initialize().

13.1.2.6 void* acc_Thread (void * arg)

Definition at line 287 of file acc.c.

References mc_platform_s::acc, acc_connection_Thread(), COND_BROADCAST, COND_WAIT, CONN_THREADS, mc_platform_s::connection_queue, mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, mc_platform, MC_RECV_CONNECTION, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::quit, mc_platform_s::quit_lock, THREAD_CREATE, THREAD_DETACH, THREAD_EXIT, and THREAD_T.

Referenced by acc_Start().

13.1.2.7 int auth_conn_rece_key (int sockfd, char * peer_name, int * nonce, unsigned char * aes_key, char * privkey, char * known_host_filename)

Definition at line 613 of file acc.c.

References read_known_host_file(), and reply_migration_process().

Referenced by listen_Thread().

13.1.2.8 void* listen_Thread (void * arg)

Definition at line 652 of file acc.c.

References mc_platform_s::acc, connection_s::addr, connection_s::AES_key, mc_platform_s::agency, auth_conn_rece_key(), BACKLOG, connection_s::clientfd, COND_BROADCAST, connection_s::connect_id, connection_New(), mc_platform_s::connection_queue, agency_s::known_host_filename, mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, connection_s::nonce, mc_platform_s::port, mc_platform_s::private_key, connection_s::remote_hostname, connection_s::serverfd, SOCKET_ERROR, mc_platform_s::sockfd, and THREAD_EXIT.

Referenced by acc_Start().

13.1.2.9 void* udplisten_Thread (void * *arg*)

Definition at line 792 of file acc.c.

References `buf`, `BUFLen`, `mc_platform_s::hostname`, `mc_platform`, `PACKAGE_VERSION`, `mc_platform_s::port`, and `UDPPORT`.

Referenced by `acc_Start()`.

13.2 /home/dko/Projects/mobilec/trunk/src/agent.c File Reference

```
#include <unistd.h>
#include "config.h"
#include <embedch.h>
#include "include/libmc.h"
#include "include/agent.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/agent_lib.h"
#include "include/interpreter_variable_data.h"
#include "include/xml_parser.h"
```

Functions

- [int agent_AddPersistentVariable](#) ([agent_p](#) agent, [int](#) task_num, const char *var_name)
- [agent_p agent_Copy](#) (const [agent_p](#) agent)
- [agent_p agent_New](#) (void)
- [agent_p agent_NewBinary](#) (struct [mc_platform_s](#) *mc_platform)
- [agent_p agent_Initialize](#) (struct [mc_platform_s](#) *mc_platform, [message_p](#) message, [int](#) id)
- [int agent_Destroy](#) ([agent_p](#) agent)
- void [agent_RunChScript](#) ([agent_p](#) agent, [mc_platform_p](#) mc_platform)
- void * [agent_ChScriptInitVar](#) ([ChInterp_t](#) *interp)
- void * [agent_RunChScriptThread](#) (void *ChAgent)

13.2.1 Function Documentation

13.2.1.1 int agent_AddPersistentVariable (agent_p agent, int task_num, const char * var_name)

Definition at line 52 of file agent.c.

References [agent_s::agent_interp](#), [agent_task_s::agent_variable_list](#), [interpreter_variable_data_s::array_dim](#), [interpreter_variable_data_s::array_extent](#), [CH_DATATYPE_SIZE](#), [CHECK_NULL](#), [interpreter_variable_data_s::data](#), [interpreter_variable_data_s::data_type](#), [agent_s::datastate](#), [MC_ERR](#), [agent_s::name](#), [interpreter_variable_data_s::name](#), [interpreter_variable_data_s::size](#), [size](#), [agent_datastate_s::task_progress](#), and [agent_datastate_s::tasks](#).

13.2.1.2 void* agent_ChScriptInitVar (ChInterp_t * interp)

Definition at line 492 of file agent.c.

References [MC_AclAddReceiver_chdl\(\)](#), [MC_AclAddReplyTo_chdl\(\)](#), [MC_AclDestroy_chdl\(\)](#), [MC_AclNew_chdl\(\)](#), [MC_AclPost_chdl\(\)](#), [MC_AclReply_chdl\(\)](#), [MC_AclRetrieve_chdl\(\)](#), [MC_AclSend_chdl\(\)](#), [MC_AclSetContent_chdl\(\)](#), [MC_AclSetConversationID_chdl\(\)](#), [MC_AclSetPerformative_chdl\(\)](#), [MC_AclSetProtocol_chdl\(\)](#), [MC_AclSetSender_chdl\(\)](#), [MC_AclWaitRetrieve_chdl\(\)](#), [MC_AddAgent_chdl\(\)](#), [MC_AgentVariableRetrieve_chdl\(\)](#), [MC_AgentVariableSave_chdl\(\)](#), [MC_Barrier_chdl\(\)](#), [MC_BarrierDelete_chdl\(\)](#), [MC_BarrierInit_chdl\(\)](#), and [MC_CallAgentFunc_chdl\(\)](#).

MC_ComposeAgent_chdl(), MC_ComposeAgentS_chdl(), MC_CondBroadcast_chdl(), MC_CondReset_chdl(), MC_CondSignal_chdl(), MC_CondWait_chdl(), MC_DeleteAgent_chdl(), MC_DeleteAgentWG_chdl(), MC_DeregisterService_chdl(), MC_DestroyServiceSearchResult_chdl(), MC_End_chdl(), MC_FindAgentByID_chdl(), MC_FindAgentByName_chdl(), MC_GetAgentID_chdl(), MC_GetAgentName_chdl(), MC_GetAgentStatus_chdl(), MC_GetAgentXMLString_chdl(), MC_GetTimeOfDay_chdl(), MC_HaltAgency_chdl(), MC_MigrateAgent_chdl(), MC_MutexLock_chdl(), MC_MutexUnlock_chdl(), MC_PrintAgentCode_chdl(), MC_RegisterService_chdl(), MC_ResumeAgency_chdl(), MC_RetrieveAgent_chdl(), MC_RetrieveAgentCode_chdl(), MC_SaveData_chdl(), MC_SearchForService_chdl(), MC_SemaphorePost_chdl(), MC_SemaphoreWait_chdl(), MC_SendAgentMigrationMessage_chdl(), MC_SendAgentMigrationMessageFile_chdl(), MC_SendSteerCommand_chdl(), MC_SetAgentStatus_chdl(), MC_SetDefaultAgentStatus_chdl(), MC_SyncDelete_chdl(), MC_SyncInit_chdl(), MC_TerminateAgent_chdl(), and MC_TerminateAgentWG_chdl().

Referenced by AP_QUEUE_STD_DEFN_TEMPLATE(), and mc_platform_Initialize().

13.2.1.3 agent_p agent_Copy (const agent_p agent)

Definition at line 134 of file agent.c.

References agent_datastate_Copy(), agent_s::agent_interp, agent_mailbox_New(), agent_s::agent_persistent, agent_s::agent_status, agent_s::agent_type, agent_s::arrival_time, agent_s::datastate, agent_s::home, agent_s::home_port, agent_s::id, agent_s::lock, agent_s::mailbox, MUTEX_INIT, MUTEX_LOCK, MUTEX_T, agent_s::name, agent_s::orphan, agent_s::owner, agent_s::return_data, and agent_s::run_lock.

Referenced by MC_CopyAgent().

13.2.1.4 int agent_Destroy (agent_p agent)

Definition at line 414 of file agent.c.

References agent_datastate_Destroy(), agent_s::agent_interp, agent_mailbox_Destroy(), agent_s::agent_status, agent_s::datastate, agent_s::home, mc_platform_s::interpreter_queue, agent_s::lock, agent_s::mailbox, MC_AGENT_NEUTRAL, agent_s::mc_platform, MC_SUCCESS, MUTEX_DESTROY, MUTEX_LOCK, agent_s::name, agent_s::owner, agent_s::run_lock, agent_s::sender, and agent_s::wg_code.

Referenced by agent_Initialize().

13.2.1.5 agent_p agent_Initialize (struct mc_platform_s * mc_platform, message_p message, int id)

Definition at line 293 of file agent.c.

References agent_s::agent_address, agent_datastate_New(), agent_Destroy(), agent_mailbox_New(), agent_s::agent_pipe_active, agent_s::agent_pipe_ready_to_read, agent_s::agent_ready_to_send, agent_s::agent_script_ready, agent_s::agent_status, agent_s::agent_thread_id, agent_s::agent_type, message_s::agent_xml_flag, agent_xml_parse(), agent_s::arrival_time, agent_s::datastate, mc_platform_s::default_agentstatus, mc_platform_s::err, agent_s::home, mc_platform_s::hostname, agent_s::id, agent_s::lock, agent_s::mailbox, MC_ERR_PARSE, agent_s::mc_platform, MC_REMOTE_AGENT, MC_RETURN_AGENT, MC_WAIT_CH, message_s::message_type, MOBILE_AGENT, MUTEX_DESTROY, MUTEX_INIT, MUTEX_T, agent_s::orphan, mc_platform_s::port, RETURN_MSG, agent_s::run_lock, agent_s::sender, agent_datastate_s::xml_agent_root, message_s::xml_payload, message_s::xml_root, and agent_datastate_s::xml_root.

Referenced by `acc_MessageHandlerThread()`, and `MC_SendAgentMigrationMessageFile()`.

13.2.1.6 `agent_p agent_New (void)`

Definition at line 197 of file `agent.c`.

References `agent_s::lock`, `MUTEX_INIT`, `MUTEX_NEW`, and `agent_s::run_lock`.

Referenced by `MC_ComposeAgentS()`.

13.2.1.7 `agent_p agent_NewBinary (struct mc_platform_s * mc_platform)`

Definition at line 220 of file `agent.c`.

References `agent_s::agent_address`, `agent_mailbox_New()`, `agent_s::agent_pipe_active`, `agent_s::agent_pipe_ready_to_read`, `agent_s::agent_ready_to_send`, `agent_s::agent_script_ready`, `agent_s::agent_status`, `agent_s::agent_thread_id`, `agent_s::arrival_time`, `agent_s::binary`, `agent_s::home`, `mc_platform_s::hostname`, `agent_s::id`, `agent_s::lock`, `agent_s::mailbox`, `MC_AGENT_ACTIVE`, `agent_s::mc_platform`, `MUTEX_INIT`, `MUTEX_T`, `agent_s::orphan`, `mc_platform_s::port`, `agent_s::run_lock`, and `agent_s::sender`.

Referenced by `MC_AddStationaryAgent()`.

13.2.1.8 `void agent_RunChScript (agent_p agent, mc_platform_p mc_platform)`

Definition at line 454 of file `agent.c`.

References `agent_RunChScriptThread()`, `agent_s::agent_status`, `agent_s::agent_thread`, `MC_AGENT_ACTIVE`, `agent_s::mc_platform`, `MC_THREAD_AGENT`, `mc_platform_s::stack_size`, and `THREAD_CREATE`.

Referenced by `ams_ManageAgentList()`.

13.2.1.9 `void* agent_RunChScriptThread (void * ChAgent)`

Definition at line 926 of file `agent.c`.

References `agent_s::agent_interp`, `mc_platform_s::ams`, `COND_SIGNAL`, `agent_s::datastate`, `agent_s::id`, `mc_platform_s::interp_options`, `mc_platform_s::interpreter_queue`, `interpreter_queue_CreateRetrieve()`, `interpreter_variable_data_Destroy()`, `interpreter_variable_data_Initialize()`, `interpreter_variable_data_InitializeFromAgent()`, `MC_AGENT_NEUTRAL`, `MC_EXEC_AGENT`, `agent_s::mc_platform`, `mc_platform`, `MC_RETURN_AGENT`, `mc_platform_s::MC_signal`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `MC_WAIT_FINISHED`, `MC_WAIT_MESSGSEND`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `agent_s::name`, `SIGNAL`, and `agent_datastate_s::task_progress`.

Referenced by `agent_RunChScript()`.

13.3 /home/dko/Projects/mobilec/trunk/src/agent_datastate.c File Reference

```
#include "config.h"
#include <mxml.h>
#include "include/agent_datastate.h"
#include "include/macros.h"
#include "include/mc_error.h"
```

Functions

- [agent_datastate_p agent_datastate_Copy](#) (const [agent_datastate_p](#) datastate)
- [agent_datastate_p agent_datastate_New](#) (void)
- [int agent_datastate_Destroy](#) ([agent_datastate_p](#) agent_datastate)

13.3.1 Function Documentation

13.3.1.1 [agent_datastate_p agent_datastate_Copy](#) (const [agent_datastate_p](#) datastate)

Definition at line 47 of file `agent_datastate.c`.

References `agent_datastate_s::agent_code`, `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_datastate_New()`, `agent_task_Copy()`, `agent_datastate_s::init_agent_status`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::persistent`, `agent_datastate_s::return_data`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `agent_Copy()`.

13.3.1.2 [int agent_datastate_Destroy](#) ([agent_datastate_p](#) agent_datastate)

Definition at line 136 of file `agent_datastate.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_task_Destroy()`, `MC_SUCCESS`, `mxmlDelete()`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::tasks`, and `agent_datastate_s::xml_root`.

Referenced by `agent_Destroy()`.

13.3.1.3 [agent_datastate_p agent_datastate_New](#) (void)

Definition at line 115 of file `agent_datastate.c`.

References `agent_datastate_s::agent_code`, `CHECK_NULL`, `agent_datastate_s::init_agent_status`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::persistent`, `agent_datastate_s::progress_modifier`, `agent_datastate_s::return_data`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, `agent_datastate_s::xml_agent_root`, and `agent_datastate_s::xml_root`.

Referenced by `agent_datastate_Copy()`, `agent_Initialize()`, and `MC_ComposeAgentS()`.

13.4 /home/dko/Projects/mobilec/trunk/src/agent_mailbox.c File Reference

```
#include "config.h"
#include "include/data_structures.h"
```

Functions

- [agent_mailbox_p agent_mailbox_New](#) (void)
- [agent_mailbox_p agent_mailbox_Copy](#) (agent_mailbox_p src)
- [int agent_mailbox_Destroy](#) (agent_mailbox_t *mailbox)
- [int agent_mailbox_Post](#) (agent_mailbox_p mailbox, fipa_acl_message_t *message)
- [fipa_acl_message_t * agent_mailbox_Retrieve](#) (agent_mailbox_p mailbox)
- [fipa_acl_message_t * agent_mailbox_WaitRetrieve](#) (agent_mailbox_p mailbox)

13.4.1 Function Documentation

13.4.1.1 agent_mailbox_p agent_mailbox_Copy (agent_mailbox_p src)

Definition at line 21 of file agent_mailbox.c.

References agent_mailbox_New(), and agent_mailbox_s::mail_queue.

13.4.1.2 int agent_mailbox_Destroy (agent_mailbox_t * mailbox)

Definition at line 29 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_Destroy().

13.4.1.3 agent_mailbox_p agent_mailbox_New (void)

Definition at line 12 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_Copy(), agent_Initialize(), agent_mailbox_Copy(), and agent_NewBinary().

13.4.1.4 int agent_mailbox_Post (agent_mailbox_p mailbox, fipa_acl_message_t * message)

Definition at line 38 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by acc_connection_Thread(), and MC_AclPost().

13.4.1.5 fipa_acl_message_t* agent_mailbox_Retrieve (agent_mailbox_p mailbox)

Definition at line 44 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_mailbox_WaitRetrieve(), and MC_AclRetrieve().

13.4.1.6 fipa_acl_message_t* agent_mailbox_WaitRetrieve (agent_mailbox_p mailbox)

Definition at line 49 of file agent_mailbox.c.

References agent_mailbox_Retrieve(), COND_WAIT, agent_mailbox_s::mail_queue, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_AclWaitRetrieve().

13.5 /home/dko/Projects/mobilec/trunk/src/agent_return_data.c File Reference

```
#include "config.h"
#include "include/interpreter_variable_data.h"
#include "include/agent.h"
```

Functions

- [interpreter_variable_data_p interpreter_variable_data_New](#) (void)
- [interpreter_variable_data_p interpreter_variable_data_InitializeFromAgent](#) ([agent_p](#) agent)
- [interpreter_variable_data_p interpreter_variable_data_Initialize](#) ([agent_p](#) agent, const char *varname)
- [int interpreter_variable_data_Destroy](#) ([interpreter_variable_data_p](#) agent_variable_data)
- [interpreter_variable_data_p interpreter_variable_data_Copy](#) ([interpreter_variable_data_p](#) src)

13.5.1 Function Documentation

13.5.1.1 [interpreter_variable_data_p interpreter_variable_data_Copy](#) ([interpreter_variable_data_p](#) src)

Definition at line 235 of file agent_return_data.c.

References [interpreter_variable_data_s::array_dim](#), [interpreter_variable_data_s::array_extent](#), [interpreter_variable_data_s::data](#), [interpreter_variable_data_s::data_type](#), [interpreter_variable_data_New\(\)](#), [interpreter_variable_data_s::name](#), and [interpreter_variable_data_s::size](#).

Referenced by [agent_task_Copy\(\)](#).

13.5.1.2 [int interpreter_variable_data_Destroy](#) ([interpreter_variable_data_p](#) agent_variable_data)

Definition at line 216 of file agent_return_data.c.

References [interpreter_variable_data_s::array_extent](#), [interpreter_variable_data_s::data](#), [MC_SUCCESS](#), and [interpreter_variable_data_s::name](#).

Referenced by [agent_RunChScriptThread\(\)](#), and [agent_task_Destroy\(\)](#).

13.5.1.3 [interpreter_variable_data_p interpreter_variable_data_Initialize](#) ([agent_p](#) agent, const char * varname)

Definition at line 145 of file agent_return_data.c.

References [agent_s::agent_interp](#), [interpreter_variable_data_s::array_dim](#), [interpreter_variable_data_s::array_extent](#), [CH_DATATYPE_SIZE](#), [CHECK_NULL](#), [interpreter_variable_data_s::data](#), [interpreter_variable_data_s::data_type](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), [interpreter_variable_data_s::name](#), [agent_s::run_lock](#), [interpreter_variable_data_s::size](#), and [size](#).

Referenced by [agent_RunChScriptThread\(\)](#).

13.5.1.4 `interpreter_variable_data_p` `interpreter_variable_data_InitializeFromAgent (agent_p agent)`

Definition at line 61 of file `agent_return_data.c`.

References `agent_s::agent_interp`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `CH_DATATYPE_SIZE`, `CHECK_NULL`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `agent_s::datastate`, `interpreter_variable_data_s::name`, `interpreter_variable_data_s::size`, `size`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_RunChScriptThread()`.

13.5.1.5 `interpreter_variable_data_p` `interpreter_variable_data_New (void)`

Definition at line 46 of file `agent_return_data.c`.

References `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `CHECK_NULL`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `interpreter_variable_data_s::name`, and `interpreter_variable_data_s::size`.

Referenced by `agent_xml_parse__data()`, `interpreter_variable_data_Copy()`, and `MC_SaveData_chdl()`.

13.6 /home/dko/Projects/mobilec/trunk/src/agent_task.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/agent_task.h"
#include "include/mc_error.h"
```

Functions

- [agent_task_p agent_task_New](#) (void)
- [agent_task_p agent_task_Copy](#) (agent_task_p task)
- [int agent_task_Destroy](#) (agent_task_p agent_task)

13.6.1 Function Documentation

13.6.1.1 agent_task_p agent_task_Copy (agent_task_p task)

Definition at line 66 of file agent_task.c.

References [agent_task_s::agent_return_data](#), [agent_task_s::agent_variable_list](#), [agent_task_s::code_id](#), [agent_task_s::init_agent_status](#), [interpreter_variable_data_Copy\(\)](#), [ListAdd\(\)](#), [ListSearch\(\)](#), [agent_task_s::num_saved_variables](#), [agent_task_s::number_of_elements](#), [agent_task_s::persistent](#), [agent_task_s::saved_variables](#), [agent_task_s::server_name](#), [agent_task_s::size_of_element_array](#), and [agent_task_s::var_name](#).

Referenced by [agent_datastate_Copy\(\)](#).

13.6.1.2 int agent_task_Destroy (agent_task_p agent_task)

Definition at line 132 of file agent_task.c.

References [agent_task_s::agent_return_data](#), [agent_task_s::agent_variable_list](#), [agent_task_s::code_id](#), [interpreter_variable_data_Destroy\(\)](#), [MC_SUCCESS](#), [agent_task_s::saved_variables](#), [agent_task_s::server_name](#), and [agent_task_s::var_name](#).

Referenced by [agent_datastate_Destroy\(\)](#).

13.6.1.3 agent_task_p agent_task_New (void)

Definition at line 48 of file agent_task.c.

References [agent_task_s::agent_variable_list](#), [agent_task_s::num_saved_variables](#), and [agent_task_s::saved_variables](#).

Referenced by [agent_xml_parse__tasks\(\)](#), and [MC_ComposeAgentS\(\)](#).

13.7 /home/dko/Projects/mobilec/trunk/src/ams.c File Reference

```
#include "config.h"
#include "include/ams.h"
#include "include/agent.h"
#include "include/data_structures.h"
#include "include/mc_platform.h"
```

Functions

- [int `ams_Destroy`](#) (`ams_p ams`)
- `ams_p` [ams_Initialize](#) (`mc_platform_p mc_platform`)
- void [ams_Print](#) (`ams_p ams`)
- [int `ams_ManageAgentList`](#) (`ams_p ams`)
- void [ams_Start](#) (`mc_platform_p mc_platform`)
- void * [ams_Thread](#) (void *arg)

13.7.1 Function Documentation

13.7.1.1 `int ams_Destroy (ams_p ams)`

Definition at line 46 of file `ams.c`.

References `COND_DESTROY`, `MC_SUCCESS`, and `MUTEX_DESTROY`.

Referenced by `mc_platform_Destroy()`.

13.7.1.2 `ams_p ams_Initialize (mc_platform_p mc_platform)`

Definition at line 57 of file `ams.c`.

References `CHECK_NULL`, `COND_INIT`, `COND_T`, `MUTEX_INIT`, and `MUTEX_T`.

Referenced by `mc_platform_Initialize()`.

13.7.1.3 `int ams_ManageAgentList (ams_p ams)`

Definition at line 116 of file `ams.c`.

References `agent_RunChScript()`, `agent_s::agent_status`, `agent_s::binary`, `ListSearch()`, `agent_s::lock`, `MC_AGENT_ACTIVE`, `MC_AGENT_NEUTRAL`, `MC_TerminateAgent()`, `MC_WAIT_CH`, `MC_WAIT_FINISHED`, `MC_WAIT_MESSGSEND`, `message_Destroy()`, `message_InitializeFromAgent()`, `message_New()`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `agent_s::name`, `agent_s::orphan`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, and `agent_s::run_lock`.

Referenced by `ams_Thread()`.

13.7.1.4 `void ams_Print (ams_p ams)`

Definition at line 84 of file `ams.c`.

References `agent_s::agent_status`, `agent_s::connect_id`, `agent_s::id`, `ListSearch()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

13.7.1.5 void ams_Start (mc_platform_p mc_platform)

Definition at line 223 of file `ams.c`.

References `mc_platform_s::ams`, `ams_Thread()`, `MC_THREAD_AMS`, `mc_platform_s::stack_size`, and `THREAD_CREATE`.

Referenced by `mc_platform_Initialize()`.

13.7.1.6 void* ams_Thread (void * arg)

Definition at line 254 of file `ams.c`.

References `mc_platform_s::ams`, `ams_ManageAgentList()`, `COND_BROADCAST`, `COND_WAIT`, `mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, and `THREAD_EXIT`.

Referenced by `ams_Start()`.

13.8 /home/dko/Projects/mobilec/trunk/src/barrier.c File Reference

```
#include "config.h"
#include "include/barrier.h"
#include "include/mc_error.h"
```

Functions

- [barrier_node_p barrier_node_Initialize](#) ([int id](#), [int num_registered](#))
- [int barrier_node_Destroy](#) ([barrier_node_p node](#))
- [int barrier_queue_Add](#) ([barrier_queue_p list](#), [barrier_node_p node](#))
- [int barrier_queue_Delete](#) ([int id](#), [barrier_queue_p list](#))
- [int barrier_queue_Destroy](#) ([barrier_queue_p queue](#))
- [barrier_node_p barrier_queue_Get](#) ([barrier_queue_p list](#), [int id](#))
- [barrier_queue_p barrier_queue_New](#) ([void](#))
- [barrier_node_p barrier_queue_Pop](#) ([barrier_queue_p queue](#))

13.8.1 Function Documentation

13.8.1.1 [int barrier_node_Destroy](#) ([barrier_node_p node](#))

Definition at line 70 of file barrier.c.

References [barrier_node_s::cond](#), [COND_DESTROY](#), [barrier_node_s::lock](#), [MC_SUCCESS](#), and [MUTEX_DESTROY](#).

Referenced by [barrier_queue_Delete\(\)](#), and [barrier_queue_Destroy\(\)](#).

13.8.1.2 [barrier_node_p barrier_node_Initialize](#) ([int id](#), [int num_registered](#))

Definition at line 45 of file barrier.c.

References [CHECK_NULL](#), [barrier_node_s::cond](#), [COND_INIT](#), [COND_T](#), [barrier_node_s::id](#), [barrier_node_s::lock](#), [MUTEX_INIT](#), [MUTEX_T](#), [node](#), [barrier_node_s::num_registered](#), and [barrier_node_s::num_waiting](#).

Referenced by [MC_BarrierInit\(\)](#).

13.8.1.3 [int barrier_queue_Add](#) ([barrier_queue_p list](#), [barrier_node_p node](#))

Definition at line 87 of file barrier.c.

References [DATA](#), [barrier_node_s::id](#), [barrier_queue_s::list](#), [ListAdd\(\)](#), [list_s::listhead](#), [barrier_queue_s::lock](#), [MC_SUCCESS](#), [MC_WARN_DUPLICATE](#), [listNode_s::next](#), [listNode_s::node_data](#), [RWLOCK_WRLock](#), [RWLOCK_WRunlock](#), and [barrier_queue_s::size](#).

Referenced by [MC_BarrierInit\(\)](#).

13.8.1.4 [int barrier_queue_Delete](#) ([int id](#), [barrier_queue_p list](#))

Definition at line 111 of file barrier.c.

References `barrier_node_Destroy()`, `barrier_node_s::id`, `barrier_queue_s::list`, `ListDelete()`, `ListSearch()`, `barrier_queue_s::lock`, `MC_ERR_NOT_FOUND`, `MC_SUCCESS`, `RWLOCK_WRLOCK`, `RWLOCK_WRUNLOCK`, `barrier_queue_s::size`, and `list_s::size`.

Referenced by `MC_BarrierDelete()`.

13.8.1.5 `int barrier_queue_Destroy (barrier_queue_p queue)`

Definition at line 131 of file `barrier.c`.

References `barrier_node_Destroy()`, `barrier_queue_Pop()`, `barrier_queue_s::list`, `ListTerminate()`, `barrier_queue_s::lock`, `MC_SUCCESS`, `node`, and `RWLOCK_DESTROY`.

Referenced by `mc_platform_Destroy()`.

13.8.1.6 `barrier_node_p barrier_queue_Get (barrier_queue_p list, int id)`

Definition at line 145 of file `barrier.c`.

References `barrier_queue_s::list`, `list_s::listhead`, `barrier_queue_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_RDLOCK`, and `RWLOCK_RDUNLOCK`.

Referenced by `MC_Barrier()`, and `MC_BarrierInit()`.

13.8.1.7 `barrier_queue_p barrier_queue_New (void)`

Definition at line 162 of file `barrier.c`.

References `CHECK_NULL`, `barrier_queue_s::list`, `ListInitialize()`, `barrier_queue_s::lock`, `RWLOCK_INIT`, and `RWLOCK_T`.

Referenced by `mc_platform_Initialize()`.

13.8.1.8 `barrier_node_p barrier_queue_Pop (barrier_queue_p queue)`

Definition at line 176 of file `barrier.c`.

References `barrier_queue_s::list`, `ListPop()`, and `node`.

Referenced by `barrier_queue_Destroy()`.

13.9 /home/dko/Projects/mobilec/trunk/src/cmd_prompt.c File Reference

```
#include <stdio.h>
#include <unistd.h>
#include "config.h"
#include <stdlib.h>
#include <string.h>
#include "include/cmd_prompt.h"
#include "include/commands.h"
```

Functions

- [cmd_prompt_p cmd_prompt_Initialize \(mc_platform_p mc_platform\)](#)
- [int cmd_prompt_Destroy \(cmd_prompt_p cmd_prompt\)](#)
- [void cmd_prompt_Start \(mc_platform_p mc_platform\)](#)
- [void * cmd_prompt_Thread \(void *arg\)](#)
- [int split_string \(char ***args, const char *buf\)](#)
- [int process_command \(command_t *cmd\)](#)
- [int exec_command \(command_t cmd, mc_platform_p global\)](#)
- [int dealloc_command \(command_t *cmd\)](#)
- [int handler_QUIT \(void *arg, mc_platform_p global\)](#)
- [int handler_HELP \(void *arg, mc_platform_p global\)](#)
- [int handler_SEND \(void *arg, mc_platform_p global\)](#)
- [int handler_PRINT_CONNECTLIST \(void *arg, mc_platform_p global\)](#)
- [int handler_PRINTLIST_MESSAGE \(void *arg, mc_platform_p global\)](#)
- [int handler_PRINTLIST_AGENTS \(void *arg, mc_platform_p global\)](#)
- [int handler_FLUSH_AGENTS \(void *arg, mc_platform_p global\)](#)
- [int handler_COMPOSE_SEND \(void *arg, mc_platform_p global\)](#)

13.9.1 Function Documentation

13.9.1.1 [int cmd_prompt_Destroy \(cmd_prompt_p *cmd_prompt*\)](#)

Definition at line 129 of file cmd_prompt.c.

References [MC_SUCCESS](#).

Referenced by [mc_platform_Destroy\(\)](#).

13.9.1.2 [cmd_prompt_p cmd_prompt_Initialize \(mc_platform_p *mc_platform*\)](#)

Definition at line 121 of file cmd_prompt.c.

Referenced by [mc_platform_Initialize\(\)](#).

13.9.1.3 void cmd_prompt_Start (mc_platform_p mc_platform)

Definition at line 136 of file cmd_prompt.c.

References mc_platform_s::cmd_prompt, cmd_prompt_Thread(), MC_THREAD_CP, mc_platform_s::stack_size, cmd_prompt_s::thread, and THREAD_CREATE.

Referenced by mc_platform_Initialize().

13.9.1.4 void* cmd_prompt_Thread (void * arg)

Definition at line 168 of file cmd_prompt.c.

References command_s::args, buf, dealloc_command(), exec_command(), command_s::index, mc_platform, command_s::num_args, process_command(), and split_string().

Referenced by cmd_prompt_Start().

13.9.1.5 int dealloc_command (command_t * cmd)

Definition at line 308 of file cmd_prompt.c.

References command_s::args, and command_s::num_args.

Referenced by cmd_prompt_Thread().

13.9.1.6 int exec_command (command_t cmd, mc_platform_p global)

Definition at line 292 of file cmd_prompt.c.

References command_s::args, cmd_handlers, command_s::index, and command_s::num_args.

Referenced by cmd_prompt_Thread().

13.9.1.7 int handler_COMPOSE_SEND (void * arg, mc_platform_p global)

Definition at line 409 of file cmd_prompt.c.

References mc_platform_s::agency, command_s::args, command_descriptions, agency_s::hostName, MC_AddAgent(), MC_ComposeAgentFromFile(), and command_s::num_args.

13.9.1.8 int handler_FLUSH_AGENTS (void * arg, mc_platform_p global)

Definition at line 403 of file cmd_prompt.c.

References mc_platform_s::agent_queue, and agent_queue_Flush().

13.9.1.9 int handler_HELP (void * arg, mc_platform_p global)

Definition at line 332 of file cmd_prompt.c.

References command_s::args, command_cmds, command_descriptions, and command_s::num_args.

13.9.1.10 int handler_PRINT_CONNECTLIST (void * *arg*, mc_platform_p *global*)

Definition at line 385 of file cmd_prompt.c.

References mc_platform_s::connection_queue.

13.9.1.11 int handler_PRINTLIST_AGENTS (void * *arg*, mc_platform_p *global*)

Definition at line 397 of file cmd_prompt.c.

References mc_platform_s::agent_queue.

13.9.1.12 int handler_PRINTLIST_MESSAGE (void * *arg*, mc_platform_p *global*)

Definition at line 391 of file cmd_prompt.c.

References mc_platform_s::message_queue.

13.9.1.13 int handler_QUIT (void * *arg*, mc_platform_p *global*)

Definition at line 323 of file cmd_prompt.c.

References COND_BROADCAST, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::quit, mc_platform_s::quit_cond, and mc_platform_s::quit_lock.

13.9.1.14 int handler_SEND (void * *arg*, mc_platform_p *global*)

Definition at line 370 of file cmd_prompt.c.

References command_s::args, command_descriptions, MC_SendAgentMigrationMessageFile(), and command_s::num_args.

13.9.1.15 int process_command (command_t * *cmd*)

Definition at line 274 of file cmd_prompt.c.

References command_s::args, command_cmds, command_s::index, and command_s::num_args.

Referenced by cmd_prompt_Thread().

13.9.1.16 int split_string (char * *args*, const char * *buf*)**

Definition at line 226 of file cmd_prompt.c.

References int.

Referenced by cmd_prompt_Thread().

13.10 /home/dko/Projects/mobilec/trunk/src/connection.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/connection.h"
#include "include/mc_error.h"
#include "include/macros.h"
```

Functions

- [int connection_Destroy](#) ([connection_p](#) connection)
- [connection_p connection_New](#) (void)
- [connection_p connection_Copy](#) ([connection_p](#) connection)

13.10.1 Function Documentation

13.10.1.1 [connection_p connection_Copy](#) ([connection_p](#) *connection*)

Definition at line 80 of file connection.c.

References [connection_s::addr](#), [connection_s::clientfd](#), [connection_s::connect_id](#), [connection_New\(\)](#), [connection_s::remote_hostname](#), and [connection_s::serverfd](#).

13.10.1.2 [int connection_Destroy](#) ([connection_p](#) *connection*)

Definition at line 48 of file connection.c.

References [connection_s::clientfd](#), [MC_SUCCESS](#), and [connection_s::remote_hostname](#).

Referenced by [acc_connection_Thread\(\)](#).

13.10.1.3 [connection_p connection_New](#) (void)

Definition at line 69 of file connection.c.

Referenced by [connection_Copy\(\)](#), and [listen_Thread\(\)](#).

13.11 /home/dko/Projects/mobilec/trunk/src/data_structures.c File Reference

```
#include "config.h"
#include "include/ap_queue_template.h"
#include "include/data_structures.h"
#include "include/interpreter_variable_data.h"
#include "include/agent.h"
```

Functions

- [AP_QUEUE_SEARCH_TEMPLATE](#) (AP_QUEUE_REMOVE_TEMPLATE(connection_queue, [Search](#), AP_QUEUE_REMOVE_TEMPLATE(connection, AP_QUEUE_REMOVE_TEMPLATE(int,([node](#)->connect_id==key)))
- [AP_QUEUE_SEARCH_TEMPLATE](#) (AP_QUEUE_REMOVE_TEMPLATE(agent_variable_list, AP_QUEUE_REMOVE_TEMPLATE([Search](#), AP_QUEUE_REMOVE_TEMPLATE(interpreter_variable_data, char *,(!strcmp([node](#)->name, key))))
- [AP_QUEUE_STD_DEFN_TEMPLATE](#) (AP_QUEUE_SEARCH_TEMPLATE(agent_queue, AP_QUEUE_SEARCH_TEMPLATE(agent)
- [int agent_queue_Flush](#) (agent_queue_p queue)
- [AP_QUEUE_STD_DEFN_TEMPLATE](#) (fipa_acl_message_p mail_queue_SearchReceivers(mail_queue_p mail_queue, fipa_acl_message)
- [AP_QUEUE_STD_DEFN_TEMPLATE](#) (mailbox_queue, agent_mailbox)
- [int AP_GENERIC_Destroy](#) (AP_GENERIC_t *blah)
- [AP_GENERIC_p AP_GENERIC_Copy](#) (AP_GENERIC_p AP_GENERIC)

13.11.1 Function Documentation

13.11.1.1 int agent_queue_Flush (agent_queue_p queue)

Definition at line 197 of file data_structures.c.

References agent_s::agent_status, agent_s::id, ListSearch(), MUTEX_LOCK, MUTEX_UNLOCK, agent_s::name, and node.

Referenced by handler_FLUSH_AGENTS().

13.11.1.2 AP_GENERIC_p AP_GENERIC_Copy (AP_GENERIC_p AP_GENERIC)

Definition at line 296 of file data_structures.c.

13.11.1.3 int AP_GENERIC_Destroy (AP_GENERIC_t * blah)

Definition at line 292 of file data_structures.c.

13.11.1.4 `AP_QUEUE_SEARCH_TEMPLATE (AP_QUEUE_REMOVE_TEMPLATE(
agent_variable_list, AP_QUEUE_REMOVE_TEMPLATE(Search,
AP_QUEUE_REMOVE_TEMPLATE(interpreter_variable_data, char *,
(!strcmp(node->name, key)))`

Definition at line 84 of file `data_structures.c`.

References `message_s::from_address`, `ListSearch()`, `message_s::message_id`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, and `message_s::to_address`.

13.11.1.5 `AP_QUEUE_SEARCH_TEMPLATE (AP_QUEUE_REMOVE_TEMPLATE(
connection_queue, Search, AP_QUEUE_REMOVE_TEMPLATE(connection,
AP_QUEUE_REMOVE_TEMPLATE(int, (node->connect_id==key))`

Definition at line 45 of file `data_structures.c`.

References `connection_s::connect_id`, `ListSearch()`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, and `connection_s::remote_hostname`.

13.11.1.6 `AP_QUEUE_STD_DEFN_TEMPLATE (mailbox_queue, agent_mailbox)`

Definition at line 253 of file `data_structures.c`.

References `agent_ChScriptInitVar()`, `COND_SIGNAL`, `ListPop()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

13.11.1.7 `AP_QUEUE_STD_DEFN_TEMPLATE (fipa_acl_message_p
mail_queue_SearchReceivers(mail_queue_p mail_queue, fipa_acl_message)`

Definition at line 214 of file `data_structures.c`.

References `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `fipa_agent_identifier_s::name`, `listNode_s::next`, `node`, `listNode_s::node_data`, and `fipa_acl_message_s::receiver`.

13.11.1.8 `AP_QUEUE_STD_DEFN_TEMPLATE (AP_QUEUE_SEARCH_TEMPLATE(
agent_queue, AP_QUEUE_SEARCH_TEMPLATE(agent)`

Definition at line 145 of file `data_structures.c`.

References `agent_s::agent_status`, `agent_s::id`, `ListSearch()`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `agent_s::name`, and `node`.

13.12 /home/dko/Projects/mobilec/trunk/src/df.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include "config.h"
#include "include/mc_platform.h"
#include "include/df.h"
#include "include/df_request.x.h"
```

Defines

- #define [REQUEST](#)(name, string, description)

Functions

- [int df_Add](#) (struct df_s *df, struct df_node_s *[node](#))
- [int df_AddRequest](#) (struct df_s *df, struct df_request_list_node_s *[node](#))
- [int df_Destroy](#) (df_p df)
- [df_p df_Initialize](#) (mc_platform_p mc_platform)
- [int df_ProcessRequest](#) (struct mc_platform_s *global)
- [int df_SearchForService](#) (df_p df, const char *searchstring, char ***agent_names, char ***service_names, [int](#) **agent_ids, [int](#) *num_entries)
- [void df_Start](#) (mc_platform_p mc_platform)
- [int df_request_list_node_Destroy](#) (df_request_list_node_p [node](#))
- [df_request_list_node_p df_request_list_node_New](#) (void)
- [int df_request_list_Destroy](#) (df_request_list_p df_request_list)
- [df_request_list_p df_request_list_New](#) (void)
- [df_request_list_node_p df_request_list_Pop](#) (df_request_list_p requests)
- [df_request_search_p df_request_search_New](#) (void)
- [int df_request_search_Destroy](#) (df_request_search_p [node](#))
- [int df_node_Destroy](#) (df_node_p df_node)
- [void * df_Thread](#) (void *arg)
- [int request_handler_REGISTER](#) (struct mc_platform_s *global, void *data)
- [int request_handler_SEARCH](#) (struct mc_platform_s *global, void *data)
- [int request_handler_SUBSCRIBE](#) (struct mc_platform_s *global, void *data)
- [int request_handler_DEREGISTER](#) (struct mc_platform_s *global, void *data)

13.12.1 Define Documentation

13.12.1.1 #define [REQUEST](#)(name, string, description)

Value:

```
if ( !strcmp(request->command, string ) ) { \
    return_code = MC_SUCCESS; \
    handler_code = request_handler_##name( \
        global, \
```

```

        request->data ); \
        request_code = REQUEST_##name; \
    } else

```

Referenced by `acc_connection_Thread()`, and `acc_MessageHandlerThread()`.

13.12.2 Function Documentation

13.12.2.1 `int df_Add (struct df_s * df, struct df_node_s * node)`

Definition at line 50 of file `df.c`.

References `ListAdd()`, `MC_SUCCESS`, and `SIGNAL`.

Referenced by `request_handler_REGISTER()`.

13.12.2.2 `int df_AddRequest (struct df_s * df, struct df_request_list_node_s * node)`

Definition at line 66 of file `df.c`.

References `ListAdd()`, and `SIGNAL`.

Referenced by `MC_DeregisterService()`, `MC_RegisterService()`, and `MC_SearchForService()`.

13.12.2.3 `int df_Destroy (df_p df)`

Definition at line 83 of file `df.c`.

References `COND_DESTROY`, `df_node_Destroy()`, `df_request_list_Destroy()`, `ListPop()`, `ListTerminate()`, `MC_SUCCESS`, `MUTEX_DESTROY`, and `MUTEX_LOCK`.

Referenced by `mc_platform_Destroy()`.

13.12.2.4 `df_p df_Initialize (mc_platform_p mc_platform)`

Definition at line 101 of file `df.c`.

References `COND_INIT`, `COND_T`, `df_request_list_New()`, `ListInitialize()`, `MUTEX_INIT`, and `MUTEX_T`.

Referenced by `mc_platform_Initialize()`.

13.12.2.5 `int df_node_Destroy (df_node_p df_node)`

Definition at line 412 of file `df.c`.

References `MC_SUCCESS`, and `MUTEX_LOCK`.

Referenced by `df_Destroy()`.

13.12.2.6 `int df_ProcessRequest (struct mc_platform_s * global)`

Definition at line 132 of file `df.c`.

References `mc_platform_s::df`, `df_request_list_Pop()`, `MC_ERR_EMPTY`, and `MC_ERR_INVALID`.

Referenced by df_Thread().

13.12.2.7 int df_request_list_Destroy (df_request_list_p df_request_list)

Definition at line 321 of file df.c.

References df_request_list_node_Destroy(), ListPop(), ListTerminate(), MC_SUCCESS, and node.

Referenced by df_Destroy().

13.12.2.8 df_request_list_p df_request_list_New (void)

Definition at line 343 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, ListInitialize(), MUTEX_INIT, and MUTEX_T.

Referenced by df_Initialize().

13.12.2.9 int df_request_list_node_Destroy (df_request_list_node_p node)

Definition at line 292 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by df_request_list_Destroy(), and MC_SearchForService().

13.12.2.10 df_request_list_node_p df_request_list_node_New (void)

Definition at line 303 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, MUTEX_T, and node.

Referenced by MC_DeregisterService(), MC_RegisterService(), and MC_SearchForService().

13.12.2.11 df_request_list_node_p df_request_list_Pop (df_request_list_p requests)

Definition at line 368 of file df.c.

References ListPop(), MUTEX_LOCK, MUTEX_UNLOCK, and node.

Referenced by df_ProcessRequest().

13.12.2.12 int df_request_search_Destroy (df_request_search_p node)

Definition at line 399 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by MC_SearchForService().

13.12.2.13 df_request_search_p df_request_search_New (void)

Definition at line 384 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, MUTEX_T, and search.

Referenced by MC_SearchForService().

13.12.2.14 `int df_SearchForService(df_p df, const char * searchstring, char *** agent_names, char *** service_names, int ** agent_ids, int * num_entries)`

Definition at line 176 of file df.c.

References MC_ERR_NOT_FOUND, MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, listNode_s::next, and listNode_s::node_data.

Referenced by request_handler_SEARCH().

13.12.2.15 `void df_Start(mc_platform_p mc_platform)`

Definition at line 261 of file df.c.

References mc_platform_s::df, df_Thread(), MC_THREAD_DF, mc_platform_s::stack_size, and THREAD_CREATE.

Referenced by mc_platform_Initialize().

13.12.2.16 `void* df_Thread(void * arg)`

Definition at line 426 of file df.c.

References COND_BROADCAST, COND_WAIT, mc_platform_s::df, df_ProcessRequest(), MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::quit, mc_platform_s::quit_lock, and THREAD_EXIT.

Referenced by df_Start().

13.12.2.17 `int request_handler_DEREGISTER(struct mc_platform_s * global, void * data)`

Definition at line 520 of file df.c.

References mc_platform_s::df, MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, listNode_s::next, node, and listNode_s::node_data.

13.12.2.18 `int request_handler_REGISTER(struct mc_platform_s * global, void * data)`

Definition at line 490 of file df.c.

References mc_platform_s::df, and df_Add().

13.12.2.19 `int request_handler_SEARCH(struct mc_platform_s * global, void * data)`

Definition at line 496 of file df.c.

References mc_platform_s::df, df_SearchForService(), MC_SUCCESS, search, and SIGNAL.

13.12.2.20 `int request_handler_SUBSCRIBE(struct mc_platform_s * global, void * data)`

Definition at line 515 of file df.c.

13.13 /home/dko/Projects/mobilec/trunk/src/dynstring.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "include/dynstring.h"
```

Functions

- [dynstring_t * dynstring_New](#) (void)
- [int dynstring_Append](#) (dynstring_t *msg, char *str)
- [int dynstring_Destroy](#) (dynstring_t *dynstring)

13.13.1 Function Documentation

13.13.1.1 int dynstring_Append (dynstring_t * msg, char * str)

Definition at line 30 of file dynstring.c.

References COMPOSE_BLOCKSIZE, dynstring_s::len, dynstring_s::message, and dynstring_s::size.

Referenced by fipa_acl_Compose(), fipa_agent_identifier_Compose(), fipa_agent_identifier_set_Compose(), fipa_DateTime_Compose(), fipa_envelope_Compose__from(), fipa_number_Compose(), fipa_performative_Compose(), fipa_protocol_Compose(), fipa_string_Compose(), fipa_url_Compose(), fipa_url_sequence_Compose(), fipa_word_Compose(), message_send_Thread(), mtp_http_CreateMessage(), and mtp_http_InitializeFromConnection().

13.13.1.2 int dynstring_Destroy (dynstring_t * dynstring)

Definition at line 56 of file dynstring.c.

References dynstring_s::message.

Referenced by fipa_envelope_Compose__from(), MC_AclSend(), message_send_Thread(), mtp_http_CreateMessage(), and mtp_http_InitializeFromConnection().

13.13.1.3 dynstring_t* dynstring_New (void)

Definition at line 14 of file dynstring.c.

References COMPOSE_BLOCKSIZE, dynstring_s::len, dynstring_s::message, and dynstring_s::size.

Referenced by fipa_acl_Compose(), fipa_envelope_Compose__from(), message_send_Thread(), mtp_http_CreateMessage(), and mtp_http_InitializeFromConnection().

13.14 /home/dko/Projects/mobilec/trunk/src/fipa_acl.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <strings.h>
#include "include/fipa_acl.h"
#include "include/mc_error.h"
#include "include/macros.h"
```

Defines

- #define [FREEMEM](#)(x) if (x != NULL) free(x)

Functions

- [fipa_acl_message_t * fipa_acl_message_New](#) (void)
- [int fipa_acl_message_Destroy](#) (fipa_acl_message_t *message)
- [fipa_acl_message_t * fipa_acl_message_Copy](#) (fipa_acl_message_t *src)
- [fipa_message_string_t * fipa_message_string_New](#) (void)
- [int fipa_message_string_Destroy](#) (fipa_message_string_t *message)
- [fipa_message_string_t * fipa_message_string_Copy](#) (fipa_message_string_t *src)
- [fipa_url_sequence_t * fipa_url_sequence_New](#) (void)
- [int fipa_url_sequence_Destroy](#) (fipa_url_sequence_t *sequence)
- [fipa_url_sequence_t * fipa_url_sequence_Copy](#) (fipa_url_sequence_t *src)
- [fipa_agent_identifier_set_t * fipa_agent_identifier_set_New](#) (void)
- [int fipa_agent_identifier_set_Destroy](#) (fipa_agent_identifier_set_t *idset)
- [fipa_agent_identifier_set_t * fipa_agent_identifier_set_Copy](#) (fipa_agent_identifier_set_t *src)
- [fipa_agent_identifier_t * fipa_agent_identifier_New](#) (void)
- [int fipa_agent_identifier_Destroy](#) (fipa_agent_identifier_t *id)
- [fipa_agent_identifier_t * fipa_agent_identifier_Copy](#) (fipa_agent_identifier_t *src)
- [fipa_expression_t * fipa_expression_New](#) (void)
- [int fipa_expression_Destroy](#) (fipa_expression_t *expr)
- [fipa_expression_t * fipa_expression_Copy](#) (fipa_expression_t *src)
- [fipa_word_t * fipa_word_New](#) (void)
- [int fipa_word_Destroy](#) (fipa_word_t *word)
- [fipa_word_t * fipa_word_Copy](#) (fipa_word_t *src)
- [fipa_string_t * fipa_string_New](#) (void)
- [int fipa_string_Destroy](#) (fipa_string_t *str)
- [fipa_string_t * fipa_string_Copy](#) (fipa_string_t *src)
- [fipa_DateTime_t * fipa_DateTime_New](#) (void)
- [int fipa_DateTime_Destroy](#) (fipa_DateTime_t *dt)
- [fipa_DateTime_t * fipa_DateTime_Copy](#) (fipa_DateTime_t *src)
- [fipa_url_t * fipa_url_New](#) (void)

- [int fipa_url_Destroy](#) ([fipa_url_t](#) *url)
- [fipa_url_t](#) * [fipa_url_Copy](#) ([fipa_url_t](#) *src)
- [fipa_number_t](#) * [fipa_number_New](#) (void)
- [int fipa_number_Destroy](#) ([fipa_number_t](#) *number)
- [fipa_number_t](#) * [fipa_number_Copy](#) ([fipa_number_t](#) *src)
- [int fipa_acl_Parse](#) ([fipa_acl_message_p](#) acl, [fipa_message_string_p](#) message)
- [int fipa_message_parameter_Parse](#) ([fipa_acl_message_p](#) acl, [fipa_message_string_p](#) message)
- [int fipa_protocol_type_Parse](#) (enum [fipa_protocol_e](#) *protocol, [fipa_message_string_p](#) message)
- [int fipa_message_type_Parse](#) (enum [fipa_performative_e](#) *performative, [fipa_message_string_p](#) message)
- [int fipa_GetAtom](#) ([fipa_message_string_p](#) message, char expected_atom)
- [int fipa_word_Parse](#) ([fipa_word_t](#) **word, [fipa_message_string_p](#) message)
- [int fipa_CheckNextToken](#) (const [fipa_message_string_p](#) message, const char *token)
- [int fipa_expression_Parse](#) ([fipa_expression_t](#) **expression, [fipa_message_string_p](#) message)
- [int fipa_GetNextWord](#) (char **word, const [fipa_message_string_p](#) message)
- [int fipa_GetWholeToken](#) (char **word, [fipa_message_string_p](#) message)
- [int fipa_datetime_Parse](#) ([fipa_DateTime_p](#) *datetime, [fipa_message_string_p](#) message)
- [int fipa_string_Parse](#) ([fipa_string_p](#) *fipa_string, [fipa_message_string_p](#) message)
- [int fipa_agent_identifier_Parse](#) ([fipa_agent_identifier_p](#) *aid, [fipa_message_string_p](#) message)
- [int fipa_url_sequence_Parse](#) ([fipa_url_sequence_p](#) *urls, [fipa_message_string_p](#) message)
- [int fipa_url_Parse](#) ([fipa_url_p](#) *url, [fipa_message_string_p](#) message)
- [int fipa_agent_identifier_set_Parse](#) ([fipa_agent_identifier_set_p](#) *agent_ids, [fipa_message_string_p](#) message)
- [int fipa_acl_Compose](#) ([dynstring_t](#) **msg, [fipa_acl_message_t](#) *acl)
- [int fipa_protocol_Compose](#) ([dynstring_t](#) *msg, enum [fipa_protocol_e](#) protocol)
- [int fipa_performative_Compose](#) ([dynstring_t](#) *msg, enum [fipa_performative_e](#) performative)
- [int fipa_url_sequence_Compose](#) ([dynstring_t](#) *msg, [fipa_url_sequence_t](#) *urls)
- [int fipa_agent_identifier_set_Compose](#) ([dynstring_t](#) *msg, [fipa_agent_identifier_set_t](#) *ids)
- [int fipa_agent_identifier_Compose](#) ([dynstring_t](#) *msg, [fipa_agent_identifier_t](#) *id)
- [int fipa_expression_Compose](#) ([dynstring_t](#) *msg, [fipa_expression_t](#) *expr)
- [int fipa_word_Compose](#) ([dynstring_t](#) *msg, [fipa_word_t](#) *word)
- [int fipa_string_Compose](#) ([dynstring_t](#) *msg, [fipa_string_t](#) *string)
- [int fipa_DateTime_Compose](#) ([dynstring_t](#) *msg, [fipa_DateTime_t](#) *date)
- [int fipa_url_Compose](#) ([dynstring_t](#) *msg, [fipa_url_t](#) *url)
- [int fipa_number_Compose](#) ([dynstring_t](#) *msg, [fipa_number_t](#) *number)
- [struct fipa_acl_message_s](#) * [fipa_Reply](#) ([struct fipa_acl_message_s](#) *acl)

13.14.1 Define Documentation

13.14.1.1 #define FREEMEM(x) if (x != NULL) free(x)

Definition at line 53 of file [fipa_acl.c](#).

Referenced by [fipa_expression_Destroy\(\)](#).

13.14.2 Function Documentation

13.14.2.1 `int fipa_acl_Compose (dynstring_t **msg, fipa_acl_message_t *acl)`

Definition at line 1232 of file fipa_acl.c.

References `fipa_acl_message_s::content`, `fipa_acl_message_s::conversation_id`, `dynstring_Append()`, `dynstring_New()`, `fipa_acl_message_s::encoding`, `fipa_agent_identifier_Compose()`, `fipa_agent_identifier_set_Compose()`, `fipa_DateTime_Compose()`, `fipa_expression_Compose()`, `fipa_performative_Compose()`, `fipa_protocol_Compose()`, `FIPA_PROTOCOL_NONE`, `fipa_string_Compose()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `fipa_acl_message_s::ontology`, `fipa_acl_message_s::performative`, `fipa_acl_message_s::protocol`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_by`, `fipa_acl_message_s::reply_to`, `fipa_acl_message_s::reply_with`, and `fipa_acl_message_s::sender`.

Referenced by `MC_AclSend()`.

13.14.2.2 `fipa_acl_message_t* fipa_acl_message_Copy (fipa_acl_message_t *src)`

Definition at line 86 of file fipa_acl.c.

References `fipa_acl_message_s::content`, `fipa_acl_message_s::conversation_id`, `fipa_acl_message_s::encoding`, `fipa_acl_message_New()`, `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_Copy()`, `fipa_DateTime_Copy()`, `fipa_expression_Copy()`, `fipa_string_Copy()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `fipa_acl_message_s::ontology`, `fipa_acl_message_s::performative`, `fipa_acl_message_s::protocol`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_by`, `fipa_acl_message_s::reply_to`, `fipa_acl_message_s::reply_with`, and `fipa_acl_message_s::sender`.

Referenced by `MC_AclSend()`.

13.14.2.3 `int fipa_acl_message_Destroy (fipa_acl_message_t *message)`

Definition at line 65 of file fipa_acl.c.

References `fipa_acl_message_s::content`, `fipa_acl_message_s::conversation_id`, `fipa_acl_message_s::encoding`, `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_Destroy()`, `fipa_DateTime_Destroy()`, `fipa_expression_Destroy()`, `fipa_string_Destroy()`, `fipa_acl_message_s::in_reply_to`, `fipa_acl_message_s::language`, `fipa_acl_message_s::ontology`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::reply_by`, `fipa_acl_message_s::reply_to`, `fipa_acl_message_s::reply_with`, and `fipa_acl_message_s::sender`.

Referenced by `acc_connection_Thread()`, and `MC_AclDestroy()`.

13.14.2.4 `fipa_acl_message_t* fipa_acl_message_New (void)`

Definition at line 57 of file fipa_acl.c.

Referenced by `acc_connection_Thread()`, `fipa_acl_message_Copy()`, `fipa_Reply()`, and `MC_AclNew()`.

13.14.2.5 `int fipa_acl_Parse (fipa_acl_message_p acl, fipa_message_string_p message)`

Definition at line 464 of file fipa_acl.c.

References `fipa_GetAtom()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `MC_ERR_PARSE`, and `fipa_acl_message_s::performative`.

Referenced by `acc_connection_Thread()`.

13.14.2.6 `int fipa_agent_identifier_Compose (dynstring_t * msg, fipa_agent_identifier_t * id)`

Definition at line 1446 of file `fipa_acl.c`.

References `fipa_agent_identifier_s::addresses`, `dynstring_Append()`, `fipa_agent_identifier_set_Compose()`, `fipa_url_sequence_Compose()`, `fipa_agent_identifier_s::name`, `fipa_agent_identifier_set_s::num`, `fipa_url_sequence_s::num`, and `fipa_agent_identifier_s::resolvers`.

Referenced by `fipa_acl_Compose()`, and `fipa_agent_identifier_set_Compose()`.

13.14.2.7 `fipa_agent_identifier_t* fipa_agent_identifier_Copy (fipa_agent_identifier_t * src)`

Definition at line 235 of file `fipa_acl.c`.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_New()`, `fipa_agent_identifier_set_Copy()`, `fipa_url_sequence_Copy()`, `fipa_agent_identifier_s::name`, and `fipa_agent_identifier_s::resolvers`.

Referenced by `fipa_acl_message_Copy()`, `fipa_acl_Param_Copy()`, `fipa_agent_identifier_set_Copy()`, and `fipa_Reply()`.

13.14.2.8 `int fipa_agent_identifier_Destroy (fipa_agent_identifier_t * id)`

Definition at line 223 of file `fipa_acl.c`.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_set_Destroy()`, `fipa_url_sequence_Destroy()`, `fipa_agent_identifier_s::name`, and `fipa_agent_identifier_s::resolvers`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, `fipa_agent_identifier_set_Destroy()`, and `MC_AclSetSender()`.

13.14.2.9 `fipa_agent_identifier_t* fipa_agent_identifier_New (void)`

Definition at line 215 of file `fipa_acl.c`.

Referenced by `fipa_agent_identifier_Copy()`, `fipa_envelope_ParseAgentIdentifier()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

13.14.2.10 `int fipa_agent_identifier_Parse (fipa_agent_identifier_p * aid, fipa_message_string_p message)`

Definition at line 1032 of file `fipa_acl.c`.

References `CHECK_NULL`, `fipa_word_s::content`, `fipa_agent_identifier_set_Parse()`, `fipa_GetAtom()`, `fipa_url_sequence_Parse()`, `fipa_word_Destroy()`, `fipa_word_Parse()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_set_Parse()`, and `fipa_message_parameter_Parse()`.

13.14.2.11 `int fipa_agent_identifier_set_Compose (dynstring_t * msg, fipa_agent_identifier_set_t * ids)`

Definition at line 1433 of file `fipa_acl.c`.

References `dynstring_Append()`, `fipa_agent_identifier_Compose()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, and `fipa_agent_identifier_set_s::num`.

Referenced by `fipa_acl_Compose()`, and `fipa_agent_identifier_Compose()`.

13.14.2.12 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_Copy (fipa_agent_identifier_set_t * src)`

Definition at line 194 of file `fipa_acl.c`.

References `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_agent_identifier_set_s::num`, and `fipa_agent_identifier_set_s::retain_order`.

Referenced by `fipa_acl_message_Copy()`, `fipa_acl_Param_Copy()`, `fipa_agent_identifier_Copy()`, and `fipa_Reply()`.

13.14.2.13 `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t * idset)`

Definition at line 182 of file `fipa_acl.c`.

References `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, and `fipa_agent_identifier_set_s::num`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, and `fipa_agent_identifier_Destroy()`.

13.14.2.14 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_New (void)`

Definition at line 174 of file `fipa_acl.c`.

Referenced by `fipa_agent_identifier_set_Copy()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, `fipa_Reply()`, `MC_AclAddReceiver()`, and `MC_AclAddReplyTo()`.

13.14.2.15 `int fipa_agent_identifier_set_Parse (fipa_agent_identifier_set_p * agent_ids, fipa_message_string_p message)`

Definition at line 1191 of file `fipa_acl.c`.

References `fipa_word_s::content`, `fipa_agent_identifier_Parse()`, `fipa_GetAtom()`, `fipa_word_Parse()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_agent_identifier_set_s::retain_order`.

Referenced by `fipa_agent_identifier_Parse()`, and `fipa_message_parameter_Parse()`.

13.14.2.16 `int fipa_CheckNextToken (const fipa_message_string_p message, const char * token)`

Definition at line 694 of file `fipa_acl.c`.

References `fipa_message_string_s::parse`.

Referenced by `fipa_expression_Parse()`.

13.14.2.17 `int fipa_DateTime_Compose (dynstring_t * msg, fipa_DateTime_t * date)`

Definition at line 1519 of file `fipa_acl.c`.

References `buf`, `fipa_DateTime_s::day`, `dynstring_Append()`, `fipa_DateTime_s::hour`, `fipa_DateTime_s::millisecond`, `fipa_DateTime_s::minute`, `fipa_DateTime_s::month`, `fipa_DateTime_s::second`, `fipa_DateTime_s::sign`, and `fipa_DateTime_s::year`.

Referenced by `fipa_acl_Compose()`, and `fipa_expression_Compose()`.

13.14.2.18 `fipa_DateTime_t* fipa_DateTime_Copy (fipa_DateTime_t * src)`

Definition at line 398 of file `fipa_acl.c`.

References `fipa_DateTime_New()`.

Referenced by `fipa_acl_envelope_Received_Copy()`, `fipa_acl_message_Copy()`, `fipa_acl_Param_Copy()`, and `fipa_expression_Copy()`.

13.14.2.19 `int fipa_DateTime_Destroy (fipa_DateTime_t * dt)`

Definition at line 391 of file `fipa_acl.c`.

Referenced by `fipa_acl_envelope_Received_Destroy()`, `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, and `fipa_expression_Destroy()`.

13.14.2.20 `fipa_DateTime_t* fipa_DateTime_New (void)`

Definition at line 383 of file `fipa_acl.c`.

Referenced by `fipa_DateTime_Copy()`.

13.14.2.21 `int fipa_datetime_Parse (fipa_DateTime_p * datetime, fipa_message_string_p message)`

Definition at line 849 of file `fipa_acl.c`.

References `buf`, `fipa_GetWholeToken()`, `MC_ERR_PARSE`, `MC_SUCCESS`, `fipa_message_string_s::parse`, and `fipa_DateTime_s::sign`.

Referenced by `fipa_envelope_HandleDate()`, `fipa_envelope_HandleReceived()`, `fipa_expression_Parse()`, and `fipa_message_parameter_Parse()`.

13.14.2.22 `int fipa_expression_Compose (dynstring_t * msg, fipa_expression_t * expr)`

Definition at line 1472 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Compose()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Compose()`, `fipa_number_Compose()`, `fipa_string_Compose()`, `fipa_word_Compose()`, `MC_ERR_PARSE`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_Compose()`, and `fipa_expression_Compose()`.

13.14.2.23 `fipa_expression_t* fipa_expression_Copy (fipa_expression_t * src)`

Definition at line 286 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Copy()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Copy()`, `fipa_expression_Destroy()`, `fipa_expression_New()`, `fipa_number_Copy()`, `fipa_string_Copy()`, `fipa_word_Copy()`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_message_Copy()`, `fipa_expression_Copy()`, and `fipa_Reply()`.

13.14.2.24 `int fipa_expression_Destroy (fipa_expression_t * expr)`

Definition at line 255 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Destroy()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Destroy()`, `fipa_number_Destroy()`, `fipa_string_Destroy()`, `fipa_word_Destroy()`, `FREEMEM`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_expression_Copy()`, and `fipa_expression_Destroy()`.

13.14.2.25 `fipa_expression_t* fipa_expression_New (void)`

Definition at line 247 of file `fipa_acl.c`.

Referenced by `fipa_expression_Copy()`, and `MC_AclSetConversationID()`.

13.14.2.26 `int fipa_expression_Parse (fipa_expression_t ** expression, fipa_message_string_p message)`

Definition at line 713 of file `fipa_acl.c`.

References `fipa_CheckNextToken()`, `fipa_datetime_Parse()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Parse()`, `fipa_GetAtom()`, `fipa_string_Parse()`, `fipa_word_Parse()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_expression_s::type`.

Referenced by `fipa_expression_Parse()`, and `fipa_message_parameter_Parse()`.

13.14.2.27 `int fipa_GetAtom (fipa_message_string_p message, char expected_atom)`

Definition at line 629 of file `fipa_acl.c`.

References `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_acl_Parse()`, `fipa_agent_identifier_Parse()`, `fipa_agent_identifier_set_Parse()`, `fipa_expression_Parse()`, `fipa_message_parameter_Parse()`, `fipa_string_Parse()`, and `fipa_url_sequence_Parse()`.

13.14.2.28 `int fipa_GetNextWord (char ** word, const fipa_message_string_p message)`

Definition at line 764 of file `fipa_acl.c`.

References `ERR`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

13.14.2.29 int fipa_GetWholeToken (char ** *word*, fipa_message_string_p *message*)

Definition at line 812 of file fipa_acl.c.

References MC_SUCCESS, and fipa_message_string_s::parse.

Referenced by fipa_datetime_Parse().

13.14.2.30 int fipa_message_parameter_Parse (fipa_acl_message_p *acl*, fipa_message_string_p *message*)

Definition at line 484 of file fipa_acl.c.

References fipa_acl_message_s::content, fipa_word_s::content, fipa_acl_message_s::conversation_id, fipa_acl_message_s::encoding, fipa_agent_identifier_Parse(), fipa_agent_identifier_set_Parse(), fipa_datetime_Parse(), fipa_expression_Parse(), fipa_GetAtom(), fipa_protocol_type_Parse(), fipa_string_Parse(), fipa_word_Destroy(), fipa_word_Parse(), fipa_acl_message_s::in_reply_to, fipa_acl_message_s::language, MC_ERR_PARSE, fipa_acl_message_s::ontology, fipa_acl_message_s::protocol, fipa_acl_message_s::receiver, fipa_acl_message_s::reply_by, fipa_acl_message_s::reply_to, fipa_acl_message_s::reply_with, and fipa_acl_message_s::sender.

Referenced by fipa_acl_Parse().

13.14.2.31 fipa_message_string_t* fipa_message_string_Copy (fipa_message_string_t * *src*)

Definition at line 128 of file fipa_acl.c.

References fipa_message_string_s::message, and fipa_message_string_s::parse.

13.14.2.32 int fipa_message_string_Destroy (fipa_message_string_t * *message*)

Definition at line 118 of file fipa_acl.c.

References fipa_message_string_s::message.

Referenced by acc_connection_Thread().

13.14.2.33 fipa_message_string_t* fipa_message_string_New (void)

Definition at line 110 of file fipa_acl.c.

Referenced by acc_connection_Thread().

13.14.2.34 int fipa_message_type_Parse (enum fipa_performative_e * *performative*, fipa_message_string_p *message*)

Definition at line 567 of file fipa_acl.c.

References fipa_word_s::content, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE, FIPA_CALL_FOR_PROPOSAL, FIPA_CANCEL, FIPA_CONFIRM, FIPA_DISCONFIRM, FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF, FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY, FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL, FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_SUBSCRIBE, fipa_word_Destroy(), fipa_word_Parse(), and MC_ERR_PARSE.

Referenced by `fipa_acl_Parse()`.

13.14.2.35 `int fipa_number_Compose (dynstring_t * msg, fipa_number_t * number)`

Definition at line 1547 of file `fipa_acl.c`.

References `dynstring_Append()`, and `fipa_number_s::str`.

Referenced by `fipa_expression_Compose()`.

13.14.2.36 `fipa_number_t* fipa_number_Copy (fipa_number_t * src)`

Definition at line 454 of file `fipa_acl.c`.

References `fipa_number_New()`, and `fipa_number_s::str`.

Referenced by `fipa_expression_Copy()`.

13.14.2.37 `int fipa_number_Destroy (fipa_number_t * number)`

Definition at line 444 of file `fipa_acl.c`.

References `fipa_number_s::str`.

Referenced by `fipa_expression_Destroy()`.

13.14.2.38 `fipa_number_t* fipa_number_New (void)`

Definition at line 436 of file `fipa_acl.c`.

Referenced by `fipa_number_Copy()`.

13.14.2.39 `int fipa_performative_Compose (dynstring_t * msg, enum fipa_performative_e performative)`

Definition at line 1345 of file `fipa_acl.c`.

References `dynstring_Append()`, `FIPA_ACCEPT_PROPOSAL`, `FIPA_AGREE`, `FIPA_CALL_FOR_PROPOSAL`, `FIPA_CANCEL`, `FIPA_CONFIRM`, `FIPA_DISCONFIRM`, `FIPA_FAILURE`, `FIPA_INFORM`, `FIPA_INFORM_IF`, `FIPA_INFORM_REF`, `FIPA_NOT_UNDERSTOOD`, `FIPA_PROPOGATE`, `FIPA_PROPOSE`, `FIPA_PROXY`, `FIPA_QUERY_IF`, `FIPA_QUERY_REF`, `FIPA_REFUSE`, `FIPA_REJECT_PROPOSAL`, `FIPA_REQUEST`, `FIPA_REQUEST_WHEN`, `FIPA_REQUEST_WHENEVER`, `FIPA_SUBSCRIBE`, and `MC_ERR_PARSE`.

Referenced by `fipa_acl_Compose()`.

13.14.2.40 `int fipa_protocol_Compose (dynstring_t * msg, enum fipa_protocol_e protocol)`

Definition at line 1302 of file `fipa_acl.c`.

References `dynstring_Append()`, `FIPA_PROTOCOL_BROKERING`, `FIPA_PROTOCOL_CONTRACT_NET`, `FIPA_PROTOCOL_DUTCH_AUCTION`, `FIPA_PROTOCOL_ENGLISH_AUCTION`, `FIPA_PROTOCOL_ITERATED_CONTRACT_NET`, `FIPA_PROTOCOL_PROPOSE`, `FIPA_PROTOCOL_QUERY`, `FIPA_PROTOCOL_RECRUITING`, `FIPA_PROTOCOL_REQUEST`, `FIPA_PROTOCOL_REQUEST_WHEN`, `FIPA_PROTOCOL_SUBSCRIBE`, and `MC_ERR_PARSE`.

Referenced by fipa_acl_Compose().

13.14.2.41 **int fipa_protocol_type_Parse** (enum fipa_protocol_e * *protocol*, fipa_message_string_p *message*)

Definition at line 527 of file fipa_acl.c.

References fipa_word_s::content, FIPA_PROTOCOL_BROKERING, FIPA_PROTOCOL_CONTRACT_NET, FIPA_PROTOCOL_DUTCH_AUCTION, FIPA_PROTOCOL_ENGLISH_AUCTION, FIPA_PROTOCOL_ITERATED_CONTRACT_NET, FIPA_PROTOCOL_PROPOSE, FIPA_PROTOCOL_QUERY, FIPA_PROTOCOL_RECRUITING, FIPA_PROTOCOL_REQUEST, FIPA_PROTOCOL_REQUEST_WHEN, FIPA_PROTOCOL_SUBSCRIBE, fipa_word_Destroy(), fipa_word_Parse(), and MC_ERR_PARSE.

Referenced by fipa_message_parameter_Parse().

13.14.2.42 **struct fipa_acl_message_s* fipa_Reply** (struct fipa_acl_message_s * *acl*) **[read]**

Definition at line 1555 of file fipa_acl.c.

References fipa_acl_message_s::conversation_id, fipa_acl_message_New(), fipa_agent_identifier_Copy(), fipa_agent_identifier_set_Copy(), fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_expression_Copy(), fipa_agent_identifier_set_s::num, fipa_acl_message_s::protocol, fipa_acl_message_s::receiver, fipa_acl_message_s::reply_to, fipa_agent_identifier_set_s::retain_order, and fipa_acl_message_s::sender.

Referenced by MC_AclReply().

13.14.2.43 **int fipa_string_Compose** (dynstring_t * *msg*, fipa_string_t * *string*)

Definition at line 1510 of file fipa_acl.c.

References fipa_string_s::content, and dynstring_Append().

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

13.14.2.44 **fipa_string_t* fipa_string_Copy** (fipa_string_t * *src*)

Definition at line 373 of file fipa_acl.c.

References fipa_string_s::content, and fipa_string_New().

Referenced by fipa_acl_message_Copy(), and fipa_expression_Copy().

13.14.2.45 **int fipa_string_Destroy** (fipa_string_t * *str*)

Definition at line 363 of file fipa_acl.c.

References fipa_string_s::content.

Referenced by fipa_acl_message_Destroy(), fipa_expression_Destroy(), and MC_AclSetContent().

13.14.2.46 **fipa_string_t* fipa_string_New** (void)

Definition at line 355 of file fipa_acl.c.

Referenced by `fipa_string_Copy()`, `MC_AclSetContent()`, and `MC_AclSetConversationID()`.

13.14.2.47 `int fipa_string_Parse (fipa_string_p * fipa_string, fipa_message_string_p message)`

Definition at line 987 of file `fipa_acl.c`.

References `fipa_string_s::content`, `fipa_GetAtom()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_expression_Parse()`, and `fipa_message_parameter_Parse()`.

13.14.2.48 `int fipa_url_Compose (dynstring_t * msg, fipa_url_t * url)`

Definition at line 1539 of file `fipa_acl.c`.

References `dynstring_Append()`, and `fipa_url_s::str`.

Referenced by `fipa_url_sequence_Compose()`.

13.14.2.49 `fipa_url_t* fipa_url_Copy (fipa_url_t * src)`

Definition at line 426 of file `fipa_acl.c`.

References `fipa_url_New()`, and `fipa_url_s::str`.

Referenced by `fipa_acl_envelope_Received_Copy()`, and `fipa_url_sequence_Copy()`.

13.14.2.50 `int fipa_url_Destroy (fipa_url_t * url)`

Definition at line 416 of file `fipa_acl.c`.

References `fipa_url_s::str`.

Referenced by `fipa_acl_envelope_Received_Destroy()`, and `fipa_url_sequence_Destroy()`.

13.14.2.51 `fipa_url_t* fipa_url_New (void)`

Definition at line 408 of file `fipa_acl.c`.

Referenced by `fipa_envelope_HandleReceived()`, `fipa_envelope_ParseAddresses()`, `fipa_url_Copy()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

13.14.2.52 `int fipa_url_Parse (fipa_url_p * url, fipa_message_string_p message)`

Definition at line 1171 of file `fipa_acl.c`.

References `fipa_word_s::content`, `fipa_word_Destroy()`, and `fipa_word_Parse()`.

Referenced by `fipa_url_sequence_Parse()`.

13.14.2.53 `int fipa_url_sequence_Compose (dynstring_t * msg, fipa_url_sequence_t * urls)`

Definition at line 1420 of file `fipa_acl.c`.

References `dynstring_Append()`, `fipa_url_Compose()`, `fipa_url_sequence_s::num`, and `fipa_url_sequence_s::urls`.

Referenced by `fipa_agent_identifier_Compose()`.

13.14.2.54 `fipa_url_sequence_t* fipa_url_sequence_Copy (fipa_url_sequence_t * src)`

Definition at line 158 of file `fipa_acl.c`.

References `fipa_url_Copy()`, `fipa_url_sequence_New()`, `fipa_url_sequence_s::num`, and `fipa_url_sequence_s::urls`.

Referenced by `fipa_agent_identifier_Copy()`.

13.14.2.55 `int fipa_url_sequence_Destroy (fipa_url_sequence_t * sequence)`

Definition at line 146 of file `fipa_acl.c`.

References `fipa_url_Destroy()`, `fipa_url_sequence_s::num`, and `fipa_url_sequence_s::urls`.

Referenced by `fipa_agent_identifier_Destroy()`.

13.14.2.56 `fipa_url_sequence_t* fipa_url_sequence_New (void)`

Definition at line 138 of file `fipa_acl.c`.

Referenced by `fipa_url_sequence_Copy()`, `fipa_url_sequence_Parse()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

13.14.2.57 `int fipa_url_sequence_Parse (fipa_url_sequence_p * urls, fipa_message_string_p message)`

Definition at line 1139 of file `fipa_acl.c`.

References `fipa_word_s::content`, `fipa_GetAtom()`, `fipa_url_Parse()`, `fipa_url_sequence_New()`, `fipa_word_Destroy()`, `fipa_word_Parse()`, and `MC_ERR_PARSE`.

Referenced by `fipa_agent_identifier_Parse()`.

13.14.2.58 `int fipa_word_Compose (dynstring_t * msg, fipa_word_t * word)`

Definition at line 1502 of file `fipa_acl.c`.

References `fipa_word_s::content`, and `dynstring_Append()`.

Referenced by `fipa_expression_Compose()`.

13.14.2.59 `fipa_word_t* fipa_word_Copy (fipa_word_t * src)`

Definition at line 345 of file `fipa_acl.c`.

References `fipa_word_s::content`, and `fipa_word_New()`.

Referenced by `fipa_expression_Copy()`.

13.14.2.60 int fipa_word_Destroy (fipa_word_t * *word*)

Definition at line 335 of file fipa_acl.c.

References fipa_word_s::content.

Referenced by fipa_agent_identifier_Parse(), fipa_expression_Destroy(), fipa_message_parameter_Parse(), fipa_message_type_Parse(), fipa_protocol_type_Parse(), fipa_url_Parse(), and fipa_url_sequence_Parse().

13.14.2.61 fipa_word_t* fipa_word_New (void)

Definition at line 327 of file fipa_acl.c.

Referenced by fipa_word_Copy().

13.14.2.62 int fipa_word_Parse (fipa_word_t ** *word*, fipa_message_string_p *message*)

Definition at line 652 of file fipa_acl.c.

References CHECK_NULL, MC_ERR_PARSE, MC_SUCCESS, and fipa_message_string_s::parse.

Referenced by fipa_agent_identifier_Parse(), fipa_agent_identifier_set_Parse(), fipa_expression_Parse(), fipa_message_parameter_Parse(), fipa_message_type_Parse(), fipa_protocol_type_Parse(), fipa_url_Parse(), and fipa_url_sequence_Parse().

13.15 /home/dko/Projects/mobilec/trunk/src/fipa_envelope.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <mxml.h>
#include <time.h>
#include "include/fipa_acl_envelope.h"
#include "include/mc_error.h"
#include "include/mc_platform.h"
```

Functions

- [fipa_acl_envelope_Received_t * fipa_acl_envelope_Received_New \(void\)](#)
- [int fipa_acl_envelope_Received_Destroy \(fipa_acl_envelope_Received_t *received\)](#)
- [fipa_acl_envelope_Received_t * fipa_acl_envelope_Received_Copy \(fipa_acl_envelope_Received_t *received\)](#)
- [fipa_acl_Param_t * fipa_acl_Param_New \(void\)](#)
- [int fipa_acl_Param_Destroy \(fipa_acl_Param_t *param\)](#)
- [fipa_acl_Param_t * fipa_acl_Param_Copy \(fipa_acl_Param_t *param\)](#)
- [fipa_acl_envelope_t * fipa_acl_envelope_New \(void\)](#)
- [int fipa_acl_envelope_Destroy \(fipa_acl_envelope_t *envelope\)](#)
- [fipa_acl_envelope_t * fipa_acl_envelope_Copy \(fipa_acl_envelope_t *envelope\)](#)
- [int fipa_envelope_Parse \(struct fipa_acl_envelope_s *envelope, const char *message\)](#)
- [int fipa_envelope_HandleEnvelope \(struct fipa_acl_envelope_s *envelope, mxml_node_t *node\)](#)
- [int fipa_envelope_HandleParams \(struct fipa_acl_envelope_s *envelope, mxml_node_t *node\)](#)
- [int fipa_envelope_HandleTo \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandleFrom \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandleComments \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandleAclRepresentation \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandlePayloadLength \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandlePayloadEncoding \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandleDate \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandleIntendedReceiver \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_HandleReceived \(struct fipa_acl_envelope_s *envelope, mxml_node_t *param_node, int cur_param\)](#)
- [int fipa_envelope_ParseAgentIdentifier \(struct fipa_agent_identifier_s **aid, mxml_node_t *agent_node\)](#)
- [int fipa_envelope_ParseAddresses \(struct fipa_agent_identifier_s *aid, mxml_node_t *addresses_node\)](#)

- `int fipa_envelope_ParseResolvers` (struct `fipa_agent_identifier_s` *aid, `mxml_node_t` *resolvers_node)
- `char * fipa_envelope_Compose` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__envelope` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__params` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__to` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__from` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__acl_representation` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__payload_encoding` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__date` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__intended_receiver` (`fipa_acl_message_t` *fipa_acl)

13.15.1 Function Documentation

13.15.1.1 `fipa_acl_envelope_t* fipa_acl_envelope_Copy` (`fipa_acl_envelope_t` * *envelope*)

Definition at line 150 of file `fipa_envelope.c`.

References `fipa_acl_envelope_New()`, `fipa_acl_Param_Copy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

13.15.1.2 `int fipa_acl_envelope_Destroy` (`fipa_acl_envelope_t` * *envelope*)

Definition at line 137 of file `fipa_envelope.c`.

References `fipa_acl_Param_Destroy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

Referenced by `acc_connection_Thread()`.

13.15.1.3 `fipa_acl_envelope_t* fipa_acl_envelope_New` (`void`)

Definition at line 129 of file `fipa_envelope.c`.

Referenced by `acc_connection_Thread()`, and `fipa_acl_envelope_Copy()`.

13.15.1.4 `fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_Copy` (`fipa_acl_envelope_Received_t` * *received*)

Definition at line 69 of file `fipa_envelope.c`.

References `fipa_acl_envelope_Received_New()`, `fipa_DateTime_Copy()`, `fipa_url_Copy()`, `fipa_acl_envelope_Received_s::received_by`, `fipa_acl_envelope_Received_s::received_date`, `fipa_acl_envelope_Received_s::received_from`, `fipa_acl_envelope_Received_s::received_id`, and `fipa_acl_envelope_Received_s::received_via`.

Referenced by `fipa_acl_Param_Copy()`.

13.15.1.5 `int fipa_acl_envelope_Received_Destroy` (`fipa_acl_envelope_Received_t` * *received*)

Definition at line 57 of file `fipa_envelope.c`.

References `fipa_DateTime_Destroy()`, `fipa_url_Destroy()`, `fipa_acl_envelope_Received_s::received_by`, `fipa_acl_envelope_Received_s::received_date`, `fipa_acl_envelope_Received_s::received_from`, `fipa_acl_envelope_Received_s::received_id`, and `fipa_acl_envelope_Received_s::received_via`.

Referenced by `fipa_acl_Param_Destroy()`.

13.15.1.6 `fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_New (void)`

Definition at line 48 of file `fipa_envelope.c`.

Referenced by `fipa_acl_envelope_Received_Copy()`, and `fipa_envelope_HandleReceived()`.

13.15.1.7 `fipa_acl_Param_t* fipa_acl_Param_Copy (fipa_acl_Param_t * param)`

Definition at line 110 of file `fipa_envelope.c`.

References `fipa_acl_Param_s::acl_representation`, `fipa_acl_Param_s::comments`, `fipa_acl_Param_s::date`, `fipa_acl_envelope_Received_Copy()`, `fipa_acl_Param_New()`, `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_Copy()`, `fipa_DateTime_Copy()`, `fipa_acl_Param_s::from`, `fipa_acl_Param_s::intended_receiver`, `fipa_acl_Param_s::payload_encoding`, `fipa_acl_Param_s::payload_length`, `fipa_acl_Param_s::received`, and `fipa_acl_Param_s::to`.

Referenced by `fipa_acl_envelope_Copy()`.

13.15.1.8 `int fipa_acl_Param_Destroy (fipa_acl_Param_t * param)`

Definition at line 93 of file `fipa_envelope.c`.

References `fipa_acl_Param_s::acl_representation`, `fipa_acl_Param_s::comments`, `fipa_acl_Param_s::date`, `fipa_acl_envelope_Received_Destroy()`, `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_Destroy()`, `fipa_DateTime_Destroy()`, `fipa_acl_Param_s::from`, `fipa_acl_Param_s::intended_receiver`, `fipa_acl_Param_s::payload_encoding`, `fipa_acl_Param_s::payload_length`, `fipa_acl_Param_s::received`, and `fipa_acl_Param_s::to`.

Referenced by `fipa_acl_envelope_Destroy()`.

13.15.1.9 `fipa_acl_Param_t* fipa_acl_Param_New (void)`

Definition at line 85 of file `fipa_envelope.c`.

Referenced by `fipa_acl_Param_Copy()`, and `fipa_envelope_HandleTo()`.

13.15.1.10 `char* fipa_envelope_Compose (fipa_acl_message_t * fipa_acl)`

Definition at line 869 of file `fipa_envelope.c`.

References `fipa_envelope_Compose__envelope()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmlAdd()`, `mxmlDelete()`, `mxmlLoadString()`, `mxmlSaveAllocString()`, and `node`.

Referenced by `MC_AclSend()`.

13.15.1.11 mxmml_node_t* fipa_envelope_Compose__acl_representation (fipa_acl_message_t * *fipa_acl*)

Definition at line 1089 of file fipa_envelope.c.

References mxmmlNewElement(), mxmmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

13.15.1.12 mxmml_node_t* fipa_envelope_Compose__date (fipa_acl_message_t * *fipa_acl*)

Definition at line 1117 of file fipa_envelope.c.

References buf, mxmmlNewElement(), mxmmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

13.15.1.13 mxmml_node_t* fipa_envelope_Compose__envelope (fipa_acl_message_t * *fipa_acl*)

Definition at line 894 of file fipa_envelope.c.

References fipa_envelope_Compose__params(), MXML_ADD_AFTER, MXML_ADD_TO_PARENT, mxmmlAdd(), mxmmlNewElement(), and node.

Referenced by fipa_envelope_Compose().

13.15.1.14 mxmml_node_t* fipa_envelope_Compose__from (fipa_acl_message_t * *fipa_acl*)

Definition at line 1021 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, buf, dynstring_Append(), dynstring_Destroy(), dynstring_New(), g_mc_platform, mc_platform_s::hostname, dynstring_s::message, mxmmlNewElement(), mxmmlNewText(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, mc_platform_s::port, fipa_acl_message_s::sender, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by fipa_envelope_Compose__params().

13.15.1.15 mxmml_node_t* fipa_envelope_Compose__intended_receiver (fipa_acl_message_t * *fipa_acl*)

Definition at line 1146 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_set_s::fipa_agent_identifiers, mxmmlNewElement(), mxmmlNewText(), fipa_agent_identifier_s::name, node, fipa_url_sequence_s::num, fipa_acl_message_s::receiver, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by fipa_envelope_Compose__params().

13.15.1.16 mxmml_node_t* fipa_envelope_Compose__params (fipa_acl_message_t * *fipa_acl*)

Definition at line 914 of file fipa_envelope.c.

References fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), fipa_envelope_Compose__intended_receiver(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), MXML_ADD_AFTER, MXML_ADD_TO_PARENT, mxmmlAdd(), mxmmlElementSetAttr(), mxmmlNewElement(), and node.

Referenced by fipa_envelope_Compose__envelope().

13.15.1.17 **mxml_node_t* fipa_envelope_Compose__payload_encoding (fipa_acl_message_t * *fipa_acl*)**

Definition at line 1103 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

13.15.1.18 **mxml_node_t* fipa_envelope_Compose__to (fipa_acl_message_t * *fipa_acl*)**

Definition at line 975 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_set_s::fipa_agent_identifiers, mxmlNewElement(), mxmlNewText(), fipa_agent_identifier_s::name, node, fipa_url_sequence_s::num, fipa_agent_identifier_set_s::num, fipa_acl_message_s::receiver, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by fipa_envelope_Compose__params().

13.15.1.19 **int fipa_envelope_HandleAclRepresentation (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)**

Definition at line 381 of file fipa_envelope.c.

References fipa_acl_Param_s::acl_representation, mxml_node_s::child, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.15.1.20 **int fipa_envelope_HandleComments (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)**

Definition at line 357 of file fipa_envelope.c.

References mxml_node_s::child, fipa_acl_Param_s::comments, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.15.1.21 **int fipa_envelope_HandleDate (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)**

Definition at line 452 of file fipa_envelope.c.

References mxml_node_s::child, fipa_acl_Param_s::date, fipa_datetime_Parse(), MC_ERR_PARSE, fipa_message_string_s::message, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, fipa_message_string_s::parse, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.15.1.22 **int fipa_envelope_HandleEnvelope** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *node*)

Definition at line 178 of file fipa_envelope.c.

References fipa_envelope_HandleParams(), MC_ERR_PARSE, MXML_DESCEND_FIRST, and mxmlFindElement().

Referenced by fipa_envelope_Parse().

13.15.1.23 **int fipa_envelope_HandleFrom** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 328 of file fipa_envelope.c.

References fipa_envelope_ParseAgentIdentifier(), fipa_acl_Param_s::from, MC_ERR_PARSE, MXML_DESCEND_FIRST, mxmlFindElement(), and fipa_acl_envelope_s::params.

Referenced by fipa_envelope_HandleParams().

13.15.1.24 **int fipa_envelope_HandleIntendedReceiver** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 482 of file fipa_envelope.c.

References fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_ParseAgentIdentifier(), fipa_acl_Param_s::intended_receiver, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlFindElement(), fipa_agent_identifier_set_s::num, fipa_acl_envelope_s::params, and fipa_agent_identifier_set_s::retain_order.

Referenced by fipa_envelope_HandleParams().

13.15.1.25 **int fipa_envelope_HandleParams** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *node*)

Definition at line 198 of file fipa_envelope.c.

References buf, fipa_envelope_HandleAclRepresentation(), fipa_envelope_HandleComments(), fipa_envelope_HandleDate(), fipa_envelope_HandleFrom(), fipa_envelope_HandleIntendedReceiver(), fipa_envelope_HandlePayloadEncoding(), fipa_envelope_HandlePayloadLength(), fipa_envelope_HandleReceived(), fipa_envelope_HandleTo(), MC_ERR_PARSE, MXML_DESCEND_FIRST, mxmlFindElement(), fipa_acl_envelope_s::num_params, and fipa_acl_envelope_s::params.

Referenced by fipa_envelope_HandleEnvelope().

13.15.1.26 **int fipa_envelope_HandlePayloadEncoding** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 428 of file fipa_envelope.c.

References mxml_node_s::child, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, fipa_acl_Param_s::payload_encoding, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.15.1.27 int fipa_envelope_HandlePayloadLength (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 405 of file fipa_envelope.c.

References mxml_node_s::child, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, fipa_acl_Param_s::payload_length, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.15.1.28 int fipa_envelope_HandleReceived (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 560 of file fipa_envelope.c.

References fipa_acl_envelope_Received_New(), fipa_datetime_Parse(), fipa_url_New(), MC_ERR_PARSE, fipa_message_string_s::message, MXML_DESCEND_FIRST, mxmlElementGetAttr(), mxmlFindElement(), node, fipa_acl_envelope_s::params, fipa_message_string_s::parse, fipa_acl_Param_s::received, fipa_acl_envelope_Received_s::received_by, fipa_acl_envelope_Received_s::received_date, fipa_acl_envelope_Received_s::received_from, fipa_acl_envelope_Received_s::received_id, fipa_acl_envelope_Received_s::received_via, and fipa_url_s::str.

Referenced by fipa_envelope_HandleParams().

13.15.1.29 int fipa_envelope_HandleTo (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 250 of file fipa_envelope.c.

References fipa_acl_Param_New(), fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_ParseAgentIdentifier(), MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlFindElement(), fipa_agent_identifier_set_s::num, fipa_acl_envelope_s::params, fipa_agent_identifier_set_s::retain_order, and fipa_acl_Param_s::to.

Referenced by fipa_envelope_HandleParams().

13.15.1.30 int fipa_envelope_Parse (struct fipa_acl_envelope_s * *envelope*, const char * *message*)

Definition at line 165 of file fipa_envelope.c.

References fipa_envelope_HandleEnvelope(), MXML_NO_CALLBACK, mxmlDelete(), and mxmlLoadString().

Referenced by acc_connection_Thread().

13.15.1.31 int fipa_envelope_ParseAddresses (struct fipa_agent_identifier_s * *aid*, mxml_node_t * *addresses_node*)

Definition at line 737 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, mxml_node_s::child, fipa_url_New(), MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_NO_DESCEND, MXML_TEXT, mxmlFindElement(), fipa_url_sequence_s::num, fipa_url_s::str, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, fipa_url_sequence_s::urls, and mxml_node_s::value.

Referenced by `fipa_envelope_ParseAgentIdentifier()`.

13.15.1.32 `int fipa_envelope_ParseAgentIdentifier (struct fipa_agent_identifier_s ** aid, mxml_node_t * agent_identifier_node)`

Definition at line 687 of file `fipa_envelope.c`.

References `mxml_node_s::child`, `fipa_agent_identifier_New()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseResolvers()`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `MXML_TEXT`, `mxmlFindElement()`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, and `fipa_envelope_ParseResolvers()`.

13.15.1.33 `int fipa_envelope_ParseResolvers (struct fipa_agent_identifier_s * aid, mxml_node_t * resolvers_node)`

Definition at line 804 of file `fipa_envelope.c`.

References `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_envelope_ParseAgentIdentifier()`, `MC_ERR_PARSE`, `MXML_DESCEND_FIRST`, `MXML_NO_DESCEND`, `mxmlFindElement()`, `fipa_agent_identifier_set_s::num`, `fipa_agent_identifier_s::resolvers`, and `fipa_agent_identifier_set_s::retain_order`.

Referenced by `fipa_envelope_ParseAgentIdentifier()`.

13.16 /home/dko/Projects/mobilec/trunk/src/include/acc.h File Reference

```
#include <sys/socket.h>
#include "macros.h"
```

Functions

- **STRUCT** (acc, struct [mc_platform_s](#) *[mc_platform](#); [THREAD_T](#) [thread](#); [THREAD_T](#) [message_handler_thread](#); [THREAD_T](#) [listen_thread](#); [THREAD_T](#) [udplisten_thread](#); [int](#) [waiting](#); [MUTEX_T](#) *[waiting_lock](#); [COND_T](#) *[waiting_cond](#); [int](#) [num_conn_threads](#); [THREAD_T](#) [conn_thread](#); [MUTEX_T](#) [conn_thread_lock](#); [COND_T](#) [conn_thread_cond](#); struct [connection_thread_arg_s](#) *[connection_thread_arg](#); [int](#) [num_msg_threads](#); [THREAD_T](#) [msg_thread](#); [MUTEX_T](#) [msg_thread_lock](#); [COND_T](#) [msg_thread_cond](#);) **STRUCT**([listen_thread_arg](#)
- **STRUCT** ([connection_thread_arg](#), struct [mc_platform_s](#) *[mc_platform](#); struct [connection_s](#) *[connection](#);) [acc_p](#) [acc_Initialize](#)(struct [mc_platform_s](#) *[mc_platform](#))
- [int](#) [acc_Destroy](#) ([acc_p](#) [acc](#))
- void [acc_Start](#) (struct [mc_platform_s](#) *[mc_platform](#))
- void * [acc_MessageHandlerThread](#) (void *[arg](#))
- void * [acc_Thread](#) (void *[arg](#))
- void * [acc_connection_Thread](#) (void *[arg](#))
- void * [listen_Thread](#) (void *[arg](#))
- void * [udplisten_Thread](#) (void *[arg](#))

Variables

- [u_long](#) [port](#)
- struct [sockaddr_in](#) * [addr](#)
- unsigned long [int](#) [client_fd](#)
- unsigned long [int](#) [server_fd](#)

13.16.1 Function Documentation

13.16.1.1 void* acc_connection_Thread (void * arg)

Definition at line 388 of file [acc.c](#).

References [mc_platform_s::acc](#), [fipa_agent_identifier_s::addresses](#), [agent_mailbox_Post\(\)](#), [mc_platform_s::agent_queue](#), [AGENT_UPDATE](#), [CANCEL](#), [CONNECT_THREAD_EXIT](#), [connection_Destroy\(\)](#), [mtp_http_s::content](#), [mtp_http_content_s::data](#), [FIPA_ACL](#), [fipa_acl_envelope_Destroy\(\)](#), [fipa_acl_envelope_New\(\)](#), [fipa_acl_message_Destroy\(\)](#), [fipa_acl_message_New\(\)](#), [fipa_acl_Parse\(\)](#), [fipa_agent_identifier_set_s::fipa_agent_identifiers](#), [fipa_envelope_Parse\(\)](#), [fipa_message_string_Destroy\(\)](#), [fipa_message_string_New\(\)](#), [mtp_http_s::http_performative](#), [HTTP_POST](#), [HTTP_PUT](#), [agent_s::mailbox](#), [agent_s::mc_platform](#), [mc_platform](#), [fipa_message_string_s::message](#), [message_s::message_body](#), [message_Destroy\(\)](#), [message_New\(\)](#), [mtp_http_s::message_parts](#), [mc_platform_s::message_queue](#), [message_s::message_type](#), [message_xml_parse\(\)](#), [MOBILE_AGENT](#), [mtp_http_Destroy\(\)](#), [mtp_http_InitializeFromConnection\(\)](#), [mtp_http_New\(\)](#), [MXML_NO_CALLBACK](#), [mxmLoadString\(\)](#), [N_UNDRSTD](#), [fipa_agent_identifier_s::name](#), [fipa_agent_identifier_set_s::num](#), [fipa_acl_envelope_s::num_params](#), [fipa_acl_envelope_s::params](#), [fipa_message_string_s::parse](#), [mc_platform_s::port](#),

mc_platform_s::private_key, QUER_IF, QUER_REF, RELAY, REQUEST, RETURN_MSG, fipa_url_s::str, SUBSCRIBE, mtp_http_s::target, fipa_acl_Param_s::to, fipa_url_sequence_s::urls, and message_s::xml_root.

Referenced by acc_Thread().

13.16.1.2 int acc_Destroy (acc_p acc)

Definition at line 93 of file acc.c.

References MC_SUCCESS.

Referenced by mc_platform_Destroy().

13.16.1.3 void* acc_MessageHandlerThread (void * arg)

Definition at line 105 of file acc.c.

References mc_platform_s::acc, agent_Initialize(), mc_platform_s::agent_queue, agent_s::agent_status, AGENT_UPDATE, mc_platform_s::ams, CANCEL, COND_BROADCAST, COND_WAIT, agent_s::datastate, FIPA_ACL, mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, agent_s::lock, MC_AGENT_NEUTRAL, mc_platform, MC_RECV_AGENT, MC_RECV_MESSAGE, MC_RECV_RETURN, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, message_Destroy(), mc_platform_s::message_queue, message_Send(), message_s::message_type, MOBILE_AGENT, MUTEX_LOCK, MUTEX_UNLOCK, N_UNDRSTD, agent_s::name, agent_datastate_s::persistent, QUER_IF, QUER_REF, mc_platform_s::quit, mc_platform_s::quit_lock, RELAY, REQUEST, RETURN_MSG, SUBSCRIBE, THREAD_EXIT, and message_s::to_address.

Referenced by acc_Start().

13.16.1.4 void acc_Start (struct mc_platform_s * mc_platform)

13.16.1.5 void* acc_Thread (void * arg)

Definition at line 287 of file acc.c.

References mc_platform_s::acc, acc_connection_Thread(), COND_BROADCAST, COND_WAIT, CONN_THREADS, mc_platform_s::connection_queue, mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, mc_platform, MC_RECV_CONNECTION, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::quit, mc_platform_s::quit_lock, THREAD_CREATE, THREAD_DETACH, THREAD_EXIT, and THREAD_T.

Referenced by acc_Start().

13.16.1.6 void* listen_Thread (void * arg)

Definition at line 652 of file acc.c.

References mc_platform_s::acc, connection_s::addr, connection_s::AES_key, mc_platform_s::agency, auth_conn_rece_key(), BACKLOG, connection_s::clientfd, COND_BROADCAST, connection_s::connect_id, connection_New(), mc_platform_s::connection_queue, agency_s::known_host_filename, mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, connection_s::nonce, mc_platform_s::port, mc_

platform_s::private_key, connection_s::remote_hostname, connection_s::serverfd, SOCKET_ERROR, mc_platform_s::sockfd, and THREAD_EXIT.

Referenced by acc_Start().

13.16.1.7 **STRUCT (connection_thread_arg, struct mc_platform_s *mc_platform; struct connection_s *connection;)**

13.16.1.8 **STRUCT (acc, struct mc_platform_s *mc_platform; THREAD_T thread; THREAD_T message_handler_thread; THREAD_T listen_thread; THREAD_T udplisten_thread; int waiting; MUTEX_T *waiting_lock; COND_T *waiting_cond; int num_conn_threads; THREAD_T conn_thread; MUTEX_T conn_thread_lock; COND_T conn_thread_cond; struct connection_thread_arg_s *connection_thread_arg; int num_msg_threads; THREAD_T msg_thread; MUTEX_T msg_thread_lock; COND_T msg_thread_cond;)**

13.16.1.9 **void* udplisten_Thread (void *arg)**

Definition at line 792 of file acc.c.

References buf, BUFLen, mc_platform_s::hostname, mc_platform, PACKAGE_VERSION, mc_platform_s::port, and UDPPORT.

Referenced by acc_Start().

13.16.2 Variable Documentation

13.16.2.1 **struct sockaddr_in* addr**

Definition at line 78 of file acc.h.

13.16.2.2 **unsigned long int client_fd**

Definition at line 79 of file acc.h.

Referenced by main(), and ssl_test().

13.16.2.3 **u_long port**

Definition at line 77 of file acc.h.

Referenced by MC_AclSend(), MC_MigrateAgent_chdl(), MC_SendAgentMigrationMessage_chdl(), MC_SendAgentMigrationMessageFile_chdl(), message_send_Thread(), and message_xml_parse__message().

13.16.2.4 **unsigned long int server_fd**

Definition at line 80 of file acc.h.

Referenced by main(), and ssl_test().

13.17 /home/dko/Projects/mobilec/trunk/src/include/agent.h File Reference

```
#include "agent_datastate.h"
#include "agent_task.h"
#include "libmc.h"
#include "message.h"
#include "macros.h"
#include "data_structures.h"
#include "agent_mailbox.h"
```

Data Structures

- struct [agent_s](#)

Typedefs

- typedef struct [agent_s](#) [agent_t](#)
- typedef [agent_t](#) * [MCAgent_t](#)
- typedef [agent_t](#) * [agent_p](#)

Functions

- [int agent_AddPersistentVariable](#) ([agent_p](#) agent, [int](#) task_num, const char *var_name)
- [agent_p agent_New](#) (void)
- [agent_p agent_NewBinary](#) (struct [mc_platform_s](#) *mc_platform)
- [agent_p agent_Copy](#) (const [agent_p](#) agent)
- [int agent_Destroy](#) ([agent_p](#) agent)
- [agent_p agent_Initialize](#) (struct [mc_platform_s](#) *mc_platform, [message_p](#) message, [int](#) id)
- void [agent_RunChScript](#) ([MCAgent_t](#) agent, struct [mc_platform_s](#) *global)
- void * [agent_RunChScriptThread](#) (void *agent)
- void * [agent_ChScriptInitVar](#) ([ChInterp_t](#) *interp)

13.17.1 Typedef Documentation

13.17.1.1 typedef [agent_t](#)* [agent_p](#)

Definition at line 111 of file agent.h.

13.17.1.2 typedef struct [agent_s](#) [agent_t](#)

Definition at line 109 of file agent.h.

13.17.1.3 typedef [agent_t](#)* [MCAgent_t](#)

Definition at line 110 of file agent.h.

13.17.2 Function Documentation

13.17.2.1 `int agent_AddPersistentVariable (agent_p agent, int task_num, const char * var_name)`

Definition at line 52 of file agent.c.

References `agent_s::agent_interp`, `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `CH_DATATYPE_SIZE`, `CHECK_NULL`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `agent_s::datastate`, `MC_ERR`, `agent_s::name`, `interpreter_variable_data_s::name`, `interpreter_variable_data_s::size`, `size`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

13.17.2.2 `void* agent_ChScriptInitVar (ChInterp_t * interp)`

Definition at line 492 of file agent.c.

References `MC_AclAddReceiver_chdl()`, `MC_AclAddReplyTo_chdl()`, `MC_AclDestroy_chdl()`, `MC_AclNew_chdl()`, `MC_AclPost_chdl()`, `MC_AclReply_chdl()`, `MC_AclRetrieve_chdl()`, `MC_AclSend_chdl()`, `MC_AclSetContent_chdl()`, `MC_AclSetConversationID_chdl()`, `MC_AclSetPerformative_chdl()`, `MC_AclSetProtocol_chdl()`, `MC_AclSetSender_chdl()`, `MC_AclWaitRetrieve_chdl()`, `MC_AddAgent_chdl()`, `MC_AgentVariableRetrieve_chdl()`, `MC_AgentVariableSave_chdl()`, `MC_Barrier_chdl()`, `MC_BarrierDelete_chdl()`, `MC_BarrierInit_chdl()`, `MC_CallAgentFunc_chdl()`, `MC_ComposeAgent_chdl()`, `MC_ComposeAgentS_chdl()`, `MC_CondBroadcast_chdl()`, `MC_CondReset_chdl()`, `MC_CondSignal_chdl()`, `MC_CondWait_chdl()`, `MC_DeleteAgent_chdl()`, `MC_DeleteAgentWG_chdl()`, `MC_DeregisterService_chdl()`, `MC_DestroyServiceSearchResult_chdl()`, `MC_End_chdl()`, `MC_FindAgentByID_chdl()`, `MC_FindAgentByName_chdl()`, `MC_GetAgentID_chdl()`, `MC_GetAgentName_chdl()`, `MC_GetAgentStatus_chdl()`, `MC_GetAgentXMLString_chdl()`, `MC_GetTimeOfDay_chdl()`, `MC_HaltAgency_chdl()`, `MC_MigrateAgent_chdl()`, `MC_MutexLock_chdl()`, `MC_MutexUnlock_chdl()`, `MC_PrintAgentCode_chdl()`, `MC_RegisterService_chdl()`, `MC_ResumeAgency_chdl()`, `MC_RetrieveAgent_chdl()`, `MC_RetrieveAgentCode_chdl()`, `MC_SaveData_chdl()`, `MC_SearchForService_chdl()`, `MC_SemaphorePost_chdl()`, `MC_SemaphoreWait_chdl()`, `MC_SendAgentMigrationMessage_chdl()`, `MC_SendAgentMigrationMessageFile_chdl()`, `MC_SendSteerCommand_chdl()`, `MC_SetAgentStatus_chdl()`, `MC_SetDefaultAgentStatus_chdl()`, `MC_SyncDelete_chdl()`, `MC_SyncInit_chdl()`, `MC_TerminateAgent_chdl()`, and `MC_TerminateAgentWG_chdl()`.

Referenced by `AP_QUEUE_STD_DEFN_TEMPLATE()`, and `mc_platform_Initialize()`.

13.17.2.3 `agent_p agent_Copy (const agent_p agent)`

Definition at line 134 of file agent.c.

References `agent_datastate_Copy()`, `agent_s::agent_interp`, `agent_mailbox_New()`, `agent_s::agent_persistent`, `agent_s::agent_status`, `agent_s::agent_type`, `agent_s::arrival_time`, `agent_s::datastate`, `agent_s::home`, `agent_s::home_port`, `agent_s::id`, `agent_s::lock`, `agent_s::mailbox`, `MUTEX_INIT`, `MUTEX_LOCK`, `MUTEX_T`, `agent_s::name`, `agent_s::orphan`, `agent_s::owner`, `agent_s::return_data`, and `agent_s::run_lock`.

Referenced by `MC_CopyAgent()`.

13.17.2.4 `int agent_Destroy (agent_p agent)`

Definition at line 414 of file agent.c.

References agent_datastate_Destroy(), agent_s::agent_interp, agent_mailbox_Destroy(), agent_s::agent_status, agent_s::datastate, agent_s::home, mc_platform_s::interpreter_queue, agent_s::lock, agent_s::mailbox, MC_AGENT_NEUTRAL, agent_s::mc_platform, MC_SUCCESS, MUTEX_DESTROY, MUTEX_LOCK, agent_s::name, agent_s::owner, agent_s::run_lock, agent_s::sender, and agent_s::wg_code.

Referenced by agent_Initialize().

13.17.2.5 agent_p agent_Initialize (struct mc_platform_s * mc_platform, message_p message, int id)

Definition at line 293 of file agent.c.

References agent_s::agent_address, agent_datastate_New(), agent_Destroy(), agent_mailbox_New(), agent_s::agent_pipe_active, agent_s::agent_pipe_ready_to_read, agent_s::agent_ready_to_send, agent_s::agent_script_ready, agent_s::agent_status, agent_s::agent_thread_id, agent_s::agent_type, message_s::agent_xml_flag, agent_xml_parse(), agent_s::arrival_time, agent_s::datastate, mc_platform_s::default_agentstatus, mc_platform_s::err, agent_s::home, mc_platform_s::hostname, agent_s::id, agent_s::lock, agent_s::mailbox, MC_ERR_PARSE, agent_s::mc_platform, MC_REMOTE_AGENT, MC_RETURN_AGENT, MC_WAIT_CH, message_s::message_type, MOBILE_AGENT, MUTEX_DESTROY, MUTEX_INIT, MUTEX_T, agent_s::orphan, mc_platform_s::port, RETURN_MSG, agent_s::run_lock, agent_s::sender, agent_datastate_s::xml_agent_root, message_s::xml_payload, message_s::xml_root, and agent_datastate_s::xml_root.

Referenced by acc_MessageHandlerThread(), and MC_SendAgentMigrationMessageFile().

13.17.2.6 agent_p agent_New (void)

Definition at line 197 of file agent.c.

References agent_s::lock, MUTEX_INIT, MUTEX_NEW, and agent_s::run_lock.

Referenced by MC_ComposeAgentS().

13.17.2.7 agent_p agent_NewBinary (struct mc_platform_s * mc_platform)

Definition at line 220 of file agent.c.

References agent_s::agent_address, agent_mailbox_New(), agent_s::agent_pipe_active, agent_s::agent_pipe_ready_to_read, agent_s::agent_ready_to_send, agent_s::agent_script_ready, agent_s::agent_status, agent_s::agent_thread_id, agent_s::arrival_time, agent_s::binary, agent_s::home, mc_platform_s::hostname, agent_s::id, agent_s::lock, agent_s::mailbox, MC_AGENT_ACTIVE, agent_s::mc_platform, MUTEX_INIT, MUTEX_T, agent_s::orphan, mc_platform_s::port, agent_s::run_lock, and agent_s::sender.

Referenced by MC_AddStationaryAgent().

13.17.2.8 void agent_RunChScript (MCAgent_t agent, struct mc_platform_s * global)

13.17.2.9 void* agent_RunChScriptThread (void * agent)

Definition at line 926 of file agent.c.

References agent_s::agent_interp, mc_platform_s::ams, COND_SIGNAL, agent_s::datastate, agent_s::id, mc_platform_s::interp_options, mc_platform_s::interpreter_queue, interpreter_queue_CreateRetrieve(),

interpreter_variable_data_Destroy(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), MC_AGENT_NEUTRAL, MC_EXEC_AGENT, agent_s::mc_platform, mc_platform, MC_RETURN_AGENT, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MC_WAIT_FINISHED, MC_WAIT_MESSGSEND, MUTEX_LOCK, MUTEX_UNLOCK, agent_s::name, SIGNAL, and agent_datastate_s::task_progress.

Referenced by agent_RunChScript().

13.18 /home/dko/Projects/mobilec/trunk/src/include/agent_datastate.h File Reference

```
#include <mxml.h>
#include "agent_task.h"
```

Data Structures

- struct [agent_datastate_s](#)

Typedefs

- typedef struct [agent_datastate_s](#) [agent_datastate_t](#)
- typedef [agent_datastate_t](#) * [agent_datastate_p](#)

Functions

- [agent_datastate_p](#) [agent_datastate_Copy](#) (const [agent_datastate_p](#) *datastate*)
- [agent_datastate_p](#) [agent_datastate_New](#) (void)
- int [agent_datastate_Destroy](#) ([agent_datastate_p](#) *agent_datastate*)

13.18.1 Typedef Documentation

13.18.1.1 typedef [agent_datastate_t](#)* [agent_datastate_p](#)

Definition at line 73 of file [agent_datastate.h](#).

13.18.1.2 typedef struct [agent_datastate_s](#) [agent_datastate_t](#)

13.18.2 Function Documentation

13.18.2.1 [agent_datastate_p](#) [agent_datastate_Copy](#) (const [agent_datastate_p](#) *datastate*)

Definition at line 47 of file [agent_datastate.c](#).

References [agent_datastate_s::agent_code](#), [agent_datastate_s::agent_code_ids](#), [agent_datastate_s::agent_codes](#), [agent_datastate_New\(\)](#), [agent_task_Copy\(\)](#), [agent_datastate_s::init_agent_status](#), [agent_datastate_s::number_of_tasks](#), [agent_datastate_s::persistent](#), [agent_datastate_s::return_data](#), [agent_datastate_s::task_progress](#), and [agent_datastate_s::tasks](#).

Referenced by [agent_Copy\(\)](#).

13.18.2.2 int [agent_datastate_Destroy](#) ([agent_datastate_p](#) *agent_datastate*)

Definition at line 136 of file [agent_datastate.c](#).

References [agent_datastate_s::agent_code_ids](#), [agent_datastate_s::agent_codes](#), [agent_task_Destroy\(\)](#), [MC_SUCCESS](#), [mxmlDelete\(\)](#), [agent_datastate_s::number_of_tasks](#), [agent_datastate_s::tasks](#), and [agent_datastate_s::xml_root](#).

Referenced by agent_Destroy().

13.18.2.3 agent_datastate_p agent_datastate_New (void)

Definition at line 115 of file agent_datastate.c.

References agent_datastate_s::agent_code, CHECK_NULL, agent_datastate_s::init_agent_status, agent_datastate_s::number_of_tasks, agent_datastate_s::persistent, agent_datastate_s::progress_modifier, agent_datastate_s::return_data, agent_datastate_s::task_progress, agent_datastate_s::tasks, agent_datastate_s::xml_agent_root, and agent_datastate_s::xml_root.

Referenced by agent_datastate_Copy(), agent_Initialize(), and MC_ComposeAgentS().

13.19 /home/dko/Projects/mobilec/trunk/src/include/agent_lib.h File Reference

Functions

- EXPORTCH [int MC_AclDestroy_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclNew_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclPost_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclReply_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclRetrieve_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSend_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclWaitRetrieve_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetProtocol_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetConversationID_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetPerformative_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetSender_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclAddReceiver_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclAddReplyTo_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclSetContent_chdl](#) (void *varg)
- EXPORTCH [int MC_AddAgent_chdl](#) (void *varg)
- EXPORTCH const void * [MC_AgentVariableRetrieve_chdl](#) (void *varg)
- EXPORTCH [int MC_AgentVariableSave_chdl](#) (void *varg)
- EXPORTCH [int MC_Barrier_chdl](#) (void *varg)
- EXPORTCH [int MC_BarrierDelete_chdl](#) (void *varg)
- EXPORTCH [int MC_BarrierInit_chdl](#) (void *varg)
- EXPORTCH [int MC_CallAgentFunc_chdl](#) (void *varg)
- EXPORTCH [int MC_ComposeAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_ComposeAgentS_chdl](#) (void *varg)
- EXPORTCH [int MC_CondBroadcast_chdl](#) (void *varg)
- EXPORTCH [int MC_CondWait_chdl](#) (void *varg)
- EXPORTCH [int MC_CondReset_chdl](#) (void *varg)
- EXPORTCH [int MC_CondSignal_chdl](#) (void *varg)
- EXPORTCH [int MC_DeleteAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_DeleteAgentWG_chdl](#) (void *varg)
- EXPORTCH [int MC_DeregisterService_chdl](#) (void *varg)
- EXPORTCH [int MC_DestroyServiceSearchResult_chdl](#) (void *varg)
- EXPORTCH [int MC_End_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_FindAgentByID_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_FindAgentByName_chdl](#) (void *varg)
- EXPORTCH [int MC_GetAgentID_chdl](#) (void *varg)
- EXPORTCH char * [MC_GetAgentName_chdl](#) (void *varg)
- EXPORTCH [int MC_GetAgentNumTasks_chdl](#) (void *varg)
- EXPORTCH char * [MC_GetAgentXMLString_chdl](#) (void *varg)
- EXPORTCH [int MC_GetTimeOfDay_chdl](#) (void *varg)
- EXPORTCH [int MC_HaltAgency_chdl](#) (void *varg)
- EXPORTCH [int MC_PrintAgentCode_chdl](#) (void *varg)
- EXPORTCH [int MC_MigrateAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_MutexLock_chdl](#) (void *varg)
- EXPORTCH [int MC_MutexUnlock_chdl](#) (void *varg)
- EXPORTCH [int MC_RegisterService_chdl](#) (void *varg)

- EXPORTCH [int MC_ResumeAgency_chdl](#) (void *varg)
- EXPORTCH [MC_Agent_t MC_RetrieveAgent_chdl](#) (void *varg)
- EXPORTCH [char * MC_RetrieveAgentCode_chdl](#) (void *varg)
- EXPORTCH [char * MC_SearchForService_chdl](#) (void *varg)
- EXPORTCH [int MC_SemaphorePost_chdl](#) (void *varg)
- EXPORTCH [int MC_SemaphoreWait_chdl](#) (void *varg)
- EXPORTCH [int MC_SendSteerCommand_chdl](#) (void *varg)
- EXPORTCH [int MC_TerminateAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_TerminateAgentWG_chdl](#) (void *varg)
- EXPORTCH [int MC_GetAgentStatus_chdl](#) (void *varg)
- EXPORTCH [int MC_SaveData_chdl](#) (void *varg)
- EXPORTCH [int MC_SendAgentMigrationMessage_chdl](#) (void *varg)
- EXPORTCH [int MC_SendAgentMigrationMessageFile_chdl](#) (void *varg)
- EXPORTCH [int MC_SetAgentStatus_chdl](#) (void *varg)
- EXPORTCH [int MC_SetDefaultAgentStatus_chdl](#) (void *varg)
- EXPORTCH [int MC_SyncDelete_chdl](#) (void *varg)
- EXPORTCH [int MC_SyncInit_chdl](#) (void *varg)

13.19.1 Function Documentation

13.19.1.1 EXPORTCH void* MC_AclAddReceiver_chdl (void * *varg*)

Definition at line 2271 of file libmc.c.

References [MC_AclAddReceiver\(\)](#).

Referenced by [agent_ChScriptInitVar\(\)](#).

13.19.1.2 EXPORTCH void* MC_AclAddReplyTo_chdl (void * *varg*)

Definition at line 2291 of file libmc.c.

References [MC_AclAddReplyTo\(\)](#).

Referenced by [agent_ChScriptInitVar\(\)](#).

13.19.1.3 EXPORTCH int MC_AclDestroy_chdl (void * *varg*)

Definition at line 2081 of file libmc.c.

References [MC_AclDestroy\(\)](#).

Referenced by [agent_ChScriptInitVar\(\)](#).

13.19.1.4 EXPORTCH void* MC_AclNew_chdl (void * *varg*)

Definition at line 2096 of file libmc.c.

References [MC_AclNew\(\)](#).

Referenced by [agent_ChScriptInitVar\(\)](#).

13.19.1.5 EXPORTCH void* MC_AclPost_chdl (void * *varg*)

Definition at line 2104 of file libmc.c.

References MC_AclPost().

Referenced by agent_ChScriptInitVar().

13.19.1.6 EXPORTCH void* MC_AclReply_chdl (void * *varg*)

Definition at line 2122 of file libmc.c.

References MC_AclReply().

Referenced by agent_ChScriptInitVar().

13.19.1.7 EXPORTCH void* MC_AclRetrieve_chdl (void * *varg*)

Definition at line 2138 of file libmc.c.

References MC_AclRetrieve().

Referenced by agent_ChScriptInitVar().

13.19.1.8 EXPORTCH int MC_AclSend_chdl (void * *varg*)

Definition at line 2154 of file libmc.c.

References CHECK_NULL, MC_AclSend(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.9 EXPORTCH void* MC_AclSetContent_chdl (void * *varg*)

Definition at line 2311 of file libmc.c.

References fipa_acl_message_s::content, and MC_AclSetContent().

Referenced by agent_ChScriptInitVar().

13.19.1.10 EXPORTCH void* MC_AclSetConversationID_chdl (void * *varg*)

Definition at line 2215 of file libmc.c.

References MC_AclSetConversationID().

Referenced by agent_ChScriptInitVar().

13.19.1.11 EXPORTCH void* MC_AclSetPerformative_chdl (void * *varg*)

Definition at line 2233 of file libmc.c.

References MC_AclSetPerformative(), and fipa_acl_message_s::performative.

Referenced by agent_ChScriptInitVar().

13.19.1.12 EXPORTCH void* MC_AclSetProtocol_chdl (void * *varg*)

Definition at line 2196 of file libmc.c.

References MC_AclSetProtocol(), and fipa_acl_message_s::protocol.

Referenced by agent_ChScriptInitVar().

13.19.1.13 EXPORTCH void* MC_AclSetSender_chdl (void * *varg*)

Definition at line 2251 of file libmc.c.

References MC_AclSetSender().

Referenced by agent_ChScriptInitVar().

13.19.1.14 EXPORTCH void* MC_AclWaitRetrieve_chdl (void * *varg*)

Definition at line 2178 of file libmc.c.

References MC_AclWaitRetrieve().

Referenced by agent_ChScriptInitVar().

13.19.1.15 EXPORTCH int MC_AddAgent_chdl (void * *varg*)

Definition at line 2331 of file libmc.c.

References CHECK_NULL, MC_AddAgent(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.16 EXPORTCH const void* MC_AgentVariableRetrieve_chdl (void * *varg*)

Definition at line 2355 of file libmc.c.

References MC_AgentVariableRetrieve().

Referenced by agent_ChScriptInitVar().

13.19.1.17 EXPORTCH int MC_AgentVariableSave_chdl (void * *varg*)

Definition at line 2379 of file libmc.c.

References MC_AgentVariableSave().

Referenced by agent_ChScriptInitVar().

13.19.1.18 EXPORTCH int MC_Barrier_chdl (void * *varg*)

Definition at line 2431 of file libmc.c.

References CHECK_NULL, MC_Barrier(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.19 EXPORTCH int MC_BarrierDelete_chdl (void * *varg*)

Definition at line 2454 of file libmc.c.

References CHECK_NULL, MC_BarrierDelete(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.20 EXPORTCH int MC_BarrierInit_chdl (void * *varg*)

Definition at line 2477 of file libmc.c.

References CHECK_NULL, MC_BarrierInit(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.21 EXPORTCH int MC_CallAgentFunc_chdl (void * *varg*)

Definition at line 2401 of file libmc.c.

References MC_CallAgentFuncVar().

Referenced by agent_ChScriptInitVar().

13.19.1.22 EXPORTCH int MC_ComposeAgent_chdl (void * *varg*)

Definition at line 2525 of file libmc.c.

References MC_ComposeAgentS().

Referenced by agent_ChScriptInitVar().

13.19.1.23 EXPORTCH int MC_ComposeAgentS_chdl (void * *varg*)

Definition at line 2566 of file libmc.c.

References MC_ComposeAgentS().

Referenced by agent_ChScriptInitVar().

13.19.1.24 EXPORTCH int MC_CondBroadcast_chdl (void * *varg*)

Definition at line 2502 of file libmc.c.

References CHECK_NULL, MC_CondBroadcast(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.25 EXPORTCH int MC_CondReset_chdl (void * *varg*)

Definition at line 2632 of file libmc.c.

References CHECK_NULL, MC_CondReset(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.26 EXPORTCH int MC_CondSignal_chdl (void * *varg*)

Definition at line 2609 of file libmc.c.

References CHECK_NULL, MC_CondSignal(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.27 EXPORTCH int MC_CondWait_chdl (void * *varg*)

Definition at line 2655 of file libmc.c.

References CHECK_NULL, MC_CondWait(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.28 EXPORTCH int MC_DeleteAgent_chdl (void * *varg*)

Definition at line 2677 of file libmc.c.

References MC_DeleteAgent(), MC_ERR_NOT_FOUND, and MC_FindAgentByName().

Referenced by agent_ChScriptInitVar().

13.19.1.29 EXPORTCH int MC_DeleteAgentWG_chdl (void * *varg*)

Definition at line 2697 of file libmc.c.

References MC_DeleteAgentWG(), MC_ERR_NOT_FOUND, and MC_FindAgentByName().

Referenced by agent_ChScriptInitVar().

13.19.1.30 EXPORTCH int MC_DeregisterService_chdl (void * *varg*)

Definition at line 2746 of file libmc.c.

References CHECK_NULL, MC_DeregisterService(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.31 EXPORTCH int MC_DestroyServiceSearchResult_chdl (void * *varg*)

Definition at line 2719 of file libmc.c.

References MC_DestroyServiceSearchResult().

Referenced by agent_ChScriptInitVar().

13.19.1.32 EXPORTCH int MC_End_chdl (void * *varg*)

Definition at line 2773 of file libmc.c.

References CHECK_NULL, MC_End(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.33 EXPORTCH MCAgent_t MC_FindAgentByID_chdl (void * *varg*)

Definition at line 2790 of file libmc.c.

References CHECK_NULL, MC_FindAgentByID(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.34 EXPORTCH MCAgent_t MC_FindAgentByName_chdl (void * *varg*)

Definition at line 2813 of file libmc.c.

References CHECK_NULL, MC_FindAgentByName(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.35 EXPORTCH int MC_GetAgentID_chdl (void * *varg*)

Definition at line 2860 of file libmc.c.

References MC_GetAgentID().

Referenced by agent_ChScriptInitVar().

13.19.1.36 EXPORTCH char* MC_GetAgentName_chdl (void * *varg*)

Definition at line 2876 of file libmc.c.

References MC_GetAgentName().

Referenced by agent_ChScriptInitVar().

13.19.1.37 EXPORTCH int MC_GetAgentNumTasks_chdl (void * *varg*)

Definition at line 2892 of file libmc.c.

References MC_GetAgentNumTasks().

13.19.1.38 EXPORTCH int MC_GetAgentStatus_chdl (void * *varg*)

Definition at line 2908 of file libmc.c.

References MC_GetAgentStatus().

Referenced by agent_ChScriptInitVar().

13.19.1.39 EXPORTCH char* MC_GetAgentXMLString_chdl (void * *varg*)

Definition at line 2924 of file libmc.c.

References MC_GetAgentXMLString().

Referenced by agent_ChScriptInitVar().

13.19.1.40 EXPORTCH int MC_GetTimeOfDay_chdl (void * *varg*)

Definition at line 2940 of file libmc.c.

Referenced by agent_ChScriptInitVar().

13.19.1.41 EXPORTCH int MC_HaltAgency_chdl (void * *varg*)

Definition at line 2955 of file libmc.c.

References CHECK_NULL, MC_HaltAgency(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.42 EXPORTCH int MC_MigrateAgent_chdl (void * *varg*)

Definition at line 2972 of file libmc.c.

References MC_MigrateAgent(), and port.

Referenced by agent_ChScriptInitVar().

13.19.1.43 EXPORTCH int MC_MutexLock_chdl (void * *varg*)

Definition at line 2991 of file libmc.c.

References CHECK_NULL, MC_MutexLock(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.44 EXPORTCH int MC_MutexUnlock_chdl (void * *varg*)

Definition at line 3015 of file libmc.c.

References CHECK_NULL, MC_MutexUnlock(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.19.1.45 EXPORTCH int MC_PrintAgentCode_chdl (void * *varg*)

Definition at line 3039 of file libmc.c.

References MC_PrintAgentCode().

Referenced by agent_ChScriptInitVar().

13.19.1.46 EXPORTCH int MC_RegisterService_chdl (void * *varg*)

Definition at line 3055 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_RegisterService().

Referenced by agent_ChScriptInitVar().

13.19.1.47 EXPORTCH int MC_ResumeAgency_chdl (void * *varg*)

Definition at line 3092 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_ResumeAgency().

Referenced by agent_ChScriptInitVar().

13.19.1.48 EXPORTCH MCAgent_t MC_RetrieveAgent_chdl (void * *varg*)

Definition at line 3110 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_RetrieveAgent().

Referenced by agent_ChScriptInitVar().

13.19.1.49 EXPORTCH char* MC_RetrieveAgentCode_chdl (void * *varg*)

Definition at line 3127 of file libmc.c.

References MC_RetrieveAgentCode().

Referenced by agent_ChScriptInitVar().

13.19.1.50 EXPORTCH int MC_SaveData_chdl (void * *varg*)

Definition at line 3143 of file libmc.c.

References agent_task_s::agent_variable_list, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, agent_s::datastate, interpreter_variable_data_New(), interpreter_variable_data_s::name, interpreter_variable_data_s::size, size, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by agent_ChScriptInitVar().

13.19.1.51 EXPORTCH char* MC_SearchForService_chdl (void * *varg*)

Definition at line 3178 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SearchForService().

Referenced by agent_ChScriptInitVar().

13.19.1.52 EXPORTCH int MC_SemaphorePost_chdl (void * *varg*)

Definition at line 3219 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SemaphorePost().

Referenced by agent_ChScriptInitVar().

13.19.1.53 EXPORTCH int MC_SemaphoreWait_chdl (void * *varg*)

Definition at line 3243 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SemaphoreWait().

Referenced by agent_ChScriptInitVar().

13.19.1.54 EXPORTCH int MC_SendAgentMigrationMessage_chdl (void * *varg*)

Definition at line 3267 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, MC_SendAgentMigrationMessage(), and port.

Referenced by agent_ChScriptInitVar().

13.19.1.55 EXPORTCH int MC_SendAgentMigrationMessageFile_chdl (void * *varg*)

Definition at line 3292 of file libmc.c.

References MC_SendAgentMigrationMessageFile(), and port.

Referenced by agent_ChScriptInitVar().

13.19.1.56 EXPORTCH int MC_SendSteerCommand_chdl (void * *varg*)

Definition at line 3312 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SendSteerCommand().

Referenced by agent_ChScriptInitVar().

13.19.1.57 EXPORTCH int MC_SetAgentStatus_chdl (void * *varg*)

Definition at line 3335 of file libmc.c.

References MC_SetAgentStatus().

Referenced by agent_ChScriptInitVar().

13.19.1.58 EXPORTCH int MC_SetDefaultAgentStatus_chdl (void * *varg*)

Definition at line 3353 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SetDefaultAgentStatus().

Referenced by agent_ChScriptInitVar().

13.19.1.59 EXPORTCH int MC_SyncDelete_chdl (void * *varg*)

Definition at line 3376 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SyncDelete().

Referenced by agent_ChScriptInitVar().

13.19.1.60 EXPORTCH int MC_SyncInit_chdl (void * *varg*)

Definition at line 3399 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SyncInit().

Referenced by agent_ChScriptInitVar().

13.19.1.61 EXPORTCH int MC_TerminateAgent_chdl (void * *varg*)

Definition at line 3423 of file libmc.c.

References MC_ERR_NOT_FOUND, MC_FindAgentByName(), and MC_TerminateAgent().

Referenced by agent_ChScriptInitVar().

13.19.1.62 EXPORTCH int MC_TerminateAgentWG_chdl (void * *varg*)

Definition at line 3444 of file libmc.c.

References MC_ERR_NOT_FOUND, MC_FindAgentByName(), and MC_TerminateAgentWG().

Referenced by agent_ChScriptInitVar().

13.20 /home/dko/Projects/mobilec/trunk/src/include/agent_mailbox.h File Reference

Data Structures

- struct [agent_mailbox_s](#)

Typedefs

- typedef struct [agent_mailbox_s](#) [agent_mailbox_t](#)
- typedef [agent_mailbox_t](#) * [agent_mailbox_p](#)

Functions

- [agent_mailbox_p](#) [agent_mailbox_New](#) (void)
- [agent_mailbox_p](#) [agent_mailbox_Copy](#) ([agent_mailbox_p](#) src)
- int [agent_mailbox_Destroy](#) ([agent_mailbox_t](#) *mailbox)
- int [agent_mailbox_Post](#) ([agent_mailbox_p](#) mailbox, [fipa_acl_message_t](#) *message)
- [fipa_acl_message_t](#) * [agent_mailbox_Retrieve](#) ([agent_mailbox_p](#) mailbox)
- [fipa_acl_message_t](#) * [agent_mailbox_WaitRetrieve](#) ([agent_mailbox_p](#) mailbox)

13.20.1 Typedef Documentation

13.20.1.1 typedef [agent_mailbox_t](#)* [agent_mailbox_p](#)

Definition at line 12 of file [agent_mailbox.h](#).

13.20.1.2 typedef struct [agent_mailbox_s](#) [agent_mailbox_t](#)

13.20.2 Function Documentation

13.20.2.1 [agent_mailbox_p](#) [agent_mailbox_Copy](#) ([agent_mailbox_p](#) src)

Definition at line 21 of file [agent_mailbox.c](#).

References [agent_mailbox_New\(\)](#), and [agent_mailbox_s::mail_queue](#).

13.20.2.2 int [agent_mailbox_Destroy](#) ([agent_mailbox_t](#) * mailbox)

Definition at line 29 of file [agent_mailbox.c](#).

References [agent_mailbox_s::mail_queue](#).

Referenced by [agent_Destroy\(\)](#).

13.20.2.3 [agent_mailbox_p](#) [agent_mailbox_New](#) (void)

Definition at line 12 of file [agent_mailbox.c](#).

References [agent_mailbox_s::mail_queue](#).

Referenced by agent_Copy(), agent_Initialize(), agent_mailbox_Copy(), and agent_NewBinary().

13.20.2.4 int agent_mailbox_Post (agent_mailbox_p *mailbox*, fipa_acl_message_t * *message*)

Definition at line 38 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by acc_connection_Thread(), and MC_AclPost().

13.20.2.5 fipa_acl_message_t* agent_mailbox_Retrieve (agent_mailbox_p *mailbox*)

Definition at line 44 of file agent_mailbox.c.

References agent_mailbox_s::mail_queue.

Referenced by agent_mailbox_WaitRetrieve(), and MC_AclRetrieve().

13.20.2.6 fipa_acl_message_t* agent_mailbox_WaitRetrieve (agent_mailbox_p *mailbox*)

Definition at line 49 of file agent_mailbox.c.

References agent_mailbox_Retrieve(), COND_WAIT, agent_mailbox_s::mail_queue, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_AclWaitRetrieve().

13.21 /home/dko/Projects/mobilec/trunk/src/include/agent_task.h File Reference

```
#include "interpreter_variable_data.h"
#include "data_structures.h"
```

Data Structures

- struct [agent_task_s](#)

Typedefs

- typedef struct [agent_task_s](#) [agent_task_t](#)
- typedef [agent_task_t](#) * [agent_task_p](#)

Functions

- [agent_task_p](#) [agent_task_New](#) (void)
- [agent_task_p](#) [agent_task_Copy](#) ([agent_task_p](#) task)
- int [agent_task_Destroy](#) ([agent_task_p](#) agent_task)

13.21.1 Typedef Documentation

13.21.1.1 typedef [agent_task_t](#)* [agent_task_p](#)

Definition at line 59 of file [agent_task.h](#).

13.21.1.2 typedef struct [agent_task_s](#) [agent_task_t](#)

13.21.2 Function Documentation

13.21.2.1 [agent_task_p](#) [agent_task_Copy](#) ([agent_task_p](#) task)

Definition at line 66 of file [agent_task.c](#).

References [agent_task_s::agent_return_data](#), [agent_task_s::agent_variable_list](#), [agent_task_s::code_id](#), [agent_task_s::init_agent_status](#), [interpreter_variable_data_Copy\(\)](#), [ListAdd\(\)](#), [ListSearch\(\)](#), [agent_task_s::num_saved_variables](#), [agent_task_s::number_of_elements](#), [agent_task_s::persistent](#), [agent_task_s::saved_variables](#), [agent_task_s::server_name](#), [agent_task_s::size_of_element_array](#), and [agent_task_s::var_name](#).

Referenced by [agent_datastate_Copy\(\)](#).

13.21.2.2 int [agent_task_Destroy](#) ([agent_task_p](#) agent_task)

Definition at line 132 of file [agent_task.c](#).

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_task_s::code_id`, `interpreter_variable_data_Destroy()`, `MC_SUCCESS`, `agent_task_s::saved_variables`, `agent_task_s::server_name`, and `agent_task_s::var_name`.

Referenced by `agent_datastate_Destroy()`.

13.21.2.3 `agent_task_p agent_task_New (void)`

Definition at line 48 of file `agent_task.c`.

References `agent_task_s::agent_variable_list`, `agent_task_s::num_saved_variables`, and `agent_task_s::saved_variables`.

Referenced by `agent_xml_parse__tasks()`, and `MC_ComposeAgentS()`.

13.22 /home/dko/Projects/mobilec/trunk/src/include/ams.h File Reference

```
#include "macros.h"
```

Functions

- **STRUCT** (ams, struct [mc_platform_s](#) *[mc_platform](#);MUTEX_T *[runflag_lock](#);COND_T *[runflag_cond](#);int [run](#);int [waiting](#);MUTEX_T *[waiting_lock](#);COND_T *[waiting_cond](#);THREAD_T [thread](#);) [ams_p](#) [ams_Initialize](#)(struct [mc_platform_s](#) *_[mc_platform](#))
- int [ams_Destroy](#) ([ams_p](#) [ams](#))
- int [ams_RemoveFinishedAgents](#) ([ams_p](#) [ams](#))
- int [ams_ManageAgentList](#) ([ams_p](#) [ams](#))
- void [ams_Print](#) ([ams_p](#) [ams](#))
- void [ams_Start](#) (struct [mc_platform_s](#) *[mc_platform](#))
- void * [ams_Thread](#) (void *[arg](#))

13.22.1 Function Documentation

13.22.1.1 int [ams_Destroy](#) ([ams_p](#) [ams](#))

Definition at line 46 of file [ams.c](#).

References [COND_DESTROY](#), [MC_SUCCESS](#), and [MUTEX_DESTROY](#).

Referenced by [mc_platform_Destroy](#)().

13.22.1.2 int [ams_ManageAgentList](#) ([ams_p](#) [ams](#))

Definition at line 116 of file [ams.c](#).

References [agent_RunChScript](#)(), [agent_s::agent_status](#), [agent_s::binary](#), [ListSearch](#)(), [agent_s::lock](#), [MC_AGENT_ACTIVE](#), [MC_AGENT_NEUTRAL](#), [MC_TerminateAgent](#)(), [MC_WAIT_CH](#), [MC_WAIT_FINISHED](#), [MC_WAIT_MESSGSEND](#), [message_Destroy](#)(), [message_InitializeFromAgent](#)(), [message_New](#)(), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), [agent_s::name](#), [agent_s::orphan](#), [mc_platform_s::quit](#), [mc_platform_s::quit_lock](#), and [agent_s::run_lock](#).

Referenced by [ams_Thread](#)().

13.22.1.3 void [ams_Print](#) ([ams_p](#) [ams](#))

Definition at line 84 of file [ams.c](#).

References [agent_s::agent_status](#), [agent_s::connect_id](#), [agent_s::id](#), [ListSearch](#)(), [MUTEX_LOCK](#), and [MUTEX_UNLOCK](#).

13.22.1.4 `int ams_RemoveFinishedAgents (ams_p ams)`

13.22.1.5 `void ams_Start (struct mc_platform_s * mc_platform)`

13.22.1.6 `void* ams_Thread (void * arg)`

Definition at line 254 of file `ams.c`.

References `mc_platform_s::ams`, `ams_ManageAgentList()`, `COND_BROADCAST`, `COND_WAIT`, `mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_lock`, and `THREAD_EXIT`.

Referenced by `ams_Start()`.

13.22.1.7 `STRUCT (ams, struct mc_platform_s *mc_platform;MUTEX_T *runflag_lock;COND_T *runflag_cond;int run;int waiting;MUTEX_T *waiting_lock;COND_T *waiting_cond;THREAD_T thread;)`

13.23 /home/dko/Projects/mobilec/trunk/src/include/ap_queue_template.h File Reference

```
#include "macros.h"
#include "mc_error.h"
#include <embedch.h>
```

Data Structures

- struct [AP_GENERIC_s](#)

Defines

- #define [AP_QUEUE_DECL_TEMPLATE](#)(name, node_type)
- #define [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(name, func_name, return_type, search_type) return_type name##_##_func_name(name##_p name, const search_type key);
- #define [AP_QUEUE_STD_DEFN_TEMPLATE](#)(name, node_type)
- #define [AP_QUEUE_SEARCH_TEMPLATE](#)(name, func_name, node_type, search_type, search_expression)
- #define [AP_QUEUE_REMOVE_TEMPLATE](#)(name, func_name, node_type, search_type, search_expression)

Typedefs

- typedef struct [AP_GENERIC_s](#) [AP_GENERIC_t](#)
- typedef [AP_GENERIC_t](#) * [AP_GENERIC_p](#)

Functions

- [AP_GENERIC_p](#) [interpreter_queue_CreateRetrieve](#) (struct [interpreter_queue_s](#) *queue, [ChOptions_t](#) *interp_options)

13.23.1 Define Documentation

13.23.1.1 #define AP_QUEUE_DECL_TEMPLATE(name, node_type)

Value:

```
typedef struct name##_s \
{ \
    int size; \
    list_p list; \
    MUTEX_T* lock; \
    COND_T* cond; \
} name##_t; \
\
typedef name##_t* name##_p; \
\
name##_p name##_New( void ); \
int name##_Destroy( name##_p name ); \
```

```

int name##_Add( name##_p name, struct node_type##_s* node ); \
name##_p name##_Copy( name##_p name ); \
struct node_type##_s* name##_Pop( name##_p name ); \
struct node_type##_s* name##_WaitPop( name##_p name ); \
struct node_type##_s* name##_SearchIndex( name##_p name, int index ); \
int name##_RemoveIndex( name##_p name, int index );

```

Definition at line 46 of file ap_queue_template.h.

13.23.1.2 **#define AP_QUEUE_GENERIC_DECL_TEMPLATE(name, func_name, return_type, search_type) return_type name##_##func_name(name##_p name, const search_type key);**

Definition at line 66 of file ap_queue_template.h.

13.23.1.3 **#define AP_QUEUE_REMOVE_TEMPLATE(name, func_name, node_type, search_type, search_expression)**

Value:

```

int name##_##func_name( name##_p name, const search_type key ) \
{ \
    int err_code = MC_ERR_NOT_FOUND; \
    struct listNode_s* parsenode; \
    struct node_type##_s* node; \
    node = NULL; \
    \
    MUTEX_LOCK( name->lock ); \
    if ( name->list->listhead == NULL ) { \
        MUTEX_UNLOCK( name->lock ); \
        return MC_ERR_NOT_FOUND; \
    } \
    for( \
        parsenode = (listNode_t*)name->list->listhead; \
        parsenode->next != NULL; \
        parsenode = (listNode_t*)parsenode->next \
    ) \
    { \
        node = (node_type##_t*)parsenode->node_data; \
        if ( search_expression ) { \
            break; \
            err_code = MC_SUCCESS; \
        } \
    } \
    MUTEX_UNLOCK( name->lock ); \
    return err_code; \
}

```

Definition at line 224 of file ap_queue_template.h.

13.23.1.4 **#define AP_QUEUE_SEARCH_TEMPLATE(name, func_name, node_type, search_type, search_expression)**

Value:

```

struct node_type##_s* name##_##func_name( name##_p name, const search_type key ) \
{ \
    listNode_t* parsenode; \

```

```

struct node_type##_s* node; \
struct node_type##_s* ret = NULL; \
node = NULL; \
\
MUTEX_LOCK(name->lock); \
if (name->list->listhead == NULL) { \
    MUTEX_UNLOCK(name->lock); \
    return NULL; \
} \
for( \
    parsenode = (listNode_t*)name->list->listhead; \
    parsenode != NULL; \
    parsenode = (listNode_t*)parsenode->next \
) \
{ \
    node = (node_type##_t*)parsenode->node_data; \
    if (search_expression){ \
        ret = node; \
        break; \
    } \
} \
MUTEX_UNLOCK(name->lock); \
return ret; \
}

```

Definition at line 194 of file ap_queue_template.h.

13.23.1.5 #define AP_QUEUE_STD_DEFN_TEMPLATE(name, node_type)

Definition at line 69 of file ap_queue_template.h.

13.23.2 Typedef Documentation

13.23.2.1 typedef AP_GENERIC_t* AP_GENERIC_p

Definition at line 43 of file ap_queue_template.h.

13.23.2.2 typedef struct AP_GENERIC_s AP_GENERIC_t

Definition at line 42 of file ap_queue_template.h.

13.23.3 Function Documentation

13.23.3.1 AP_GENERIC_p interpreter_queue_CreateRetrieve (struct interpreter_queue_s * queue, ChOptions_t * interp_options)

Referenced by agent_RunChScriptThread().

13.24 /home/dko/Projects/mobilec/trunk/src/include/barrier.h File Reference

```
#include "macros.h"
#include "../mc_list/list.h"
#include "mc_rwlock.h"
```

Data Structures

- struct [barrier_node_s](#)
- struct [barrier_queue_s](#)

Typedefs

- typedef struct [barrier_node_s](#) [barrier_node_t](#)
- typedef [barrier_node_t](#) * [barrier_node_p](#)
- typedef struct [barrier_queue_s](#) [barrier_queue_t](#)
- typedef [barrier_queue_t](#) * [barrier_queue_p](#)

Functions

- [barrier_node_p](#) [barrier_node_Initialize](#) (int id, int num_registered)
- int [barrier_node_Destroy](#) ([barrier_node_p](#) node)
- int [barrier_queue_Add](#) ([barrier_queue_p](#) list, [barrier_node_p](#) node)
- int [barrier_queue_Delete](#) (int id, [barrier_queue_p](#) list)
- int [barrier_queue_Destroy](#) ([barrier_queue_p](#) queue)
- [barrier_node_p](#) [barrier_queue_Get](#) ([barrier_queue_p](#) list, int id)
- [barrier_queue_p](#) [barrier_queue_New](#) (void)
- [barrier_node_p](#) [barrier_queue_Pop](#) ([barrier_queue_p](#) queue)

13.24.1 Typedef Documentation

13.24.1.1 typedef [barrier_node_t](#)* [barrier_node_p](#)

Definition at line 49 of file [barrier.h](#).

13.24.1.2 typedef struct [barrier_node_s](#) [barrier_node_t](#)

13.24.1.3 typedef [barrier_queue_t](#)* [barrier_queue_p](#)

Definition at line 57 of file [barrier.h](#).

13.24.1.4 typedef struct barrier_queue_s barrier_queue_t

13.24.2 Function Documentation

13.24.2.1 int barrier_node_Destroy (barrier_node_p node)

Definition at line 70 of file barrier.c.

References barrier_node_s::cond, COND_DESTROY, barrier_node_s::lock, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by barrier_queue_Delete(), and barrier_queue_Destroy().

13.24.2.2 barrier_node_p barrier_node_Initialize (int id, int num_registered)

Definition at line 45 of file barrier.c.

References CHECK_NULL, barrier_node_s::cond, COND_INIT, COND_T, barrier_node_s::id, barrier_node_s::lock, MUTEX_INIT, MUTEX_T, node, barrier_node_s::num_registered, and barrier_node_s::num_waiting.

Referenced by MC_BarrierInit().

13.24.2.3 int barrier_queue_Add (barrier_queue_p list, barrier_node_p node)

Definition at line 87 of file barrier.c.

References DATA, barrier_node_s::id, barrier_queue_s::list, ListAdd(), list_s::listhead, barrier_queue_s::lock, MC_SUCCESS, MC_WARN_DUPLICATE, listNode_s::next, listNode_s::node_data, RWLOCK_WRLock, RWLOCK_WRunLock, and barrier_queue_s::size.

Referenced by MC_BarrierInit().

13.24.2.4 int barrier_queue_Delete (int id, barrier_queue_p list)

Definition at line 111 of file barrier.c.

References barrier_node_Destroy(), barrier_node_s::id, barrier_queue_s::list, ListDelete(), ListSearch(), barrier_queue_s::lock, MC_ERR_NOT_FOUND, MC_SUCCESS, RWLOCK_WRLock, RWLOCK_WRunLock, barrier_queue_s::size, and list_s::size.

Referenced by MC_BarrierDelete().

13.24.2.5 int barrier_queue_Destroy (barrier_queue_p queue)

Definition at line 131 of file barrier.c.

References barrier_node_Destroy(), barrier_queue_Pop(), barrier_queue_s::list, ListTerminate(), barrier_queue_s::lock, MC_SUCCESS, node, and RWLOCK_DESTROY.

Referenced by mc_platform_Destroy().

13.24.2.6 barrier_node_p barrier_queue_Get (barrier_queue_p list, int id)

Definition at line 145 of file barrier.c.

References `barrier_queue_s::list`, `list_s::listhead`, `barrier_queue_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_RDLOCK`, and `RWLOCK_RDUNLOCK`.

Referenced by `MC_Barrier()`, and `MC_BarrierInit()`.

13.24.2.7 `barrier_queue_p barrier_queue_New (void)`

Definition at line 162 of file `barrier.c`.

References `CHECK_NULL`, `barrier_queue_s::list`, `ListInitialize()`, `barrier_queue_s::lock`, `RWLOCK_INIT`, and `RWLOCK_T`.

Referenced by `mc_platform_Initialize()`.

13.24.2.8 `barrier_node_p barrier_queue_Pop (barrier_queue_p queue)`

Definition at line 176 of file `barrier.c`.

References `barrier_queue_s::list`, `ListPop()`, and `node`.

Referenced by `barrier_queue_Destroy()`.

13.25 /home/dko/Projects/mobilec/trunk/src/include/cmd_prompt.h File Reference

```
#include "macros.h"
```

Data Structures

- struct [cmd_prompt_s](#)
- struct [command_s](#)

Defines

- #define [CMDLINE_SIZE](#) 80

Typedefs

- typedef struct [cmd_prompt_s](#) [cmd_prompt_t](#)
- typedef [cmd_prompt_t](#) * [cmd_prompt_p](#)
- typedef struct [command_s](#) [command_t](#)
- typedef [command_t](#) * [command_p](#)

Functions

- [int cmd_prompt_Destroy](#) ([cmd_prompt_p](#) cmd_prompt)
- [cmd_prompt_p cmd_prompt_Initialize](#) (struct [mc_platform_s](#) *[mc_platform](#))
- [void cmd_prompt_Start](#) (struct [mc_platform_s](#) *[mc_platform](#))
- [int split_string](#) (char ***args, const char *buf)
- [int process_command](#) ([command_t](#) *cmd)
- [void * cmd_prompt_Thread](#) (void *arg)
- [int exec_command](#) ([command_t](#) cmd, struct [mc_platform_s](#) *global)
- [int dealloc_command](#) ([command_t](#) *cmd)

13.25.1 Define Documentation

13.25.1.1 #define CMDLINE_SIZE 80

Definition at line 37 of file cmd_prompt.h.

13.25.2 Typedef Documentation

13.25.2.1 typedef cmd_prompt_t* cmd_prompt_p

Definition at line 44 of file cmd_prompt.h.

13.25.2.2 `typedef struct cmd_prompt_s cmd_prompt_t`

13.25.2.3 `typedef command_t* command_p`

Definition at line 51 of file `cmd_prompt.h`.

13.25.2.4 `typedef struct command_s command_t`

13.25.3 Function Documentation

13.25.3.1 `int cmd_prompt_Destroy (cmd_prompt_p cmd_prompt)`

Definition at line 129 of file `cmd_prompt.c`.

References `MC_SUCCESS`.

Referenced by `mc_platform_Destroy()`.

13.25.3.2 `cmd_prompt_p cmd_prompt_Initialize (struct mc_platform_s * mc_platform)`

13.25.3.3 `void cmd_prompt_Start (struct mc_platform_s * mc_platform)`

13.25.3.4 `void* cmd_prompt_Thread (void * arg)`

Definition at line 168 of file `cmd_prompt.c`.

References `command_s::args`, `buf`, `dealloc_command()`, `exec_command()`, `command_s::index`, `mc_platform`, `command_s::num_args`, `process_command()`, and `split_string()`.

Referenced by `cmd_prompt_Start()`.

13.25.3.5 `int dealloc_command (command_t * cmd)`

Definition at line 308 of file `cmd_prompt.c`.

References `command_s::args`, and `command_s::num_args`.

Referenced by `cmd_prompt_Thread()`.

13.25.3.6 `int exec_command (command_t cmd, struct mc_platform_s * global)`

13.25.3.7 `int process_command (command_t * cmd)`

Definition at line 274 of file `cmd_prompt.c`.

References `command_s::args`, `command_cmds`, `command_s::index`, and `command_s::num_args`.

Referenced by `cmd_prompt_Thread()`.

13.25.3.8 `int split_string (char *** args, const char * buf)`

Definition at line 226 of file `cmd_prompt.c`.

References `int`.

Referenced by cmd_prompt_Thread().

13.26 /home/dko/Projects/mobilec/trunk/src/include/commands.h File Reference

```
#include "macros.h"
#include "mc_platform.h"
#include "commands.x.h"
```

Defines

- #define [COMMAND](#)(name, cmd, desc) [int](#) handler_##name (void *arg, [mc_platform_p](#) global);
- #define [COMMAND](#)(name, cmd, description) [COMMAND_##name](#) ,
- #define [COMMAND](#)(name, cmd, description) &handler_##name ,
- #define [COMMAND](#)(name, cmd, description) description ,
- #define [COMMAND](#)(name, cmd, description) cmd ,

Typedefs

- typedef [int](#)(* [cmd_handler_t](#))(void *, [mc_platform_p](#) global)

Enumerations

- enum [command_indices_e](#)

Functions

- [COMMAND](#) (QUIT,"quit","This command ends the MobileC application.") [COMMAND](#)(HELP
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList [COMMAND](#) (PRINTLIST_MESSAGE,"printlist_message","This command prints all the list entries on the linked list:\n \ message_queue") [COMMAND](#)(PRINTLIST_AGENTS
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [COMMAND](#) (FLUSH_AGENTS,"flush_agents","This command flushes all of the agents on the system which are still\n\ on the agent list.\n") [COMMAND](#)(COMPOSE_SEND

Variables

- [cmd_handler_t](#) cmd_handlers []
- char * [command_descriptions](#) []
- char * [command_cmds](#) []

13.26.1 Define Documentation

13.26.1.1 #define [COMMAND](#)(name, cmd, description) cmd ,

Definition at line 48 of file commands.h.

13.26.1.2 #define COMMAND(name, cmd, description) description ,

Definition at line 48 of file commands.h.

13.26.1.3 #define COMMAND(name, cmd, description) &handler_##name ,

Definition at line 48 of file commands.h.

13.26.1.4 #define COMMAND(name, cmd, description) COMMAND_##name ,

Definition at line 48 of file commands.h.

13.26.1.5 #define COMMAND(name, cmd, desc) int handler_##name (void *arg, mc_platform_p global);

Definition at line 48 of file commands.h.

13.26.2 Typedef Documentation**13.26.2.1 typedef int(* cmd_handler_t)(void *, mc_platform_p global)**

Definition at line 45 of file commands.h.

13.26.3 Enumeration Type Documentation**13.26.3.1 enum command_indices_e**

Definition at line 47 of file commands.h.

13.26.4 Function Documentation

13.26.4.1 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n command_indices_e::COMMAND (FLUSH_AGENTS, "flush_agents", "This command flushes all of the agents on the system which are still\n\ on the agent list.\n")

Type Constraints

13.26.4.2 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList command_indices_e::COMMAND (PRINTLIST_MESSAGE, "printlist_message", "This command prints all the list entries on the linked list:\n \ message_queue")

Type Constraints

13.26.4.3 `command_indices_e::COMMAND (QUIT, "quit", "This command ends the MobileC application.")`

13.26.5 Variable Documentation

13.26.5.1 `cmd_handler_t cmd_handlers[]`

Definition at line 54 of file `commands.h`.

Referenced by `exec_command()`.

13.26.5.2 `char* command_cmds[]`

Definition at line 68 of file `commands.h`.

Referenced by `handler_HELP()`, and `process_command()`.

13.26.5.3 `char* command_descriptions[]`

Definition at line 61 of file `commands.h`.

Referenced by `handler_COMPOSE_SEND()`, `handler_HELP()`, and `handler_SEND()`.

13.27 /home/dko/Projects/mobilec/trunk/src/include/commands.x.h File Reference

Functions

- [COMMAND](#) (QUIT,"quit","This command ends the MobileC application.") [COMMAND](#)(HELP
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList [COMMAND](#) (PRINTLIST_MESSAGE,"printlist_message","This command prints all the list entries on the linked list:\n \ message_queue") [COMMAND](#)(PRINTLIST_AGENTS
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [COMMAND](#) (FLUSH_AGENTS,"flush_agents","This command flushes all of the agents on the system which are still\n\ on the agent list.\n") [COMMAND](#)(COMPOSE_SEND

Variables

- [help](#)
- This command displays [help](#) for any command Usage is
- This command displays [help](#) for any command Usage [send](#)
- This command displays [help](#) for any command Usage This command sends a client to a host Usage [print_connectlist](#)
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList [printlist_agents](#)
- This command displays [help](#) for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [compose_send](#)

13.27.1 Function Documentation

- 13.27.1.1 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n [COMMAND](#) (FLUSH_AGENTS, "flush_agents", "This command flushes all of the agents on the system which are still\n\ on the agent list.\n")**

Type Constraints

- 13.27.1.2 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList [COMMAND](#) (PRINTLIST_MESSAGE, "printlist_message", "This command prints all the list entries on the linked list:\n \ message_queue")**

Type Constraints

13.27.1.3 COMMAND (QUIT, "quit", "This command ends the MobileC application.")

13.27.2 Variable Documentation

13.27.2.1 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList This command prints all the agents on the system which are still n on the agent list n compose_send

Definition at line 62 of file commands.x.h.

13.27.2.2 help

Definition at line 40 of file commands.x.h.

13.27.2.3 This command displays help for any command Usage This command sends a client to a host Usage is

Definition at line 40 of file commands.x.h.

13.27.2.4 This command displays help for any command Usage This command sends a client to a host Usage print_connectlist

Definition at line 40 of file commands.x.h.

13.27.2.5 This command displays help for any command Usage This command sends a client to a host Usage This command prints the list entries in the connectList printlist_agents

Definition at line 55 of file commands.x.h.

13.27.2.6 This command displays help for any command Usage send

Definition at line 40 of file commands.x.h.

Referenced by extract_nonce_from_MA(), initiate_migration_process(), message_send_Thread(), mtp_http_InitializeFromConnection(), reply_migration_process(), and send_AES_en_MA().

13.28 /home/dko/Projects/mobilec/trunk/src/include/connection.h File Reference

```
#include <sys/types.h>
#include <netinet/in.h>
```

Data Structures

- struct [connection_s](#)

Typedefs

- typedef struct [connection_s](#) [connection_t](#)
- typedef [connection_t](#) * [connection_p](#)

Functions

- [connection_p](#) [connection_New](#) (void)
- int [connection_Destroy](#) ([connection_p](#) connection)
- [connection_p](#) [connection_Copy](#) ([connection_p](#) connection)
- void [connection_Close](#) ([connection_p](#) connection)
- void [connection_SetHosts](#) ([connection_p](#) connection, char *remote_host, char *local_host)
- int [connection_CreateMessage](#) ([connection_p](#) connection)
- int [connection_SendtoConnection](#) ([connection_p](#) connection, char *send_buffer)
- int [connection_RecvFromConnection](#) ([connection_p](#) connection, struct [mc_platform_s](#) *global)

13.28.1 Typedef Documentation

13.28.1.1 typedef [connection_t](#)* [connection_p](#)

Definition at line 66 of file [connection.h](#).

13.28.1.2 typedef struct [connection_s](#) [connection_t](#)

13.28.2 Function Documentation

13.28.2.1 void [connection_Close](#) ([connection_p](#) *connection*)

13.28.2.2 [connection_p](#) [connection_Copy](#) ([connection_p](#) *connection*)

Definition at line 80 of file [connection.c](#).

References [connection_s::addr](#), [connection_s::clientfd](#), [connection_s::connect_id](#), [connection_New\(\)](#), [connection_s::remote_hostname](#), and [connection_s::serverfd](#).

13.28.2.3 int connection_CreateMessage (connection_p *connection*)**13.28.2.4 int connection_Destroy (connection_p *connection*)**

Definition at line 48 of file connection.c.

References connection_s::clientfd, MC_SUCCESS, and connection_s::remote_hostname.

Referenced by acc_connection_Thread().

13.28.2.5 connection_p connection_New (void)

Definition at line 69 of file connection.c.

Referenced by connection_Copy(), and listen_Thread().

13.28.2.6 int connection_RecvFromConnection (connection_p *connection*, struct mc_platform_s * *global*)**13.28.2.7 int connection_SendtoConnection (connection_p *connection*, char * *send_buffer*)****13.28.2.8 void connection_SetHosts (connection_p *connection*, char * *remote_host*, char * *local_host*)**

13.29 /home/dko/Projects/mobilec/trunk/src/include/data_structures.h File Reference

```
#include "ap_queue_template.h"
#include "../mc_list/list.h"
#include "connection.h"
#include "fipa_acl.h"
#include "agent_mailbox.h"
#include "message.h"
#include "interpreter_variable_data.h"
```

Functions

- [AP_QUEUE_DECL_TEMPLATE](#) (connection_queue, connection) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(connection_queue
- [connection_t](#) [int](#) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#) (connection_queue, [Remove](#), [int](#), [int](#)) [int](#) [connection_queue_Print](#)(connection_queue_p clist)
- [AP_QUEUE_DECL_TEMPLATE](#) (message_queue, message) [int](#) [message_queue_Print](#)(message_queue_p queue)
- [AP_QUEUE_DECL_TEMPLATE](#) (agent_queue, agent) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(agent_queue
- struct [agent_s](#) [int](#) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#) (agent_queue, [SearchName](#), struct [agent_s](#) *, char *) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(agent_queue
- struct [agent_s](#) [int](#) [int](#) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#) (agent_queue, [RemoveName](#), [int](#), char *) [int](#) [agent_queue_Print](#)(agent_queue_p queue)
- [AP_QUEUE_DECL_TEMPLATE](#) (agent_variable_list, interpreter_variable_data) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(agent_variable_list
- [interpreter_variable_data_t](#) char * [AP_QUEUE_GENERIC_DECL_TEMPLATE](#) (agent_variable_list, [Remove](#), [int](#), char *) [AP_QUEUE_DECL_TEMPLATE](#)(mail_queue
- [interpreter_variable_data_t](#) char *fipa_acl_message [AP_QUEUE_GENERIC_DECL_TEMPLATE](#) (mail_queue, [SearchReceivers](#), [fipa_acl_message_p](#), char *) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(mail_queue
- [interpreter_variable_data_t](#) char *fipa_acl_message char * [AP_QUEUE_DECL_TEMPLATE](#) (mailbox_queue, agent_mailbox) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(mailbox_queue
- [interpreter_variable_data_t](#) char *fipa_acl_message char char * [AP_QUEUE_GENERIC_DECL_TEMPLATE](#) (mailbox_queue, [SearchID](#), [agent_mailbox_p](#), [int](#)) [AP_QUEUE_GENERIC_DECL_TEMPLATE](#)(mailbox_queue

Variables

- [Search](#)
- struct [agent_s](#) [int](#) [Remove](#)
- struct [agent_s](#) [int](#) [int](#)
- [interpreter_variable_data_t](#) char *fipa_acl_message [RemoveName](#)
- [interpreter_variable_data_t](#) char *fipa_acl_message char * [SearchName](#)
- [interpreter_variable_data_t](#) char *fipa_acl_message char [agent_mailbox_p](#)
- [interpreter_variable_data_t](#) char *fipa_acl_message char char * [RemoveID](#)

13.29.1 Function Documentation

13.29.1.1 `interpreter_variable_data_t char* fipa_acl_message char*`
`AP_QUEUE_DECL_TEMPLATE (mailbox_queue, agent_mailbox)`

Type Constraints

13.29.1.2 `AP_QUEUE_DECL_TEMPLATE (agent_variable_list, interpreter_variable_data)`

13.29.1.3 `AP_QUEUE_DECL_TEMPLATE (agent_queue, agent)`

13.29.1.4 `AP_QUEUE_DECL_TEMPLATE (message_queue, message)`

13.29.1.5 `AP_QUEUE_DECL_TEMPLATE (connection_queue, connection)`

13.29.1.6 `interpreter_variable_data_t char* fipa_acl_message char char*`
`AP_QUEUE_GENERIC_DECL_TEMPLATE (mailbox_queue, SearchID,`
`agent_mailbox_p, int)`

Type Constraints

13.29.1.7 `interpreter_variable_data_t char* fipa_acl_message AP_QUEUE_GENERIC-`
`DECL_TEMPLATE (mail_queue, SearchReceivers, fipa_acl_message_p, char`
`*)`

13.29.1.8 `interpreter_variable_data_t char* AP_QUEUE_GENERIC_DECL_TEMPLATE`
`(agent_variable_list, Remove, int, char *)`

Type Constraints

13.29.1.9 `struct agent_s int int AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_queue,`
`RemoveName, int, char *)`

Type Constraints

13.29.1.10 `struct agent_s int AP_QUEUE_GENERIC_DECL_TEMPLATE (agent_queue,`
`SearchName, struct agent_s *, char *)` **[read]**

Type Constraints

13.29.1.11 `connection_t` `int` `AP_QUEUE_GENERIC_DECL_TEMPLATE` (`connection_queue`,
`Remove`, `int`, `int`)

Type Constraints

13.29.2 Variable Documentation

13.29.2.1 `interpreter_variable_data_t` `char*` `fipa_acl_message` `char` `agent_mailbox_p`

Definition at line 162 of file `data_structures.h`.

13.29.2.2 `interpreter_variable_data_t` `char` `*fipa_acl_message` `char` `char` `int`

Definition at line 101 of file `data_structures.h`.

Referenced by `aes_en_de()`, `LibMC::MCAgency::ChInitializeOptions()`, `default_callback()`, `main()`, `mpi_lsb()`, `mpi_read_string()`, `net_accept()`, `rece_de_msg()`, `remove_nonce_from_MA()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_verify()`, `send_AES_en_MA()`, `sha4_finish()`, `sha4_update()`, `split_string()`, and `ssl_parse_certificate()`.

13.29.2.3 `struct` `agent_s` `int` `Remove`

Definition at line 101 of file `data_structures.h`.

13.29.2.4 `interpreter_variable_data_t` `char*` `fipa_acl_message` `char` `char*` `RemoveID`

Definition at line 176 of file `data_structures.h`.

13.29.2.5 `interpreter_variable_data_t` `char*` `fipa_acl_message` `RemoveName`

Definition at line 149 of file `data_structures.h`.

13.29.2.6 `Search`

Definition at line 53 of file `data_structures.h`.

13.29.2.7 `interpreter_variable_data_t` `char*` `fipa_acl_message` `char*` `SearchName`

Definition at line 162 of file `data_structures.h`.

13.30 /home/dko/Projects/mobilec/trunk/src/include/df.h File Reference

```
#include "../mc_list/list.h"
#include "../include/macros.h"
#include "df_request.x.h"
```

Defines

- #define [REQUEST](#)(name, string, description) REQUEST_##name,
- #define [REQUEST](#)(name, string, description) [int](#) request_handler_##name (struct [mc_platform_s](#) *global, void* data);

Enumerations

- enum [df_request_list_index_e](#)
- enum [service_types_e](#) {
[ZERO](#), [MISC](#), [INSERT](#), [SOME](#),
[TYPES](#), [HERE](#) }

Functions

- [REQUEST](#) (SUBSCRIBE,"subscribe","Subscribe to a directory facilitator") [REQUEST](#)(REGISTER
- Register services with the directory facilitator [REQUEST](#) (DEREGISTER,"deregister","Deregisters mobile agent services from a directory facilitator.") [REQUEST](#)(SEARCH
- [STRUCT](#) (df_request_list_node, [MUTEX_T](#) *[lock](#);[COND_T](#) *[cond](#);const char *command;void *data;[int](#) data_size;) [STRUCT](#)(df_request_list
- [STRUCT](#) (df_search_results, char **agent_names;char **service_names;[int](#) *agent_ids;[int](#) num_results;) [STRUCT](#)(df_request_search
- [STRUCT](#) (df_node, [MUTEX_T](#) *[lock](#);[int](#) agent_id;char *agent_name;[int](#) num_services;char **service_names;enum [service_types_e](#) service_types;) [STRUCT](#)(df
- [STRUCT](#) (df_deregister, [int](#) agent_id;char *service_name;) [int](#) df_Add(struct df_s *df
- [int](#) df_AddRequest (struct df_s *df, struct df_request_list_node_s *[node](#))
- [int](#) df_Destroy (df_p df)
- df_p df_Initialize (struct [mc_platform_s](#) *[mc_platform](#))
- [int](#) df_ProcessRequest (struct [mc_platform_s](#) *global)
- [int](#) df_SearchForService (df_p df, const char *searchstring, char ***agent_names, char ***service_names, [int](#) **agent_ids, [int](#) *num_entries)
- void df_Start (struct [mc_platform_s](#) *[mc_platform](#))
- [int](#) df_node_Destroy (df_node_p df_node)
- [int](#) df_request_list_node_Destroy (df_request_list_node_p [node](#))
- df_request_list_node_p df_request_list_node_New (void)
- [int](#) df_request_list_Destroy (df_request_list_p df_request_list)
- df_request_list_p df_request_list_New (void)
- df_request_list_node_p df_request_list_Pop (df_request_list_p requests)
- df_request_search_p df_request_search_New (void)
- [int](#) df_request_search_Destroy (df_request_search_p [node](#))
- void * df_Thread (void *arg)

Variables

- `MUTEX_T * lock`
- `COND_T * cond`
- `int size`
- `list_p request_list`
- `char * search_string`
- `df_search_results_p search_results`
- `struct mc_platform_s * mc_platform`
- `list_p service_list`
- `int num_entries`
- `int waiting`
- `MUTEX_T * waiting_lock`
- `COND_T * waiting_cond`
- `THREAD_T thread`
- `struct df_node_s * node`

13.30.1 Define Documentation

13.30.1.1 `#define REQUEST(name, string, description) int request_handler_##name (struct mc_platform_s *global, void* data);`

Definition at line 193 of file df.h.

13.30.1.2 `#define REQUEST(name, string, description) REQUEST_##name,`

Definition at line 193 of file df.h.

13.30.2 Enumeration Type Documentation

13.30.2.1 `enum df_request_list_index_e`

Definition at line 46 of file df.h.

13.30.2.2 `enum service_types_e`

Enumerator:

ZERO
MISC
INSERT
SOME
TYPES
HERE

Definition at line 53 of file df.h.

13.30.3 Function Documentation

13.30.3.1 `int df_AddRequest (struct df_s * df, struct df_request_list_node_s * node)`

Definition at line 66 of file df.c.

References ListAdd(), and SIGNAL.

Referenced by MC_DeregisterService(), MC_RegisterService(), and MC_SearchForService().

13.30.3.2 `int df_Destroy (df_p df)`

Definition at line 83 of file df.c.

References COND_DESTROY, df_node_Destroy(), df_request_list_Destroy(), ListPop(), ListTerminate(), MC_SUCCESS, MUTEX_DESTROY, and MUTEX_LOCK.

Referenced by mc_platform_Destroy().

13.30.3.3 `df_p df_Initialize (struct mc_platform_s * mc_platform)`

13.30.3.4 `int df_node_Destroy (df_node_p df_node)`

Definition at line 412 of file df.c.

References MC_SUCCESS, and MUTEX_LOCK.

Referenced by df_Destroy().

13.30.3.5 `int df_ProcessRequest (struct mc_platform_s * global)`

Definition at line 132 of file df.c.

References mc_platform_s::df, df_request_list_Pop(), MC_ERR_EMPTY, and MC_ERR_INVALID.

Referenced by df_Thread().

13.30.3.6 `int df_request_list_Destroy (df_request_list_p df_request_list)`

Definition at line 321 of file df.c.

References df_request_list_node_Destroy(), ListPop(), ListTerminate(), MC_SUCCESS, and node.

Referenced by df_Destroy().

13.30.3.7 `df_request_list_p df_request_list_New (void)`

Definition at line 343 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, ListInitialize(), MUTEX_INIT, and MUTEX_T.

Referenced by df_Initialize().

13.30.3.8 `int df_request_list_node_Destroy (df_request_list_node_p node)`

Definition at line 292 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by df_request_list_Destroy(), and MC_SearchForService().

13.30.3.9 df_request_list_node_p df_request_list_node_New (void)

Definition at line 303 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, MUTEX_T, and node.

Referenced by MC_DeregisterService(), MC_RegisterService(), and MC_SearchForService().

13.30.3.10 df_request_list_node_p df_request_list_Pop (df_request_list_p requests)

Definition at line 368 of file df.c.

References ListPop(), MUTEX_LOCK, MUTEX_UNLOCK, and node.

Referenced by df_ProcessRequest().

13.30.3.11 int df_request_search_Destroy (df_request_search_p node)

Definition at line 399 of file df.c.

References COND_DESTROY, MC_SUCCESS, and MUTEX_DESTROY.

Referenced by MC_SearchForService().

13.30.3.12 df_request_search_p df_request_search_New (void)

Definition at line 384 of file df.c.

References CHECK_NULL, COND_INIT, COND_T, MUTEX_INIT, MUTEX_T, and search.

Referenced by MC_SearchForService().

13.30.3.13 int df_SearchForService (df_p df, const char * searchstring, char *** agent_names, char *** service_names, int ** agent_ids, int * num_entries)

Definition at line 176 of file df.c.

References MC_ERR_NOT_FOUND, MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, listNode_s::next, and listNode_s::node_data.

Referenced by request_handler_SEARCH().

13.30.3.14 void df_Start (struct mc_platform_s * mc_platform)

13.30.3.15 void* df_Thread (void * arg)

Definition at line 426 of file df.c.

References COND_BROADCAST, COND_WAIT, mc_platform_s::df, df_ProcessRequest(), MC_SUCCESS, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::quit, mc_platform_s::quit_lock, and THREAD_EXIT.

Referenced by df_Start().

13.30.3.16 Register services with the directory facilitator df_request_list_index_e::REQUEST (DEREGISTER, "deregister", "Deregisters mobile agent services from a directory facilitator.")

Type Constraints

13.30.3.17 df_request_list_index_e::REQUEST (SUBSCRIBE, "subscribe", "Subscribe to a directory facilitator")

13.30.3.18 STRUCT (df_deregister, int agent_id;char *service_name;)

13.30.3.19 STRUCT (df_node, MUTEX_T *lock;int agent_id;char *agent_name;int num_services;char **service_names;enum service_types_e service_types;)

13.30.3.20 STRUCT (df_search_results, char **agent_names;char **service_names;int *agent_ids;int num_results;)

13.30.3.21 STRUCT (df_request_list_node, MUTEX_T *lock;COND_T *cond;const char *command;void *data;int data_size;)

13.30.4 Variable Documentation

13.30.4.1 COND_T * cond

Definition at line 74 of file df.h.

13.30.4.2 MUTEX_T * lock

Definition at line 73 of file df.h.

13.30.4.3 struct mc_platform_s* mc_platform

Definition at line 107 of file df.h.

Referenced by acc_connection_Thread(), acc_MessageHandlerThread(), acc_Thread(), agent_RunChScriptThread(), ams_Thread(), cmd_prompt_Thread(), listen_Thread(), MC_Initialize(), mc_platform_Initialize(), MC_WaitAgent(), message_send_Thread(), and udplisten_Thread().

13.30.4.4 struct df_node_s* node

Definition at line 129 of file df.h.

Referenced by add_variable(), agent_queue_Flush(), agent_xml_compose(), agent_xml_compose__agent_code(), agent_xml_compose__agent_data(), agent_xml_compose__create_row_nodes(), agent_xml_compose__data(), agent_xml_compose__gaf_message(), agent_xml_compose__home(), agent_xml_compose__message(), agent_xml_compose__mobile_agent(), agent_xml_compose__name(),

agent_xml_compose__owner(), agent_xml_compose__row(), agent_xml_compose__task(), agent_xml_compose__tasks(), agent_xml_compose__wg_code(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_node_Initialize(), barrier_queue_Destroy(), barrier_queue_Pop(), df_request_list_Destroy(), df_request_list_node_New(), df_request_list_Pop(), fipa_envelope_Compose(), fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__envelope(), fipa_envelope_Compose__intended_receiver(), fipa_envelope_Compose__params(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), fipa_envelope_HandleReceived(), main(), MC_Barrier(), MC_BarrierInit(), MC_SyncInit(), mxml_new(), mxmlNewCustom(), mxmlNewElement(), mxmlNewInteger(), mxmlNewOpaque(), mxmlNewReal(), mxmlNewText(), mxmlNewTextf(), request_handler_DEREGISTER(), scan_file(), syncListNew(), write_element(), xml_get_deep_child(), and xml_new_cdata().

13.30.4.5 int num_entries

Definition at line 113 of file df.h.

13.30.4.6 df_request_list_p request_list

Definition at line 76 of file df.h.

13.30.4.7 df_search_results_p search_results

Definition at line 91 of file df.h.

13.30.4.8 char* search_string

Definition at line 90 of file df.h.

13.30.4.9 list_p service_list

Definition at line 110 of file df.h.

13.30.4.10 int size

Definition at line 75 of file df.h.

Referenced by agent_AddPersistentVariable(), agent_xml_compose__create_row_nodes(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), MC_GetAgentReturnData(), MC_SaveData_chdl(), MC_WaitAgent(), mxml_vsnprintf(), receive_AES_en_MA(), and send_AES_en_MA().

13.30.4.11 THREAD_T thread

Definition at line 117 of file df.h.

13.30.4.12 int waiting

Definition at line 114 of file df.h.

13.30.4.13 COND_T* waiting_cond

Definition at line 116 of file df.h.

13.30.4.14 MUTEX_T* waiting_lock

Definition at line 115 of file df.h.

13.31 /home/dko/Projects/mobilec/trunk/src/include/df_request.x.h File Reference

Functions

- [REQUEST](#) (SUBSCRIBE, "subscribe", "Subscribe to a directory facilitator") [REQUEST](#) (REGISTER, "register", "Registers mobile agent services with a directory facilitator.")
- Register services with the directory facilitator [REQUEST](#) (DEREGISTER, "deregister", "Deregisters mobile agent services from a directory facilitator.") [REQUEST](#) (SEARCH, "search", "Searches for a service in the directory.")

Variables

- [register](#)
- Register services with the directory facilitator [search](#)

13.31.1 Function Documentation

13.31.1.1 Register services with the directory facilitator [REQUEST](#) (DEREGISTER, "deregister", "Deregisters mobile agent services from a directory facilitator.")

Type Constraints

13.31.1.2 [REQUEST](#) (SUBSCRIBE, "subscribe", "Subscribe to a directory facilitator")

13.31.2 Variable Documentation

13.31.2.1 [register](#)

Definition at line 43 of file df_request.x.h.

13.31.2.2 Register services with the directory facilitator [search](#)

Definition at line 54 of file df_request.x.h.

Referenced by [df_request_search_New\(\)](#), [MC_SearchForService\(\)](#), and [request_handler_SEARCH\(\)](#).

13.32 /home/dko/Projects/mobilec/trunk/src/include/dynstring.h File Reference

Data Structures

- struct [dynstring_s](#)

Defines

- #define [COMPOSE_BLOCKSIZE](#) 1024

Typedefs

- typedef struct [dynstring_s](#) [dynstring_t](#)
- typedef [dynstring_t](#) * [dynstring_p](#)

Functions

- [dynstring_t](#) * [dynstring_New](#) (void)
- int [dynstring_Append](#) ([dynstring_t](#) *msg, char *str)
- int [dynstring_Destroy](#) ([dynstring_t](#) *dynstring)

13.32.1 Define Documentation

13.32.1.1 #define COMPOSE_BLOCKSIZE 1024

Definition at line 8 of file [dynstring.h](#).

Referenced by [dynstring_Append\(\)](#), and [dynstring_New\(\)](#).

13.32.2 Typedef Documentation

13.32.2.1 typedef [dynstring_t](#)* [dynstring_p](#)

Definition at line 14 of file [dynstring.h](#).

13.32.2.2 typedef struct [dynstring_s](#) [dynstring_t](#)

13.32.3 Function Documentation

13.32.3.1 int [dynstring_Append](#) ([dynstring_t](#) *msg, char *str)

Definition at line 30 of file [dynstring.c](#).

References [COMPOSE_BLOCKSIZE](#), [dynstring_s::len](#), [dynstring_s::message](#), and [dynstring_s::size](#).

Referenced by [fipa_acl_Compose\(\)](#), [fipa_agent_identifier_Compose\(\)](#), [fipa_agent_identifier_set_-Compose\(\)](#), [fipa_DateTime_Compose\(\)](#), [fipa_envelope_Compose__from\(\)](#), [fipa_number_Compose\(\)](#),

fipa_performative_Compose(), fipa_protocol_Compose(), fipa_string_Compose(), fipa_url_Compose(), fipa_url_sequence_Compose(), fipa_word_Compose(), message_send_Thread(), mtp_http_CreateMessage(), and mtp_http_InitializeFromConnection().

13.32.3.2 int dynstring_Destroy (dynstring_t * *dynstring*)

Definition at line 56 of file dynstring.c.

References dynstring_s::message.

Referenced by fipa_envelope_Compose__from(), MC_AclSend(), message_send_Thread(), mtp_http_CreateMessage(), and mtp_http_InitializeFromConnection().

13.32.3.3 dynstring_t* dynstring_New (void)

Definition at line 14 of file dynstring.c.

References COMPOSE_BLOCKSIZE, dynstring_s::len, dynstring_s::message, and dynstring_s::size.

Referenced by fipa_acl_Compose(), fipa_envelope_Compose__from(), message_send_Thread(), mtp_http_CreateMessage(), and mtp_http_InitializeFromConnection().

13.33 /home/dko/Projects/mobilec/trunk/src/include/fipa_acl.h File Reference

```
#include "dynstring.h"
```

Data Structures

- struct [fipa_acl_message_s](#)
- struct [fipa_message_string_s](#)
- struct [fipa_url_sequence_s](#)
- struct [fipa_agent_identifier_set_s](#)
- struct [fipa_agent_identifier_s](#)
- struct [fipa_expression_s](#)
- union [fipa_expression_s::content_u](#)
- struct [fipa_word_s](#)
- struct [fipa_string_s](#)
- struct [fipa_DateTime_s](#)
- struct [fipa_url_s](#)
- struct [fipa_number_s](#)

Typedefs

- typedef enum [fipa_expression_type_e](#) [fipa_expression_type_t](#)
- typedef struct [fipa_acl_message_s](#) [fipa_acl_message_t](#)
- typedef [fipa_acl_message_t](#) * [fipa_acl_message_p](#)
- typedef struct [fipa_message_string_s](#) [fipa_message_string_t](#)
- typedef [fipa_message_string_t](#) * [fipa_message_string_p](#)
- typedef struct [fipa_url_sequence_s](#) [fipa_url_sequence_t](#)
- typedef [fipa_url_sequence_t](#) * [fipa_url_sequence_p](#)
- typedef struct [fipa_agent_identifier_set_s](#) [fipa_agent_identifier_set_t](#)
- typedef [fipa_agent_identifier_set_t](#) * [fipa_agent_identifier_set_p](#)
- typedef struct [fipa_agent_identifier_s](#) [fipa_agent_identifier_t](#)
- typedef [fipa_agent_identifier_t](#) * [fipa_agent_identifier_p](#)
- typedef struct [fipa_expression_s](#) [fipa_expression_t](#)
- typedef [fipa_expression_t](#) * [fipa_expression_p](#)
- typedef struct [fipa_word_s](#) [fipa_word_t](#)
- typedef [fipa_word_t](#) * [fipa_word_p](#)
- typedef struct [fipa_string_s](#) [fipa_string_t](#)
- typedef [fipa_string_t](#) * [fipa_string_p](#)
- typedef struct [fipa_DateTime_s](#) [fipa_DateTime_t](#)
- typedef [fipa_DateTime_t](#) * [fipa_DateTime_p](#)
- typedef struct [fipa_url_s](#) [fipa_url_t](#)
- typedef [fipa_url_t](#) * [fipa_url_p](#)
- typedef struct [fipa_number_s](#) [fipa_number_t](#)
- typedef [fipa_number_t](#) * [fipa_number_p](#)

Enumerations

- enum `fipa_performative_e` {
`FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,`
`FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,`
`FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,`
`FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,`
`FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,`
`FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_-`
`SUBSCRIBE,`
`FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,`
`FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,`
`FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,`
`FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,`
`FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,`
`FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_-`
`SUBSCRIBE }`
- enum `fipa_protocol_e` {
`FIPA_PROTOCOL_ERROR = -1, FIPA_PROTOCOL_NONE, FIPA_PROTOCOL_REQUEST,`
`FIPA_PROTOCOL_QUERY,`
`FIPA_PROTOCOL_REQUEST_WHEN, FIPA_PROTOCOL_CONTRACT_NET, FIPA_-`
`PROTOCOL_ITERATED_CONTRACT_NET, FIPA_PROTOCOL_ENGLISH_AUCTION,`
`FIPA_PROTOCOL_DUTCH_AUCTION, FIPA_PROTOCOL_BROKERING, FIPA_-`
`PROTOCOL_RECRUITING, FIPA_PROTOCOL_SUBSCRIBE,`
`FIPA_PROTOCOL_PROPOSE, FIPA_PROTOCOL_END }`
- enum `fipa_expression_type_e` {
`FIPA_EXPR_ZERO, FIPA_EXPR_WORD, FIPA_EXPR_STRING, FIPA_EXPR_NUMBER,`
`FIPA_EXPR_DATETIME, FIPA_EXPR_EXPRESSION }`

Functions

- `int fipa_performative_Compose (dynstring_t *msg, enum fipa_performative_e performative)`
- `int fipa_protocol_Compose (dynstring_t *msg, enum fipa_protocol_e protocol)`
- `fipa_acl_message_t * fipa_acl_message_New (void)`
- `int fipa_acl_message_Destroy (fipa_acl_message_t *message)`
- `fipa_acl_message_t * fipa_acl_message_Copy (fipa_acl_message_t *src)`
- `int fipa_acl_Compose (dynstring_t **msg, fipa_acl_message_t *acl)`
- `fipa_message_string_t * fipa_message_string_New (void)`
- `int fipa_message_string_Destroy (fipa_message_string_t *message)`
- `fipa_message_string_t * fipa_message_string_Copy (fipa_message_string_t *src)`
- `fipa_url_sequence_t * fipa_url_sequence_New (void)`
- `int fipa_url_sequence_Destroy (fipa_url_sequence_t *sequence)`
- `fipa_url_sequence_t * fipa_url_sequence_Copy (fipa_url_sequence_t *src)`
- `int fipa_url_sequence_Compose (dynstring_t *msg, fipa_url_sequence_t *urls)`
- `fipa_agent_identifier_set_t * fipa_agent_identifier_set_New (void)`
- `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t *idset)`

- [fipa_agent_identifier_set_t * fipa_agent_identifier_set_Copy \(fipa_agent_identifier_set_t *src\)](#)
- [int fipa_agent_identifier_set_Compose \(dynstring_t *msg, fipa_agent_identifier_set_t *ids\)](#)
- [fipa_agent_identifier_t * fipa_agent_identifier_New \(void\)](#)
- [int fipa_agent_identifier_Destroy \(fipa_agent_identifier_t *id\)](#)
- [fipa_agent_identifier_t * fipa_agent_identifier_Copy \(fipa_agent_identifier_t *src\)](#)
- [int fipa_agent_identifier_Compose \(dynstring_t *msg, fipa_agent_identifier_t *id\)](#)
- [fipa_expression_t * fipa_expression_New \(void\)](#)
- [int fipa_expression_Destroy \(fipa_expression_t *expr\)](#)
- [fipa_expression_t * fipa_expression_Copy \(fipa_expression_t *src\)](#)
- [int fipa_expression_Compose \(dynstring_t *msg, fipa_expression_t *expr\)](#)
- [fipa_word_t * fipa_word_New \(void\)](#)
- [int fipa_word_Destroy \(fipa_word_t *word\)](#)
- [fipa_word_t * fipa_word_Copy \(fipa_word_t *src\)](#)
- [int fipa_word_Compose \(dynstring_t *msg, fipa_word_t *word\)](#)
- [fipa_string_t * fipa_string_New \(void\)](#)
- [int fipa_string_Destroy \(fipa_string_t *str\)](#)
- [fipa_string_t * fipa_string_Copy \(fipa_string_t *src\)](#)
- [int fipa_string_Compose \(dynstring_t *msg, fipa_string_t *string\)](#)
- [fipa_DateTime_t * fipa_DateTime_New \(void\)](#)
- [int fipa_DateTime_Destroy \(fipa_DateTime_t *dt\)](#)
- [fipa_DateTime_t * fipa_DateTime_Copy \(fipa_DateTime_t *src\)](#)
- [int fipa_DateTime_Compose \(dynstring_t *msg, fipa_DateTime_t *date\)](#)
- [fipa_url_t * fipa_url_New \(void\)](#)
- [int fipa_url_Destroy \(fipa_url_t *url\)](#)
- [fipa_url_t * fipa_url_Copy \(fipa_url_t *src\)](#)
- [int fipa_url_Compose \(dynstring_t *msg, fipa_url_t *url\)](#)
- [fipa_number_t * fipa_number_New \(void\)](#)
- [int fipa_number_Destroy \(fipa_number_t *number\)](#)
- [fipa_number_t * fipa_number_Copy \(fipa_number_t *src\)](#)
- [int fipa_number_Compose \(dynstring_t *msg, fipa_number_t *number\)](#)
- [int fipa_acl_Parse \(struct fipa_acl_message_s *acl, fipa_message_string_p message\)](#)
- [int fipa_message_parameter_Parse \(struct fipa_acl_message_s *acl, fipa_message_string_p message\)](#)
- [int fipa_protocol_type_Parse \(enum fipa_protocol_e *protocol, fipa_message_string_p message\)](#)
- [int fipa_message_type_Parse \(enum fipa_performative_e *performative, fipa_message_string_p message\)](#)
- [int fipa_GetAtom \(fipa_message_string_p message, char expected_atom\)](#)
- [int fipa_word_Parse \(fipa_word_t **word, fipa_message_string_p message\)](#)
- [int fipa_CheckNextToken \(const fipa_message_string_p message, const char *token\)](#)
- [int fipa_expression_Parse \(fipa_expression_t **expression, fipa_message_string_p message\)](#)
- [int fipa_GetNextWord \(char **word, const fipa_message_string_p message\)](#)
- [int fipa_GetWholeToken \(char **word, const fipa_message_string_p message\)](#)
- [int fipa_datetime_Parse \(fipa_DateTime_p *datetime, fipa_message_string_p message\)](#)
- [int fipa_string_Parse \(fipa_string_p *fipa_string, fipa_message_string_p message\)](#)
- [int fipa_agent_identifier_Parse \(fipa_agent_identifier_p *aid, fipa_message_string_p message\)](#)
- [int fipa_agent_identifier_set_Parse \(fipa_agent_identifier_set_p *agent_ids, fipa_message_string_p message\)](#)
- [int fipa_url_sequence_Parse \(fipa_url_sequence_p *urls, fipa_message_string_p message\)](#)
- [int fipa_url_Parse \(fipa_url_p *url, fipa_message_string_p message\)](#)
- [struct fipa_acl_message_s * fipa_Reply \(struct fipa_acl_message_s *acl\)](#)

13.33.1 Typedef Documentation

13.33.1.1 typedef fipa_acl_message_t* fipa_acl_message_p

Definition at line 136 of file fipa_acl.h.

13.33.1.2 typedef struct fipa_acl_message_s fipa_acl_message_t

13.33.1.3 typedef fipa_agent_identifier_t* fipa_agent_identifier_p

Definition at line 186 of file fipa_acl.h.

13.33.1.4 typedef fipa_agent_identifier_set_t* fipa_agent_identifier_set_p

Definition at line 172 of file fipa_acl.h.

13.33.1.5 typedef struct fipa_agent_identifier_set_s fipa_agent_identifier_set_t

13.33.1.6 typedef struct fipa_agent_identifier_s fipa_agent_identifier_t

13.33.1.7 typedef fipa_DateTime_t* fipa_DateTime_p

Definition at line 246 of file fipa_acl.h.

13.33.1.8 typedef struct fipa_DateTime_s fipa_DateTime_t

13.33.1.9 typedef fipa_expression_t* fipa_expression_p

Definition at line 205 of file fipa_acl.h.

13.33.1.10 typedef struct fipa_expression_s fipa_expression_t

13.33.1.11 typedef enum fipa_expression_type_e fipa_expression_type_t

13.33.1.12 typedef fipa_message_string_t* fipa_message_string_p

Definition at line 148 of file fipa_acl.h.

13.33.1.13 typedef struct fipa_message_string_s fipa_message_string_t

13.33.1.14 typedef fipa_number_t* fipa_number_p

Definition at line 268 of file fipa_acl.h.

13.33.1.15 typedef struct fipa_number_s fipa_number_t

13.33.1.16 typedef fipa_string_t* fipa_string_p

Definition at line 227 of file fipa_acl.h.

13.33.1.17 `typedef struct fipa_string_s fipa_string_t`

13.33.1.18 `typedef fipa_url_t* fipa_url_p`

Definition at line 257 of file fipa_acl.h.

13.33.1.19 `typedef fipa_url_sequence_t* fipa_url_sequence_p`

Definition at line 159 of file fipa_acl.h.

13.33.1.20 `typedef struct fipa_url_sequence_s fipa_url_sequence_t`

13.33.1.21 `typedef struct fipa_url_s fipa_url_t`

13.33.1.22 `typedef fipa_word_t* fipa_word_p`

Definition at line 216 of file fipa_acl.h.

13.33.1.23 `typedef struct fipa_word_s fipa_word_t`

13.33.2 Enumeration Type Documentation

13.33.2.1 `enum fipa_expression_type_e`

Enumerator:

FIPA_EXPR_ZERO

FIPA_EXPR_WORD

FIPA_EXPR_STRING

FIPA_EXPR_NUMBER

FIPA_EXPR_DATETIME

FIPA_EXPR_EXPRESSION

Definition at line 94 of file fipa_acl.h.

13.33.2.2 `enum fipa_performative_e`

Enumerator:

FIPA_ERROR

FIPA_ZERO

FIPA_ACCEPT_PROPOSAL

FIPA_AGREE

FIPA_CANCEL

FIPA_CALL_FOR_PROPOSAL

FIPA_CONFIRM

FIPA_DISCONFIRM

FIPA_FAILURE
FIPA_INFORM
FIPA_INFORM_IF
FIPA_INFORM_REF
FIPA_NOT_UNDERSTOOD
FIPA_PROPOGATE
FIPA_PROPOSE
FIPA_PROXY
FIPA_QUERY_IF
FIPA_QUERY_REF
FIPA_REFUSE
FIPA_REJECT_PROPOSAL
FIPA_REQUEST
FIPA_REQUEST_WHEN
FIPA_REQUEST_WHENEVER
FIPA_SUBSCRIBE
FIPA_ERROR
FIPA_ZERO
FIPA_ACCEPT_PROPOSAL
FIPA_AGREE
FIPA_CANCEL
FIPA_CALL_FOR_PROPOSAL
FIPA_CONFIRM
FIPA_DISCONFIRM
FIPA_FAILURE
FIPA_INFORM
FIPA_INFORM_IF
FIPA_INFORM_REF
FIPA_NOT_UNDERSTOOD
FIPA_PROPOGATE
FIPA_PROPOSE
FIPA_PROXY
FIPA_QUERY_IF
FIPA_QUERY_REF
FIPA_REFUSE
FIPA_REJECT_PROPOSAL
FIPA_REQUEST
FIPA_REQUEST_WHEN
FIPA_REQUEST_WHENEVER
FIPA_SUBSCRIBE

Definition at line 41 of file fipa_acl.h.

13.33.2.3 enum fipa_protocol_e

Enumerator:

FIPA_PROTOCOL_ERROR
FIPA_PROTOCOL_NONE
FIPA_PROTOCOL_REQUEST
FIPA_PROTOCOL_QUERY
FIPA_PROTOCOL_REQUEST_WHEN
FIPA_PROTOCOL_CONTRACT_NET
FIPA_PROTOCOL_ITERATED_CONTRACT_NET
FIPA_PROTOCOL_ENGLISH_AUCTION
FIPA_PROTOCOL_DUTCH_AUCTION
FIPA_PROTOCOL_BROKERING
FIPA_PROTOCOL_RECRUITING
FIPA_PROTOCOL_SUBSCRIBE
FIPA_PROTOCOL_PROPOSE
FIPA_PROTOCOL_END

Definition at line 72 of file fipa_acl.h.

13.33.3 Function Documentation

13.33.3.1 int fipa_acl_Compose (dynstring_t ** msg, fipa_acl_message_t * acl)

Definition at line 1232 of file fipa_acl.c.

References fipa_acl_message_s::content, fipa_acl_message_s::conversation_id, dynstring_Append(), dynstring_New(), fipa_acl_message_s::encoding, fipa_agent_identifier_Compose(), fipa_agent_identifier_set_Compose(), fipa_DateTime_Compose(), fipa_expression_Compose(), fipa_performative_Compose(), fipa_protocol_Compose(), FIPA_PROTOCOL_NONE, fipa_string_Compose(), fipa_acl_message_s::in_reply_to, fipa_acl_message_s::language, fipa_acl_message_s::ontology, fipa_acl_message_s::performative, fipa_acl_message_s::protocol, fipa_acl_message_s::receiver, fipa_acl_message_s::reply_by, fipa_acl_message_s::reply_to, fipa_acl_message_s::reply_with, and fipa_acl_message_s::sender.

Referenced by MC_AclSend().

13.33.3.2 fipa_acl_message_t* fipa_acl_message_Copy (fipa_acl_message_t * src)

Definition at line 86 of file fipa_acl.c.

References fipa_acl_message_s::content, fipa_acl_message_s::conversation_id, fipa_acl_message_s::encoding, fipa_acl_message_New(), fipa_agent_identifier_Copy(), fipa_agent_identifier_set_Copy(), fipa_DateTime_Copy(), fipa_expression_Copy(), fipa_string_Copy(), fipa_acl_message_s::in_reply_to, fipa_acl_message_s::language, fipa_acl_message_s::ontology, fipa_acl_message_s::performative, fipa_acl_message_s::protocol, fipa_acl_message_s::receiver, fipa_acl_message_s::reply_by, fipa_acl_message_s::reply_to, fipa_acl_message_s::reply_with, and fipa_acl_message_s::sender.

Referenced by MC_AclSend().

13.33.3.3 int fipa_acl_message_Destroy (fipa_acl_message_t * message)

Definition at line 65 of file fipa_acl.c.

References fipa_acl_message_s::content, fipa_acl_message_s::conversation_id, fipa_acl_message_s::encoding, fipa_agent_identifier_Destroy(), fipa_agent_identifier_set_Destroy(), fipa_DateTime_Destroy(), fipa_expression_Destroy(), fipa_string_Destroy(), fipa_acl_message_s::in_reply_to, fipa_acl_message_s::language, fipa_acl_message_s::ontology, fipa_acl_message_s::receiver, fipa_acl_message_s::reply_by, fipa_acl_message_s::reply_to, fipa_acl_message_s::reply_with, and fipa_acl_message_s::sender.

Referenced by acc_connection_Thread(), and MC_AclDestroy().

13.33.3.4 fipa_acl_message_t* fipa_acl_message_New (void)

Definition at line 57 of file fipa_acl.c.

Referenced by acc_connection_Thread(), fipa_acl_message_Copy(), fipa_Reply(), and MC_AclNew().

13.33.3.5 int fipa_acl_Parse (struct fipa_acl_message_s * acl, fipa_message_string_p message)**13.33.3.6 int fipa_agent_identifier_Compose (dynstring_t * msg, fipa_agent_identifier_t * id)**

Definition at line 1446 of file fipa_acl.c.

References fipa_agent_identifier_s::addresses, dynstring_Append(), fipa_agent_identifier_set_Compose(), fipa_url_sequence_Compose(), fipa_agent_identifier_s::name, fipa_agent_identifier_set_s::num, fipa_url_sequence_s::num, and fipa_agent_identifier_s::resolvers.

Referenced by fipa_acl_Compose(), and fipa_agent_identifier_set_Compose().

13.33.3.7 fipa_agent_identifier_t* fipa_agent_identifier_Copy (fipa_agent_identifier_t * src)

Definition at line 235 of file fipa_acl.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_New(), fipa_agent_identifier_set_Copy(), fipa_url_sequence_Copy(), fipa_agent_identifier_s::name, and fipa_agent_identifier_s::resolvers.

Referenced by fipa_acl_message_Copy(), fipa_acl_Param_Copy(), fipa_agent_identifier_set_Copy(), and fipa_Reply().

13.33.3.8 int fipa_agent_identifier_Destroy (fipa_agent_identifier_t * id)

Definition at line 223 of file fipa_acl.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_set_Destroy(), fipa_url_sequence_Destroy(), fipa_agent_identifier_s::name, and fipa_agent_identifier_s::resolvers.

Referenced by fipa_acl_message_Destroy(), fipa_acl_Param_Destroy(), fipa_agent_identifier_set_Destroy(), and MC_AclSetSender().

13.33.3.9 fipa_agent_identifier_t* fipa_agent_identifier_New (void)

Definition at line 215 of file fipa_acl.c.

Referenced by `fipa_agent_identifier_Copy()`, `fipa_envelope_ParseAgentIdentifier()`, `MC_AclAddReceiver()`, `MC_AclAddReplyTo()`, and `MC_AclSetSender()`.

13.33.3.10 `int fipa_agent_identifier_Parse (fipa_agent_identifier_p * aid, fipa_message_string_p message)`

Definition at line 1032 of file `fipa_acl.c`.

References `CHECK_NULL`, `fipa_word_s::content`, `fipa_agent_identifier_set_Parse()`, `fipa_GetAtom()`, `fipa_url_sequence_Parse()`, `fipa_word_Destroy()`, `fipa_word_Parse()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_set_Parse()`, and `fipa_message_parameter_Parse()`.

13.33.3.11 `int fipa_agent_identifier_set_Compose (dynstring_t * msg, fipa_agent_identifier_set_t * ids)`

Definition at line 1433 of file `fipa_acl.c`.

References `dynstring_Append()`, `fipa_agent_identifier_Compose()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, and `fipa_agent_identifier_set_s::num`.

Referenced by `fipa_acl_Compose()`, and `fipa_agent_identifier_Compose()`.

13.33.3.12 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_Copy (fipa_agent_identifier_set_t * src)`

Definition at line 194 of file `fipa_acl.c`.

References `fipa_agent_identifier_Copy()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_agent_identifier_set_s::num`, and `fipa_agent_identifier_set_s::retain_order`.

Referenced by `fipa_acl_message_Copy()`, `fipa_acl_Param_Copy()`, `fipa_agent_identifier_Copy()`, and `fipa_Reply()`.

13.33.3.13 `int fipa_agent_identifier_set_Destroy (fipa_agent_identifier_set_t * idset)`

Definition at line 182 of file `fipa_acl.c`.

References `fipa_agent_identifier_Destroy()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, and `fipa_agent_identifier_set_s::num`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_acl_Param_Destroy()`, and `fipa_agent_identifier_Destroy()`.

13.33.3.14 `fipa_agent_identifier_set_t* fipa_agent_identifier_set_New (void)`

Definition at line 174 of file `fipa_acl.c`.

Referenced by `fipa_agent_identifier_set_Copy()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, `fipa_Reply()`, `MC_AclAddReceiver()`, and `MC_AclAddReplyTo()`.

13.33.3.15 int fipa_agent_identifier_set_Parse (fipa_agent_identifier_set_p * agent_ids, fipa_message_string_p message)

Definition at line 1191 of file fipa_acl.c.

References fipa_word_s::content, fipa_agent_identifier_Parse(), fipa_GetAtom(), fipa_word_Parse(), MC_ERR_PARSE, MC_SUCCESS, and fipa_agent_identifier_set_s::retain_order.

Referenced by fipa_agent_identifier_Parse(), and fipa_message_parameter_Parse().

13.33.3.16 int fipa_CheckNextToken (const fipa_message_string_p message, const char * token)

Definition at line 694 of file fipa_acl.c.

References fipa_message_string_s::parse.

Referenced by fipa_expression_Parse().

13.33.3.17 int fipa_DateTime_Compose (dynstring_t * msg, fipa_DateTime_t * date)

Definition at line 1519 of file fipa_acl.c.

References buf, fipa_DateTime_s::day, dynstring_Append(), fipa_DateTime_s::hour, fipa_DateTime_s::millisecond, fipa_DateTime_s::minute, fipa_DateTime_s::month, fipa_DateTime_s::second, fipa_DateTime_s::sign, and fipa_DateTime_s::year.

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

13.33.3.18 fipa_DateTime_t* fipa_DateTime_Copy (fipa_DateTime_t * src)

Definition at line 398 of file fipa_acl.c.

References fipa_DateTime_New().

Referenced by fipa_acl_envelope_Received_Copy(), fipa_acl_message_Copy(), fipa_acl_Param_Copy(), and fipa_expression_Copy().

13.33.3.19 int fipa_DateTime_Destroy (fipa_DateTime_t * dt)

Definition at line 391 of file fipa_acl.c.

Referenced by fipa_acl_envelope_Received_Destroy(), fipa_acl_message_Destroy(), fipa_acl_Param_Destroy(), and fipa_expression_Destroy().

13.33.3.20 fipa_DateTime_t* fipa_DateTime_New (void)

Definition at line 383 of file fipa_acl.c.

Referenced by fipa_DateTime_Copy().

13.33.3.21 int fipa_datetime_Parse (fipa_DateTime_p * datetime, fipa_message_string_p message)

Definition at line 849 of file fipa_acl.c.

References `buf`, `fipa_GetWholeToken()`, `MC_ERR_PARSE`, `MC_SUCCESS`, `fipa_message_string_s::parse`, and `fipa_DateTime_s::sign`.

Referenced by `fipa_envelope_HandleDate()`, `fipa_envelope_HandleReceived()`, `fipa_expression_Parse()`, and `fipa_message_parameter_Parse()`.

13.33.3.22 `int fipa_expression_Compose (dynstring_t * msg, fipa_expression_t * expr)`

Definition at line 1472 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Compose()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Compose()`, `fipa_number_Compose()`, `fipa_string_Compose()`, `fipa_word_Compose()`, `MC_ERR_PARSE`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_Compose()`, and `fipa_expression_Compose()`.

13.33.3.23 `fipa_expression_t* fipa_expression_Copy (fipa_expression_t * src)`

Definition at line 286 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Copy()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Copy()`, `fipa_expression_Destroy()`, `fipa_expression_New()`, `fipa_number_Copy()`, `fipa_string_Copy()`, `fipa_word_Copy()`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_message_Copy()`, `fipa_expression_Copy()`, and `fipa_Reply()`.

13.33.3.24 `int fipa_expression_Destroy (fipa_expression_t * expr)`

Definition at line 255 of file `fipa_acl.c`.

References `fipa_expression_s::content`, `fipa_expression_s::content_u::datetime`, `fipa_expression_s::content_u::expression`, `fipa_DateTime_Destroy()`, `FIPA_EXPR_DATETIME`, `FIPA_EXPR_EXPRESSION`, `FIPA_EXPR_NUMBER`, `FIPA_EXPR_STRING`, `FIPA_EXPR_WORD`, `fipa_expression_Destroy()`, `fipa_number_Destroy()`, `fipa_string_Destroy()`, `fipa_word_Destroy()`, `FREEMEM`, `fipa_expression_s::content_u::number`, `fipa_expression_s::content_u::string`, `fipa_expression_s::type`, and `fipa_expression_s::content_u::word`.

Referenced by `fipa_acl_message_Destroy()`, `fipa_expression_Copy()`, and `fipa_expression_Destroy()`.

13.33.3.25 `fipa_expression_t* fipa_expression_New (void)`

Definition at line 247 of file `fipa_acl.c`.

Referenced by `fipa_expression_Copy()`, and `MC_AclSetConversationID()`.

13.33.3.26 int fipa_expression_Parse (fipa_expression_t ** *expression*, fipa_message_string_p *message*)

Definition at line 713 of file fipa_acl.c.

References fipa_CheckNextToken(), fipa_datetime_Parse(), FIPA_EXPR_DATETIME, FIPA_EXPR_EXPRESSION, FIPA_EXPR_STRING, FIPA_EXPR_WORD, fipa_expression_Parse(), fipa_GetAtom(), fipa_string_Parse(), fipa_word_Parse(), MC_ERR_PARSE, MC_SUCCESS, and fipa_expression_s::type.

Referenced by fipa_expression_Parse(), and fipa_message_parameter_Parse().

13.33.3.27 int fipa_GetAtom (fipa_message_string_p *message*, char *expected_atom*)

Definition at line 629 of file fipa_acl.c.

References MC_ERR_PARSE, MC_SUCCESS, and fipa_message_string_s::parse.

Referenced by fipa_acl_Parse(), fipa_agent_identifier_Parse(), fipa_agent_identifier_set_Parse(), fipa_expression_Parse(), fipa_message_parameter_Parse(), fipa_string_Parse(), and fipa_url_sequence_Parse().

13.33.3.28 int fipa_GetNextWord (char ** *word*, const fipa_message_string_p *message*)

Definition at line 764 of file fipa_acl.c.

References ERR, MC_SUCCESS, and fipa_message_string_s::parse.

13.33.3.29 int fipa_GetWholeToken (char ** *word*, const fipa_message_string_p *message*)

Definition at line 812 of file fipa_acl.c.

References MC_SUCCESS, and fipa_message_string_s::parse.

Referenced by fipa_datetime_Parse().

13.33.3.30 int fipa_message_parameter_Parse (struct fipa_acl_message_s * *acl*, fipa_message_string_p *message*)**13.33.3.31 fipa_message_string_t* fipa_message_string_Copy (fipa_message_string_t * *src*)**

Definition at line 128 of file fipa_acl.c.

References fipa_message_string_s::message, and fipa_message_string_s::parse.

13.33.3.32 int fipa_message_string_Destroy (fipa_message_string_t * *message*)

Definition at line 118 of file fipa_acl.c.

References fipa_message_string_s::message.

Referenced by acc_connection_Thread().

13.33.3.33 fipa_message_string_t* fipa_message_string_New (void)

Definition at line 110 of file fipa_acl.c.

Referenced by `acc_connection_Thread()`.

13.33.3.34 `int fipa_message_type_Parse (enum fipa_performative_e * performative, fipa_message_string_p message)`

Definition at line 567 of file `fipa_acl.c`.

References `fipa_word_s::content`, `FIPA_ACCEPT_PROPOSAL`, `FIPA_AGREE`, `FIPA_CALL_FOR_PROPOSAL`, `FIPA_CANCEL`, `FIPA_CONFIRM`, `FIPA_DISCONFIRM`, `FIPA_FAILURE`, `FIPA_INFORM`, `FIPA_INFORM_IF`, `FIPA_INFORM_REF`, `FIPA_NOT_UNDERSTOOD`, `FIPA_PROPOGATE`, `FIPA_PROPOSE`, `FIPA_PROXY`, `FIPA_QUERY_IF`, `FIPA_QUERY_REF`, `FIPA_REFUSE`, `FIPA_REJECT_PROPOSAL`, `FIPA_REQUEST`, `FIPA_REQUEST_WHEN`, `FIPA_REQUEST_WHENEVER`, `FIPA_SUBSCRIBE`, `fipa_word_Destroy()`, `fipa_word_Parse()`, and `MC_ERR_PARSE`.

Referenced by `fipa_acl_Parse()`.

13.33.3.35 `int fipa_number_Compose (dynstring_t * msg, fipa_number_t * number)`

Definition at line 1547 of file `fipa_acl.c`.

References `dynstring_Append()`, and `fipa_number_s::str`.

Referenced by `fipa_expression_Compose()`.

13.33.3.36 `fipa_number_t* fipa_number_Copy (fipa_number_t * src)`

Definition at line 454 of file `fipa_acl.c`.

References `fipa_number_New()`, and `fipa_number_s::str`.

Referenced by `fipa_expression_Copy()`.

13.33.3.37 `int fipa_number_Destroy (fipa_number_t * number)`

Definition at line 444 of file `fipa_acl.c`.

References `fipa_number_s::str`.

Referenced by `fipa_expression_Destroy()`.

13.33.3.38 `fipa_number_t* fipa_number_New (void)`

Definition at line 436 of file `fipa_acl.c`.

Referenced by `fipa_number_Copy()`.

13.33.3.39 `int fipa_performative_Compose (dynstring_t * msg, enum fipa_performative_e performative)`

Definition at line 1345 of file `fipa_acl.c`.

References `dynstring_Append()`, `FIPA_ACCEPT_PROPOSAL`, `FIPA_AGREE`, `FIPA_CALL_FOR_PROPOSAL`, `FIPA_CANCEL`, `FIPA_CONFIRM`, `FIPA_DISCONFIRM`, `FIPA_FAILURE`,

FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF, FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY, FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL, FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_SUBSCRIBE, and MC_ERR_PARSE.

Referenced by fipa_acl_Compose().

13.33.3.40 int fipa_protocol_Compose (dynstring_t * *msg*, enum fipa_protocol_e *protocol*)

Definition at line 1302 of file fipa_acl.c.

References dynstring_Append(), FIPA_PROTOCOL_BROKERING, FIPA_PROTOCOL_CONTRACT_NET, FIPA_PROTOCOL_DUTCH_AUCTION, FIPA_PROTOCOL_ENGLISH_AUCTION, FIPA_PROTOCOL_ITERATED_CONTRACT_NET, FIPA_PROTOCOL_PROPOSE, FIPA_PROTOCOL_QUERY, FIPA_PROTOCOL_RECRUITING, FIPA_PROTOCOL_REQUEST, FIPA_PROTOCOL_REQUEST_WHEN, FIPA_PROTOCOL_SUBSCRIBE, and MC_ERR_PARSE.

Referenced by fipa_acl_Compose().

13.33.3.41 int fipa_protocol_type_Parse (enum fipa_protocol_e * *protocol*, fipa_message_string_p *message*)

Definition at line 527 of file fipa_acl.c.

References fipa_word_s::content, FIPA_PROTOCOL_BROKERING, FIPA_PROTOCOL_CONTRACT_NET, FIPA_PROTOCOL_DUTCH_AUCTION, FIPA_PROTOCOL_ENGLISH_AUCTION, FIPA_PROTOCOL_ITERATED_CONTRACT_NET, FIPA_PROTOCOL_PROPOSE, FIPA_PROTOCOL_QUERY, FIPA_PROTOCOL_RECRUITING, FIPA_PROTOCOL_REQUEST, FIPA_PROTOCOL_REQUEST_WHEN, FIPA_PROTOCOL_SUBSCRIBE, fipa_word_Destroy(), fipa_word_Parse(), and MC_ERR_PARSE.

Referenced by fipa_message_parameter_Parse().

13.33.3.42 struct fipa_acl_message_s* fipa_Reply (struct fipa_acl_message_s * *acl*) [read]

Definition at line 1555 of file fipa_acl.c.

References fipa_acl_message_s::conversation_id, fipa_acl_message_New(), fipa_agent_identifier_Copy(), fipa_agent_identifier_set_Copy(), fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_expression_Copy(), fipa_agent_identifier_set_s::num, fipa_acl_message_s::protocol, fipa_acl_message_s::receiver, fipa_acl_message_s::reply_to, fipa_agent_identifier_set_s::retain_order, and fipa_acl_message_s::sender.

Referenced by MC_AclReply().

13.33.3.43 int fipa_string_Compose (dynstring_t * *msg*, fipa_string_t * *string*)

Definition at line 1510 of file fipa_acl.c.

References fipa_string_s::content, and dynstring_Append().

Referenced by fipa_acl_Compose(), and fipa_expression_Compose().

13.33.3.44 fipa_string_t* fipa_string_Copy (fipa_string_t * *src*)

Definition at line 373 of file fipa_acl.c.

References fipa_string_s::content, and fipa_string_New().

Referenced by fipa_acl_message_Copy(), and fipa_expression_Copy().

13.33.3.45 int fipa_string_Destroy (fipa_string_t * *str*)

Definition at line 363 of file fipa_acl.c.

References fipa_string_s::content.

Referenced by fipa_acl_message_Destroy(), fipa_expression_Destroy(), and MC_AclSetContent().

13.33.3.46 fipa_string_t* fipa_string_New (void)

Definition at line 355 of file fipa_acl.c.

Referenced by fipa_string_Copy(), MC_AclSetContent(), and MC_AclSetConversationID().

13.33.3.47 int fipa_string_Parse (fipa_string_p * *fipa_string*, fipa_message_string_p *message*)

Definition at line 987 of file fipa_acl.c.

References fipa_string_s::content, fipa_GetAtom(), MC_ERR_PARSE, MC_SUCCESS, and fipa_message_string_s::parse.

Referenced by fipa_expression_Parse(), and fipa_message_parameter_Parse().

13.33.3.48 int fipa_url_Compose (dynstring_t * *msg*, fipa_url_t * *url*)

Definition at line 1539 of file fipa_acl.c.

References dynstring_Append(), and fipa_url_s::str.

Referenced by fipa_url_sequence_Compose().

13.33.3.49 fipa_url_t* fipa_url_Copy (fipa_url_t * *src*)

Definition at line 426 of file fipa_acl.c.

References fipa_url_New(), and fipa_url_s::str.

Referenced by fipa_acl_envelope_Received_Copy(), and fipa_url_sequence_Copy().

13.33.3.50 int fipa_url_Destroy (fipa_url_t * *url*)

Definition at line 416 of file fipa_acl.c.

References fipa_url_s::str.

Referenced by fipa_acl_envelope_Received_Destroy(), and fipa_url_sequence_Destroy().

13.33.3.51 fipa_url_t* fipa_url_New (void)

Definition at line 408 of file fipa_acl.c.

Referenced by fipa_envelope_HandleReceived(), fipa_envelope_ParseAddresses(), fipa_url_Copy(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

13.33.3.52 int fipa_url_Parse (fipa_url_p * url, fipa_message_string_p message)

Definition at line 1171 of file fipa_acl.c.

References fipa_word_s::content, fipa_word_Destroy(), and fipa_word_Parse().

Referenced by fipa_url_sequence_Parse().

13.33.3.53 int fipa_url_sequence_Compose (dynstring_t * msg, fipa_url_sequence_t * urls)

Definition at line 1420 of file fipa_acl.c.

References dynstring_Append(), fipa_url_Compose(), fipa_url_sequence_s::num, and fipa_url_sequence_s::urls.

Referenced by fipa_agent_identifier_Compose().

13.33.3.54 fipa_url_sequence_t* fipa_url_sequence_Copy (fipa_url_sequence_t * src)

Definition at line 158 of file fipa_acl.c.

References fipa_url_Copy(), fipa_url_sequence_New(), fipa_url_sequence_s::num, and fipa_url_sequence_s::urls.

Referenced by fipa_agent_identifier_Copy().

13.33.3.55 int fipa_url_sequence_Destroy (fipa_url_sequence_t * sequence)

Definition at line 146 of file fipa_acl.c.

References fipa_url_Destroy(), fipa_url_sequence_s::num, and fipa_url_sequence_s::urls.

Referenced by fipa_agent_identifier_Destroy().

13.33.3.56 fipa_url_sequence_t* fipa_url_sequence_New (void)

Definition at line 138 of file fipa_acl.c.

Referenced by fipa_url_sequence_Copy(), fipa_url_sequence_Parse(), MC_AclAddReceiver(), MC_AclAddReplyTo(), and MC_AclSetSender().

13.33.3.57 int fipa_url_sequence_Parse (fipa_url_sequence_p * urls, fipa_message_string_p message)

Definition at line 1139 of file fipa_acl.c.

References fipa_word_s::content, fipa_GetAtom(), fipa_url_Parse(), fipa_url_sequence_New(), fipa_word_Destroy(), fipa_word_Parse(), and MC_ERR_PARSE.

Referenced by `fipa_agent_identifier_Parse()`.

13.33.3.58 `int fipa_word_Compose (dynstring_t * msg, fipa_word_t * word)`

Definition at line 1502 of file `fipa_acl.c`.

References `fipa_word_s::content`, and `dynstring_Append()`.

Referenced by `fipa_expression_Compose()`.

13.33.3.59 `fipa_word_t* fipa_word_Copy (fipa_word_t * src)`

Definition at line 345 of file `fipa_acl.c`.

References `fipa_word_s::content`, and `fipa_word_New()`.

Referenced by `fipa_expression_Copy()`.

13.33.3.60 `int fipa_word_Destroy (fipa_word_t * word)`

Definition at line 335 of file `fipa_acl.c`.

References `fipa_word_s::content`.

Referenced by `fipa_agent_identifier_Parse()`, `fipa_expression_Destroy()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `fipa_protocol_type_Parse()`, `fipa_url_Parse()`, and `fipa_url_sequence_Parse()`.

13.33.3.61 `fipa_word_t* fipa_word_New (void)`

Definition at line 327 of file `fipa_acl.c`.

Referenced by `fipa_word_Copy()`.

13.33.3.62 `int fipa_word_Parse (fipa_word_t ** word, fipa_message_string_p message)`

Definition at line 652 of file `fipa_acl.c`.

References `CHECK_NULL`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `fipa_message_string_s::parse`.

Referenced by `fipa_agent_identifier_Parse()`, `fipa_agent_identifier_set_Parse()`, `fipa_expression_Parse()`, `fipa_message_parameter_Parse()`, `fipa_message_type_Parse()`, `fipa_protocol_type_Parse()`, `fipa_url_Parse()`, and `fipa_url_sequence_Parse()`.

13.34 /home/dko/Projects/mobilec/trunk/src/include/fipa_acl_envelope.h File Reference

```
#include "fipa_acl.h"
```

Data Structures

- struct [fipa_acl_envelope_Received_s](#)
- struct [fipa_acl_Param_s](#)
- struct [fipa_acl_envelope_s](#)

Typedefs

- typedef struct [fipa_acl_envelope_Received_s](#) [fipa_acl_envelope_Received_t](#)
- typedef struct [fipa_acl_Param_s](#) [fipa_acl_Param_t](#)
- typedef struct [fipa_acl_envelope_s](#) [fipa_acl_envelope_t](#)
- typedef [fipa_acl_envelope_t](#) * [fipa_acl_envelope_p](#)

Functions

- [fipa_acl_envelope_Received_t](#) * [fipa_acl_envelope_Received_New](#) (void)
- int [fipa_acl_envelope_Received_Destroy](#) ([fipa_acl_envelope_Received_t](#) *received)
- [fipa_acl_envelope_Received_t](#) * [fipa_acl_envelope_Received_Copy](#) ([fipa_acl_envelope_Received_t](#) *received)
- [fipa_acl_Param_t](#) * [fipa_acl_Param_New](#) (void)
- int [fipa_acl_Param_Destroy](#) ([fipa_acl_Param_t](#) *param)
- [fipa_acl_Param_t](#) * [fipa_acl_Param_Copy](#) ([fipa_acl_Param_t](#) *param)
- [fipa_acl_envelope_t](#) * [fipa_acl_envelope_New](#) (void)
- int [fipa_acl_envelope_Destroy](#) ([fipa_acl_envelope_t](#) *envelope)
- [fipa_acl_envelope_t](#) * [fipa_acl_envelope_Copy](#) ([fipa_acl_envelope_t](#) *envelope)
- int [fipa_envelope_Parse](#) (struct [fipa_acl_envelope_s](#) *envelope, const char *message)
- int [fipa_envelope_HandleEnvelope](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *node)
- int [fipa_envelope_HandleParams](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *node)
- int [fipa_envelope_HandleTo](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandleFrom](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandleComments](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandleAclRepresentation](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandlePayloadLength](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandlePayloadEncoding](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandleDate](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)
- int [fipa_envelope_HandleIntendedReceiver](#) (struct [fipa_acl_envelope_s](#) *envelope, [mxml_node_t](#) *param_node, int cur_param)

- `int fipa_envelope_HandleReceived` (struct `fipa_acl_envelope_s` *envelope, `mxml_node_t` *param_node, `int` cur_param)
- `int fipa_envelope_ParseAgentIdentifier` (struct `fipa_agent_identifier_s` **aid, `mxml_node_t` *agent_identifier_node)
- `int fipa_envelope_ParseAddresses` (struct `fipa_agent_identifier_s` *aid, `mxml_node_t` *addresses_node)
- `int fipa_envelope_ParseResolvers` (struct `fipa_agent_identifier_s` *aid, `mxml_node_t` *resolvers_node)
- `char * fipa_envelope_Compose` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__envelope` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__params` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__to` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__from` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__acl_representation` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__payload_encoding` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__date` (`fipa_acl_message_t` *fipa_acl)
- `mxml_node_t * fipa_envelope_Compose__intended_receiver` (`fipa_acl_message_t` *fipa_acl)

13.34.1 Typedef Documentation

13.34.1.1 `typedef fipa_acl_envelope_t* fipa_acl_envelope_p`

Definition at line 45 of file `fipa_acl_envelope.h`.

13.34.1.2 `typedef struct fipa_acl_envelope_Received_s fipa_acl_envelope_Received_t`

13.34.1.3 `typedef struct fipa_acl_envelope_s fipa_acl_envelope_t`

13.34.1.4 `typedef struct fipa_acl_Param_s fipa_acl_Param_t`

13.34.2 Function Documentation

13.34.2.1 `fipa_acl_envelope_t* fipa_acl_envelope_Copy (fipa_acl_envelope_t * envelope)`

Definition at line 150 of file `fipa_envelope.c`.

References `fipa_acl_envelope_New()`, `fipa_acl_Param_Copy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

13.34.2.2 `int fipa_acl_envelope_Destroy (fipa_acl_envelope_t * envelope)`

Definition at line 137 of file `fipa_envelope.c`.

References `fipa_acl_Param_Destroy()`, `fipa_acl_envelope_s::num_params`, and `fipa_acl_envelope_s::params`.

Referenced by `acc_connection_Thread()`.

13.34.2.3 fipa_acl_envelope_t* fipa_acl_envelope_New (void)

Definition at line 129 of file fipa_envelope.c.

Referenced by acc_connection_Thread(), and fipa_acl_envelope_Copy().

13.34.2.4 fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_Copy (fipa_acl_envelope_Received_t * *received*)

Definition at line 69 of file fipa_envelope.c.

References fipa_acl_envelope_Received_New(), fipa_DateTime_Copy(), fipa_url_Copy(), fipa_acl_envelope_Received_s::received_by, fipa_acl_envelope_Received_s::received_date, fipa_acl_envelope_Received_s::received_from, fipa_acl_envelope_Received_s::received_id, and fipa_acl_envelope_Received_s::received_via.

Referenced by fipa_acl_Param_Copy().

13.34.2.5 int fipa_acl_envelope_Received_Destroy (fipa_acl_envelope_Received_t * *received*)

Definition at line 57 of file fipa_envelope.c.

References fipa_DateTime_Destroy(), fipa_url_Destroy(), fipa_acl_envelope_Received_s::received_by, fipa_acl_envelope_Received_s::received_date, fipa_acl_envelope_Received_s::received_from, fipa_acl_envelope_Received_s::received_id, and fipa_acl_envelope_Received_s::received_via.

Referenced by fipa_acl_Param_Destroy().

13.34.2.6 fipa_acl_envelope_Received_t* fipa_acl_envelope_Received_New (void)

Definition at line 48 of file fipa_envelope.c.

Referenced by fipa_acl_envelope_Received_Copy(), and fipa_envelope_HandleReceived().

13.34.2.7 fipa_acl_Param_t* fipa_acl_Param_Copy (fipa_acl_Param_t * *param*)

Definition at line 110 of file fipa_envelope.c.

References fipa_acl_Param_s::acl_representation, fipa_acl_Param_s::comments, fipa_acl_Param_s::date, fipa_acl_envelope_Received_Copy(), fipa_acl_Param_New(), fipa_agent_identifier_Copy(), fipa_agent_identifier_set_Copy(), fipa_DateTime_Copy(), fipa_acl_Param_s::from, fipa_acl_Param_s::intended_receiver, fipa_acl_Param_s::payload_encoding, fipa_acl_Param_s::payload_length, fipa_acl_Param_s::received, and fipa_acl_Param_s::to.

Referenced by fipa_acl_envelope_Copy().

13.34.2.8 int fipa_acl_Param_Destroy (fipa_acl_Param_t * *param*)

Definition at line 93 of file fipa_envelope.c.

References fipa_acl_Param_s::acl_representation, fipa_acl_Param_s::comments, fipa_acl_Param_s::date, fipa_acl_envelope_Received_Destroy(), fipa_agent_identifier_Destroy(), fipa_agent_identifier_set_Destroy(), fipa_DateTime_Destroy(), fipa_acl_Param_s::from, fipa_acl_Param_s::intended_receiver, fipa_acl_Param_s::payload_encoding, fipa_acl_Param_s::payload_length, fipa_acl_Param_s::received, and fipa_acl_Param_s::to.

Referenced by `fipa_acl_envelope_Destroy()`.

13.34.2.9 `fipa_acl_Param_t* fipa_acl_Param_New (void)`

Definition at line 85 of file `fipa_envelope.c`.

Referenced by `fipa_acl_Param_Copy()`, and `fipa_envelope_HandleTo()`.

13.34.2.10 `char* fipa_envelope_Compose (fipa_acl_message_t * fipa_acl)`

Definition at line 869 of file `fipa_envelope.c`.

References `fipa_envelope_Compose__envelope()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmAdd()`, `mxmDelete()`, `mxmLoadString()`, `mxmSaveAllocString()`, and `node`.

Referenced by `MC_AclSend()`.

13.34.2.11 `mxml_node_t* fipa_envelope_Compose__acl_representation (fipa_acl_message_t * fipa_acl)`

Definition at line 1089 of file `fipa_envelope.c`.

References `mxmNewElement()`, `mxmNewText()`, and `node`.

Referenced by `fipa_envelope_Compose__params()`.

13.34.2.12 `mxml_node_t* fipa_envelope_Compose__date (fipa_acl_message_t * fipa_acl)`

Definition at line 1117 of file `fipa_envelope.c`.

References `buf`, `mxmNewElement()`, `mxmNewText()`, and `node`.

Referenced by `fipa_envelope_Compose__params()`.

13.34.2.13 `mxml_node_t* fipa_envelope_Compose__envelope (fipa_acl_message_t * fipa_acl)`

Definition at line 894 of file `fipa_envelope.c`.

References `fipa_envelope_Compose__params()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `mxmAdd()`, `mxmNewElement()`, and `node`.

Referenced by `fipa_envelope_Compose()`.

13.34.2.14 `mxml_node_t* fipa_envelope_Compose__from (fipa_acl_message_t * fipa_acl)`

Definition at line 1021 of file `fipa_envelope.c`.

References `fipa_agent_identifier_s::addresses`, `buf`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `g_mc_platform`, `mc_platform_s::hostname`, `dynstring_s::message`, `mxmNewElement()`, `mxmNewText()`, `fipa_agent_identifier_s::name`, `fipa_url_sequence_s::num`, `mc_platform_s::port`, `fipa_acl_message_s::sender`, `fipa_url_s::str`, and `fipa_url_sequence_s::urls`.

Referenced by `fipa_envelope_Compose__params()`.

13.34.2.15 mxml_node_t* fipa_envelope_Compose__intended_receiver (fipa_acl_message_t * *fipa_acl*)

Definition at line 1146 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_set_s::fipa_agent_identifiers, mxml-NewElement(), mxmlNewText(), fipa_agent_identifier_s::name, node, fipa_url_sequence_s::num, fipa_acl_message_s::receiver, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by fipa_envelope_Compose__params().

13.34.2.16 mxml_node_t* fipa_envelope_Compose__params (fipa_acl_message_t * *fipa_acl*)

Definition at line 914 of file fipa_envelope.c.

References fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), fipa_envelope_Compose__intended_receiver(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), MXML_ADD_AFTER, MXML_ADD_TO_PARENT, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), and node.

Referenced by fipa_envelope_Compose__envelope().

13.34.2.17 mxml_node_t* fipa_envelope_Compose__payload_encoding (fipa_acl_message_t * *fipa_acl*)

Definition at line 1103 of file fipa_envelope.c.

References mxmlNewElement(), mxmlNewText(), and node.

Referenced by fipa_envelope_Compose__params().

13.34.2.18 mxml_node_t* fipa_envelope_Compose__to (fipa_acl_message_t * *fipa_acl*)

Definition at line 975 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_set_s::fipa_agent_identifiers, mxml-NewElement(), mxmlNewText(), fipa_agent_identifier_s::name, node, fipa_url_sequence_s::num, fipa_agent_identifier_set_s::num, fipa_acl_message_s::receiver, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by fipa_envelope_Compose__params().

13.34.2.19 int fipa_envelope_HandleAclRepresentation (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 381 of file fipa_envelope.c.

References fipa_acl_Param_s::acl_representation, mxml_node_s::child, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.34.2.20 **int fipa_envelope_HandleComments** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 357 of file fipa_envelope.c.

References mxml_node_s::child, fipa_acl_Param_s::comments, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.34.2.21 **int fipa_envelope_HandleDate** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 452 of file fipa_envelope.c.

References mxml_node_s::child, fipa_acl_Param_s::date, fipa_datetime_Parse(), MC_ERR_PARSE, fipa_message_string_s::message, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, fipa_message_string_s::parse, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.34.2.22 **int fipa_envelope_HandleEnvelope** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *node*)

Definition at line 178 of file fipa_envelope.c.

References fipa_envelope_HandleParams(), MC_ERR_PARSE, MXML_DESCEND_FIRST, and mxmlFindElement().

Referenced by fipa_envelope_Parse().

13.34.2.23 **int fipa_envelope_HandleFrom** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 328 of file fipa_envelope.c.

References fipa_envelope_ParseAgentIdentifier(), fipa_acl_Param_s::from, MC_ERR_PARSE, MXML_DESCEND_FIRST, mxmlFindElement(), and fipa_acl_envelope_s::params.

Referenced by fipa_envelope_HandleParams().

13.34.2.24 **int fipa_envelope_HandleIntendedReceiver** (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 482 of file fipa_envelope.c.

References fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_ParseAgentIdentifier(), fipa_acl_Param_s::intended_receiver, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlFindElement(), fipa_agent_identifier_set_s::num, fipa_acl_envelope_s::params, and fipa_agent_identifier_set_s::retain_order.

Referenced by fipa_envelope_HandleParams().

13.34.2.25 int fipa_envelope_HandleParams (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *node*)

Definition at line 198 of file fipa_envelope.c.

References buf, fipa_envelope_HandleAclRepresentation(), fipa_envelope_HandleComments(), fipa_envelope_HandleDate(), fipa_envelope_HandleFrom(), fipa_envelope_HandleIntendedReceiver(), fipa_envelope_HandlePayloadEncoding(), fipa_envelope_HandlePayloadLength(), fipa_envelope_HandleReceived(), fipa_envelope_HandleTo(), MC_ERR_PARSE, MXML_DESCEND_FIRST, mxmlFindElement(), fipa_acl_envelope_s::num_params, and fipa_acl_envelope_s::params.

Referenced by fipa_envelope_HandleEnvelope().

13.34.2.26 int fipa_envelope_HandlePayloadEncoding (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 428 of file fipa_envelope.c.

References mxml_node_s::child, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, fipa_acl_Param_s::payload_encoding, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.34.2.27 int fipa_envelope_HandlePayloadLength (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 405 of file fipa_envelope.c.

References mxml_node_s::child, MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), fipa_acl_envelope_s::params, fipa_acl_Param_s::payload_length, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleParams().

13.34.2.28 int fipa_envelope_HandleReceived (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 560 of file fipa_envelope.c.

References fipa_acl_envelope_Received_New(), fipa_datetime_Parse(), fipa_url_New(), MC_ERR_PARSE, fipa_message_string_s::message, MXML_DESCEND_FIRST, mxmlElementGetAttr(), mxmlFindElement(), node, fipa_acl_envelope_s::params, fipa_message_string_s::parse, fipa_acl_Param_s::received, fipa_acl_envelope_Received_s::received_by, fipa_acl_envelope_Received_s::received_date, fipa_acl_envelope_Received_s::received_from, fipa_acl_envelope_Received_s::received_id, fipa_acl_envelope_Received_s::received_via, and fipa_url_s::str.

Referenced by fipa_envelope_HandleParams().

13.34.2.29 int fipa_envelope_HandleTo (struct fipa_acl_envelope_s * *envelope*, mxml_node_t * *param_node*, int *cur_param*)

Definition at line 250 of file fipa_envelope.c.

References fipa_acl_Param_New(), fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_ParseAgentIdentifier(), MC_ERR_PARSE, MXML_DESCEND_FIRST,

MXML_NO_DESCEND, mxmlFindElement(), fipa_agent_identifier_set_s::num, fipa_acl_envelope_s::params, fipa_agent_identifier_set_s::retain_order, and fipa_acl_Param_s::to.

Referenced by fipa_envelope_HandleParams().

13.34.2.30 **int fipa_envelope_Parse (struct fipa_acl_envelope_s * *envelope*, const char * *message*)**

Definition at line 165 of file fipa_envelope.c.

References fipa_envelope_HandleEnvelope(), MXML_NO_CALLBACK, mxmlDelete(), and mxmlLoadString().

Referenced by acc_connection_Thread().

13.34.2.31 **int fipa_envelope_ParseAddresses (struct fipa_agent_identifier_s * *aid*, mxml_node_t * *addresses_node*)**

Definition at line 737 of file fipa_envelope.c.

References fipa_agent_identifier_s::addresses, mxml_node_s::child, fipa_url_New(), MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_NO_DESCEND, MXML_TEXT, mxmlFindElement(), fipa_url_sequence_s::num, fipa_url_s::str, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, fipa_url_sequence_s::urls, and mxml_node_s::value.

Referenced by fipa_envelope_ParseAgentIdentifier().

13.34.2.32 **int fipa_envelope_ParseAgentIdentifier (struct fipa_agent_identifier_s ** *aid*, mxml_node_t * *agent_identifier_node*)**

Definition at line 687 of file fipa_envelope.c.

References mxml_node_s::child, fipa_agent_identifier_New(), fipa_envelope_ParseAddresses(), fipa_envelope_ParseResolvers(), MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_TEXT, mxmlFindElement(), mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by fipa_envelope_HandleFrom(), fipa_envelope_HandleIntendedReceiver(), fipa_envelope_HandleTo(), and fipa_envelope_ParseResolvers().

13.34.2.33 **int fipa_envelope_ParseResolvers (struct fipa_agent_identifier_s * *aid*, mxml_node_t * *resolvers_node*)**

Definition at line 804 of file fipa_envelope.c.

References fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_ParseAgentIdentifier(), MC_ERR_PARSE, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlFindElement(), fipa_agent_identifier_set_s::num, fipa_agent_identifier_s::resolvers, and fipa_agent_identifier_set_s::retain_order.

Referenced by fipa_envelope_ParseAgentIdentifier().

13.35 /home/dko/Projects/mobilec/trunk/src/include/fipa_comm.h File Reference

```
#include <fipa_acl.h>
```

Data Structures

- struct [fipa_comm_message_check_t](#)
- struct [fipa_comm_reply_t](#)
- struct [fipa_comm_action_t](#)
- struct [fipa_comm_performative_t](#)
- struct [fipa_list_t](#)
- struct [fipa_comm_protocol_cn_t](#)
- struct [fipa_comm_protocol_t](#)
- struct [fipa_comm_t](#)

Defines

- #define [true](#) 1
- #define [false](#) 0
- #define [MC_NUM_PERFORMATIVES](#) 23

Typedefs

- typedef [int](#)(* [fipa_comm_callback](#))(char *received_content, char **reply_content)
- typedef [int](#)(* [protocol_contract_net_callback](#))(int numBids, char **bids, char *winners)
- typedef struct [fipa_comm_message_check_t](#) [fipa_comm_message_check_s](#)
- typedef struct [fipa_comm_message_check_t](#) * [fipa_comm_message_check_p](#)
- typedef struct [fipa_comm_reply_t](#) [fipa_comm_reply_s](#)
- typedef struct [fipa_comm_reply_t](#) * [fipa_comm_reply_p](#)
- typedef struct [fipa_comm_action_t](#) [fipa_comm_action_s](#)
- typedef struct [fipa_comm_action_t](#) * [fipa_comm_action_p](#)
- typedef struct [fipa_comm_performative_t](#) [fipa_comm_performative_s](#)
- typedef struct [fipa_comm_performative_t](#) * [fipa_comm_performative_p](#)
- typedef struct [fipa_list_t](#) [fipa_list_s](#)
- typedef struct [fipa_list_t](#) * [fipa_list_p](#)
- typedef struct [fipa_comm_protocol_cn_t](#) [fipa_comm_protocol_cn_s](#)
- typedef struct [fipa_comm_protocol_cn_t](#) * [fipa_comm_protocol_cn_p](#)
- typedef struct [fipa_comm_protocol_t](#) * [fipa_comm_protocol_p](#)
- typedef struct [fipa_comm_protocol_t](#) [fipa_comm_protocol_s](#)
- typedef struct [fipa_comm_t](#) [fipa_comm_s](#)
- typedef struct [fipa_comm_t](#) * [fipa_comm_p](#)
- typedef struct [fipa_comm_t](#) [MCFIPAComm_s](#)
- typedef struct [fipa_comm_t](#) * [MCFIPAComm_p](#)

Enumerations

- enum `fipa_comm_type_e` {
`FIPA_COMM_ACTION`, `FIPA_COMM_CHECK`, `FIPA_COMM_REPLY`, `FIPA_COMM_CALLBACK`,
`FIPA_COMM_END` }
- enum `fipa_comm_logic_type_e` { `FIPA_COMM_LOGIC_ERROR` = -1, `FIPA_COMM_LOGIC_OR`, `FIPA_COMM_LOGIC_AND` }
- enum `fipa_comm_check_type_e` {
`FIPA_COMM_CHECK_ERROR` = -1, `FIPA_COMM_CHECK_SENDER_NAME`, `FIPA_COMM_CHECK_SENDER_ADDRESS`, `FIPA_COMM_CHECK_CONTENT`,
`FIPA_COMM_CHECK_CONVERSATIONID`, `FIPA_COMM_CHECK_LANGUAGE`, `FIPA_COMM_CHECK_ONTOLOGY`, `FIPA_COMM_CHECK_ENCODING` }
- enum `fipa_comm_action_type_e` {
`FIPA_COMM_ACTION_ERROR` = -1, `FIPA_COMM_ACTION_BOTH`, `FIPA_COMM_ACTION_REPLY`, `FIPA_COMM_ACTION_CALLBACK`,
`FIPA_COMM_ACTION_NONE` }
- enum `fipa_comm_protocol_cn_state_e` {
`FIPA_COMM_PROTOCOL_CN_ERROR` = -1, `FIPA_COMM_PROTOCOL_CN_START`, `FIPA_COMM_PROTOCOL_CN_CFP`, `FIPA_COMM_PROTOCOL_CN_BID`,
`FIPA_COMM_PROTOCOL_CN_RESULT`, `FIPA_COMM_PROTOCOL_CN_DONE` }

Functions

- `fipa_comm_p mc_FIPAComm_New ()`
- `int mc_FIPAComm_HandleMessage (fipa_comm_p fcomm, fipa_acl_message_p acl)`
- `int mc_FIPAComm_HandleMessageOnce (fipa_comm_p fcomm)`
- `int mc_FIPAComm_HandleMessageWait (fipa_comm_p fcomm)`
- `void mc_FIPAComm_Destroy (fipa_comm_p)`
- `int mc_FIPAComm_RegisterPerformative (fipa_comm_p fcomm, enum fipa_performative_e type)`
- `int mc_FIPAComm_DeregisterPerformative (fipa_comm_p fcomm, enum fipa_performative_e type)`
- `int mc_FIPAComm_SetDefaultReply (fipa_comm_p fcomm, enum fipa_performative_e type, char *content)`
- `int mc_FIPAComm_SetPerformativeDefaultReply (fipa_comm_p fcomm, enum fipa_performative_e ptype, enum fipa_performative_e rtype, char *content)`
- `int mc_FIPAComm_SetAgent (fipa_comm_p fcomm, void *agent, char *agent_name, char *agent_address)`
- `int mc_FIPAComm_RegisterAction (fipa_comm_p fcomm, enum fipa_performative_e pType, enum fipa_comm_action_type_e aType,...)`
- `int mc_FIPAComm_RegisterActionS (fipa_performative_e type, fipa_comm_action_p act)`
- `fipa_comm_protocol_p mc_FIPAComm_New ()`
- `int mc_FIPAComm_RegisterProtocolCallback (fipa_comm_p fcomm, enum fipa_protocol_e protocolType, fipa_comm_callback func)`
- `int mc_FIPAComm_RegisterProtocol (fipa_comm_p fcomm, enum fipa_protocol_e protocolType, char *protocolID, void *info, char *network)`
- `void fipa_comm_protocol_info_COPY (fipa_comm_protocol_p proto, void *info)`
- `int fipa_list_Add (fipa_list_p *list, fipa_acl_message_p acl)`
- `int mc_FIPAComm_StartProtocol (fipa_comm_p fcomm, enum fipa_protocol_e type, char *id)`
- `int mc_FIPAComm_HandleProtocol (fipa_comm_p fcomm, fipa_acl_message_p acl)`

- [fipa_comm_protocol_p mc_FIPAComm_AddNewProtocol](#) ([fipa_comm_p](#) fcomm, enum [fipa_protocol_e](#) type, char *id)
- [int mc_FIPAComm_FSM_CN](#) ([fipa_comm_p](#) fcomm, [fipa_acl_message_p](#) acl)
- [fipa_comm_protocol_p mc_FIPAComm_NewProtocol](#) (enum [fipa_protocol_e](#) type, char *id)
- [int mc_FIPAComm_SetProtocolState](#) ([fipa_comm_protocol_p](#) protocol, char state)

13.35.1 Define Documentation

13.35.1.1 #define false 0

Definition at line 11 of file [fipa_comm.h](#).

13.35.1.2 #define MC_NUM_PERFORMATIVES 23

Definition at line 14 of file [fipa_comm.h](#).

13.35.1.3 #define true 1

Definition at line 7 of file [fipa_comm.h](#).

13.35.2 Typedef Documentation

13.35.2.1 typedef struct fipa_comm_action_t* fipa_comm_action_p

Definition at line 109 of file [fipa_comm.h](#).

13.35.2.2 typedef struct fipa_comm_action_t fipa_comm_action_s

Definition at line 108 of file [fipa_comm.h](#).

13.35.2.3 typedef int(* fipa_comm_callback)(char *received_content, char **reply_content)

Definition at line 17 of file [fipa_comm.h](#).

13.35.2.4 typedef struct fipa_comm_message_check_t* fipa_comm_message_check_p

Definition at line 80 of file [fipa_comm.h](#).

13.35.2.5 typedef struct fipa_comm_message_check_t fipa_comm_message_check_s

Definition at line 79 of file [fipa_comm.h](#).

13.35.2.6 typedef struct fipa_comm_t* fipa_comm_p

Definition at line 193 of file [fipa_comm.h](#).

13.35.2.7 typedef struct fipa_comm_performative_t* fipa_comm_performative_p

Definition at line 121 of file fipa_comm.h.

13.35.2.8 typedef struct fipa_comm_performative_t fipa_comm_performative_s

Definition at line 120 of file fipa_comm.h.

13.35.2.9 typedef struct fipa_comm_protocol_cn_t* fipa_comm_protocol_cn_p

Definition at line 156 of file fipa_comm.h.

13.35.2.10 typedef struct fipa_comm_protocol_cn_t fipa_comm_protocol_cn_s

Definition at line 155 of file fipa_comm.h.

13.35.2.11 typedef struct fipa_comm_protocol_t* fipa_comm_protocol_p

Definition at line 172 of file fipa_comm.h.

13.35.2.12 typedef struct fipa_comm_protocol_t fipa_comm_protocol_s

Definition at line 173 of file fipa_comm.h.

13.35.2.13 typedef struct fipa_comm_reply_t* fipa_comm_reply_p

Definition at line 93 of file fipa_comm.h.

13.35.2.14 typedef struct fipa_comm_reply_t fipa_comm_reply_s

Definition at line 92 of file fipa_comm.h.

13.35.2.15 typedef struct fipa_comm_t fipa_comm_s

Definition at line 192 of file fipa_comm.h.

13.35.2.16 typedef struct fipa_list_t* fipa_list_p

Definition at line 141 of file fipa_comm.h.

13.35.2.17 typedef struct fipa_list_t fipa_list_s

Definition at line 140 of file fipa_comm.h.

13.35.2.18 typedef struct fipa_comm_t* MCFIPAComm_p

Definition at line 195 of file fipa_comm.h.

13.35.2.19 typedef struct fipa_comm_t MCFIPAComm_s

Definition at line 194 of file fipa_comm.h.

13.35.2.20 typedef int(* protocol_contract_net_callback)(int numBids, char **bids, char *winners)

Definition at line 18 of file fipa_comm.h.

13.35.3 Enumeration Type Documentation**13.35.3.1 enum fipa_comm_action_type_e**

Enumerator:

FIPA_COMM_ACTION_ERROR
FIPA_COMM_ACTION_BOTH
FIPA_COMM_ACTION_REPLY
FIPA_COMM_ACTION_CALLBACK
FIPA_COMM_ACTION_NONE

Definition at line 45 of file fipa_comm.h.

13.35.3.2 enum fipa_comm_check_type_e

Enumerator:

FIPA_COMM_CHECK_ERROR
FIPA_COMM_CHECK_SENDER_NAME
FIPA_COMM_CHECK_SENDER_ADDRESS
FIPA_COMM_CHECK_CONTENT
FIPA_COMM_CHECK_CONVERSATIONID
FIPA_COMM_CHECK_LANGUAGE
FIPA_COMM_CHECK_ONTOLOGY
FIPA_COMM_CHECK_ENCODING

Definition at line 34 of file fipa_comm.h.

13.35.3.3 enum fipa_comm_logic_type_e

Enumerator:

FIPA_COMM_LOGIC_ERROR
FIPA_COMM_LOGIC_OR

FIPA_COMM_LOGIC_AND

Definition at line 28 of file fipa_comm.h.

13.35.3.4 enum fipa_comm_protocol_cn_state_e

Enumerator:

FIPA_COMM_PROTOCOL_CN_ERROR

FIPA_COMM_PROTOCOL_CN_START

FIPA_COMM_PROTOCOL_CN_CFP

FIPA_COMM_PROTOCOL_CN_BID

FIPA_COMM_PROTOCOL_CN_RESULT

FIPA_COMM_PROTOCOL_CN_DONE

Definition at line 127 of file fipa_comm.h.

13.35.3.5 enum fipa_comm_type_e

Enumerator:

FIPA_COMM_ACTION

FIPA_COMM_CHECK

FIPA_COMM_REPLY

FIPA_COMM_CALLBACK

FIPA_COMM_END

Definition at line 20 of file fipa_comm.h.

13.35.4 Function Documentation

- 13.35.4.1 void fipa_comm_protocol_info_COPY (fipa_comm_protocol_p *proto*, void * *info*)
- 13.35.4.2 int fipa_list_Add (fipa_list_p * *list*, fipa_acl_message_p *acl*)
- 13.35.4.3 fipa_comm_protocol_p mc_FIPAComm_AddNewProtocol (fipa_comm_p *fcomm*, enum fipa_protocol_e *type*, char * *id*)
- 13.35.4.4 int mc_FIPAComm_DeregisterPerformative (fipa_comm_p *fcomm*, enum fipa_performative_e *type*)
- 13.35.4.5 void mc_FIPAComm_Destroy (fipa_comm_p)
- 13.35.4.6 int mc_FIPAComm_FSM_CN (fipa_comm_p *fcomm*, fipa_acl_message_p *acl*)
- 13.35.4.7 int mc_FIPAComm_HandleMessage (fipa_comm_p *fcomm*, fipa_acl_message_p *acl*)
- 13.35.4.8 int mc_FIPAComm_HandleMessageOnce (fipa_comm_p *fcomm*)
- 13.35.4.9 int mc_FIPAComm_HandleMessageWait (fipa_comm_p *fcomm*)
- 13.35.4.10 int mc_FIPAComm_HandleProtocol (fipa_comm_p *fcomm*, fipa_acl_message_p *acl*)
- 13.35.4.11 fipa_comm_p mc_FIPAComm_New ()
- 13.35.4.12 fipa_comm_protocol_p mc_FIPAComm_NewProtocol (enum fipa_protocol_e *type*, char * *id*)
- 13.35.4.13 int mc_FIPAComm_RegisterAction (fipa_comm_p *fcomm*, enum fipa_performative_e *pType*, enum fipa_comm_action_type_e *aType*, ...)
- 13.35.4.14 int mc_FIPAComm_RegisterActionS (fipa_performative_e *type*, fipa_comm_action_p *act*)
- 13.35.4.15 int mc_FIPAComm_RegisterPerformative (fipa_comm_p *fcomm*, enum fipa_performative_e *type*)
- 13.35.4.16 int mc_FIPAComm_RegisterProtocol (fipa_comm_p *fcomm*, enum fipa_protocol_e *protocolType*, char * *protocolID*, void * *info*, char * *network*)
- 13.35.4.17 int mc_FIPAComm_RegisterProtocolCallback (fipa_comm_p *fcomm*, enum fipa_protocol_e *protocolType*, fipa_comm_callback_func)
- 13.35.4.18 int mc_FIPAComm_SetAgent (fipa_comm_p *fcomm*, void * *agent*, char * *agent_name*, char * *agent_address*)
- 13.35.4.19 int mc_FIPAComm_SetDefaultReply (fipa_comm_p *fcomm*, enum fipa_performative_e *type*, char * *content*)
- 13.35.4.20 int mc_FIPAComm_SetPerformativeDefaultReply (fipa_comm_p *fcomm*, enum fipa_performative_e *ptype*, enum fipa_performative_e *rtype*, char * *content*)
- 13.35.4.21 int mc_FIPAComm_SetProtocolState (fipa_comm_protocol_p *protocol*, char *state*)
- 13.35.4.22 int mc_FIPAComm_StartProtocol (fipa_comm_p *fcomm*, enum fipa_protocol_e *type*, char * *id*)
- 13.35.4.23 fipa_comm_protocol_p mc_FIPAProtocol_New ()

13.36 /home/dko/Projects/mobilec/trunk/src/include/host_id.h File Reference

Data Structures

- struct [host_id_s](#)

Typedefs

- typedef struct [host_id_s](#) [host_id_t](#)
- typedef [host_id_t](#) * [host_id_p](#)

Functions

- [host_id_p](#) [host_id_Initialize](#) (const char *hostname, [int](#) port)
- void [host_id_Destroy](#) ([host_id_p](#) host_id)
- [int](#) [host_id_Compare](#) ([host_id_p](#) host1, [host_id_p](#) host2)

13.36.1 Typedef Documentation

13.36.1.1 typedef [host_id_t](#)* [host_id_p](#)

Definition at line 43 of file [host_id.h](#).

13.36.1.2 typedef struct [host_id_s](#) [host_id_t](#)

13.36.2 Function Documentation

13.36.2.1 [int](#) [host_id_Compare](#) ([host_id_p](#) *host1*, [host_id_p](#) *host2*)

13.36.2.2 void [host_id_Destroy](#) ([host_id_p](#) *host_id*)

13.36.2.3 [host_id_p](#) [host_id_Initialize](#) (const char * *hostname*, [int](#) *port*)

13.37 /home/dko/Projects/mobilec/trunk/src/include/interpreter_variable_data.h File Reference

```
#include <ch.h>
```

Data Structures

- struct [interpreter_variable_data_s](#)

Typedefs

- typedef struct [interpreter_variable_data_s](#) [interpreter_variable_data_t](#)
- typedef [interpreter_variable_data_t](#) * [interpreter_variable_data_p](#)

Functions

- [interpreter_variable_data_p](#) [interpreter_variable_data_New](#) (void)
- [interpreter_variable_data_p](#) [interpreter_variable_data_Copy](#) ([interpreter_variable_data_p](#) src)
- [interpreter_variable_data_p](#) [interpreter_variable_data_InitializeFromAgent](#) (struct [agent_s](#) *agent)
- [interpreter_variable_data_p](#) [interpreter_variable_data_Initialize](#) (struct [agent_s](#) *agent, const char *varname)
- int [interpreter_variable_data_Destroy](#) ([interpreter_variable_data_p](#) agent_return_data)

13.37.1 Typedef Documentation

13.37.1.1 typedef [interpreter_variable_data_t](#)* [interpreter_variable_data_p](#)

Definition at line 51 of file [interpreter_variable_data.h](#).

13.37.1.2 typedef struct [interpreter_variable_data_s](#) [interpreter_variable_data_t](#)

Definition at line 50 of file [interpreter_variable_data.h](#).

13.37.2 Function Documentation

13.37.2.1 [interpreter_variable_data_p](#) [interpreter_variable_data_Copy](#) ([interpreter_variable_data_p](#) src)

Definition at line 235 of file [agent_return_data.c](#).

References [interpreter_variable_data_s::array_dim](#), [interpreter_variable_data_s::array_extent](#), [interpreter_variable_data_s::data](#), [interpreter_variable_data_s::data_type](#), [interpreter_variable_data_New\(\)](#), [interpreter_variable_data_s::name](#), and [interpreter_variable_data_s::size](#).

Referenced by [agent_task_Copy\(\)](#).

13.37.2.2 int interpreter_variable_data_Destroy (interpreter_variable_data_p agent_return_data)

Definition at line 216 of file agent_return_data.c.

References interpreter_variable_data_s::array_extent, interpreter_variable_data_s::data, MC_SUCCESS, and interpreter_variable_data_s::name.

Referenced by agent_RunChScriptThread(), and agent_task_Destroy().

13.37.2.3 interpreter_variable_data_p interpreter_variable_data_Initialize (struct agent_s * agent, const char * varname)

13.37.2.4 interpreter_variable_data_p interpreter_variable_data_InitializeFromAgent (struct agent_s * agent)

13.37.2.5 interpreter_variable_data_p interpreter_variable_data_New (void)

Definition at line 46 of file agent_return_data.c.

References interpreter_variable_data_s::array_dim, interpreter_variable_data_s::array_extent, CHECK_NULL, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, interpreter_variable_data_s::name, and interpreter_variable_data_s::size.

Referenced by agent_xml_parse__data(), interpreter_variable_data_Copy(), and MC_SaveData_chdl().

13.38 /home/dko/Projects/mobilec/trunk/src/include/libmc.h File Reference

MobileC api header file. `#include <embedch.h>`

Data Structures

- struct [agency_s](#)
The agency handle.
- struct [MCAgencyOptions_s](#)
User modifiable agency options.
- struct [agent_thread_arg_s](#)

Defines

- `#define` [THREAD_T](#) `pthread_t`
- `#define` [EXPORTMC](#)
- `#define` [MC_Wait](#)(arg1) `MC_MainLoop(arg1)`

Typedefs

- `typedef enum` [error_code_e](#) `error_code_t`
- `typedef enum` [MC_SteerCommand_e](#) `MC_SteerCommand_t`
Available commands for MC_Steer.
- `typedef struct` [agency_s](#) `agency_t`
The agency handle.
- `typedef` [agency_t](#) * `agency_p`
- `typedef` [agency_p](#) `MCAgency_t`
- `typedef struct` [MCAgencyOptions_s](#) `MCAgencyOptions_t`
User modifiable agency options.
- `typedef struct` [agent_thread_arg_s](#) `agent_thread_arg_t`
- `typedef struct` [agent_s](#) `agent_t`
- `typedef` [agent_t](#) * `MCAgent_t`
- `typedef` [agent_t](#) * `agent_p`

Enumerations

- `enum` [error_code_e](#) {
[MC_SUCCESS](#) = 0, [MC_ERR](#), [MC_ERR_CONNECT](#), [MC_ERR_PARSE](#),
[MC_ERR_EMPTY](#), [MC_ERR_INVALID](#), [MC_ERR_INVALID_ARGS](#), [MC_ERR_NOT_FOUND](#),
[MC_ERR_MEMORY](#), [MC_ERR_SEND](#), [MC_WARN_DUPLICATE](#), [MC_SUCCESS](#) = 0,

```
ERR, MC_ERR_CONNECT, MC_ERR_PARSE, MC_ERR_EMPTY,
MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_ERR_NOT_FOUND, MC_ERR_
MEMORY,
MC_ERR_SEND, MC_WARN_DUPLICATE }
```

- enum MC_ThreadIndex_e {

```
MC_THREAD_DF = 0, MC_THREAD_AMS, MC_THREAD_ACC, MC_THREAD_CP,
MC_THREAD_AGENT, MC_THREAD_ALL }
```

MobileC Module indices.

- enum MC_SteerCommand_e { MC_RUN = 0, MC_SUSPEND, MC_RESTART, MC_STOP }

Available commands for MC_Steer.

- enum MC_Signal_e {

```
MC_NO_SIGNAL = 0x00, MC_RECV_CONNECTION = 0x01, MC_RECV_MESSAGE = 0x02,
MC_RECV_AGENT = 0x04,
MC_RECV_RETURN = 0x08, MC_EXEC_AGENT = 0x10, MC_ALL_SIGNALS = 0x20 }
```

MobileC system signals.

- enum MC_AgentType_e { MC_NONE = -1, MC_REMOTE_AGENT = 0, MC_LOCAL_AGENT, MC_RETURN_AGENT }

- enum MC_AgentStatus_e {

```
MC_NO_STATUS = -1, MC_WAIT_CH = 0, MC_WAIT_MESSGSEND, MC_AGENT_ACTIVE,
MC_AGENT_NEUTRAL, MC_AGENT_SUSPENDED, MC_WAIT_FINISHED }
```

An agent's current execution state.

- enum fipa_performative_e {

```
FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,
FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,
FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,
FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,
FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,
FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_
SUBSCRIBE,
FIPA_ERROR = -1, FIPA_ZERO, FIPA_ACCEPT_PROPOSAL, FIPA_AGREE,
FIPA_CANCEL, FIPA_CALL_FOR_PROPOSAL, FIPA_CONFIRM, FIPA_DISCONFIRM,
FIPA_FAILURE, FIPA_INFORM, FIPA_INFORM_IF, FIPA_INFORM_REF,
FIPA_NOT_UNDERSTOOD, FIPA_PROPOGATE, FIPA_PROPOSE, FIPA_PROXY,
FIPA_QUERY_IF, FIPA_QUERY_REF, FIPA_REFUSE, FIPA_REJECT_PROPOSAL,
FIPA_REQUEST, FIPA_REQUEST_WHEN, FIPA_REQUEST_WHENEVER, FIPA_
SUBSCRIBE }
```

Functions

- EXPORTMC [int MC_AclDestroy](#) (struct [fipa_acl_message_s](#) *message)
Destroy a FIPA ACL message.
- EXPORTMC struct [fipa_acl_message_s](#) * [MC_AclNew](#) (void)
Allocate a new ACL Message.
- EXPORTMC [int MC_AclPost](#) ([MCAgent_t](#) agent, struct [fipa_acl_message_s](#) *message)
Post ACL message to agent.
- EXPORTMC struct [fipa_acl_message_s](#) * [MC_AclReply](#) (struct [fipa_acl_message_s](#) *acl_message)
Reply to an ACL message.
- EXPORTMC struct [fipa_acl_message_s](#) * [MC_AclRetrieve](#) ([MCAgent_t](#) agent)
Retrieve an ACL message.
- EXPORTMC [int MC_AclSend](#) ([MCAgency_t](#) attr, struct [fipa_acl_message_s](#) *acl)
Send a composed ACL Message.
- EXPORTMC struct [fipa_acl_message_s](#) * [MC_AclWaitRetrieve](#) ([MCAgent_t](#) agent)
Wait for and retrieve an ACL message.
- EXPORTMC [int MC_AclSetProtocol](#) (struct [fipa_acl_message_s](#) *acl, enum [fipa_protocol_e](#) performative)
- EXPORTMC [int MC_AclSetConversationID](#) (struct [fipa_acl_message_s](#) *acl, char *id)
- EXPORTMC [int MC_AclSetPerformative](#) (struct [fipa_acl_message_s](#) *acl, enum [fipa_performative_e](#) performative)
- EXPORTMC [int MC_AclSetSender](#) (struct [fipa_acl_message_s](#) *acl, const char *name, const char *address)
- EXPORTMC [int MC_AclAddReceiver](#) (struct [fipa_acl_message_s](#) *acl, const char *name, const char *address)
- EXPORTMC [int MC_AclAddReplyTo](#) (struct [fipa_acl_message_s](#) *acl, const char *name, const char *address)
- EXPORTMC [int MC_AclSetContent](#) (struct [fipa_acl_message_s](#) *acl, const char *content)
- EXPORTMC [int MC_AddAgent](#) ([MCAgency_t](#) attr, [MCAgent_t](#) agent)
Add an agent to the agency 'attr'.
- [int MC_AddStationaryAgent](#) ([MCAgency_t](#) agency, void *(*agent_thread)(struct [agent_thread_arg_s](#) *), const char *name, void *agent_args)
- const void * [MC_AgentVariableRetrieve](#) ([MCAgent_t](#) agent, const char *var_name, [int](#) task_num)
Retrieve a pointer to a previously saved variable.
- [int MC_AgentVariableRetrieveInfo](#) ([MCAgent_t](#) agent, const char *var_name, [int](#) task_num, const void **data, [int](#) *dim, const [int](#) **extent)
Retrieve a info about a previously saved variable.
- [int MC_AgentVariableSave](#) ([MCAgent_t](#) agent, const char *var_name)
Mark an agent variable for saving.

- EXPORTMC [int MC_BarrierDelete](#) (MCAgency_t attr, [int](#) id)
Find and delete an initialized MobileC Barrier.
- EXPORTMC [int MC_BarrierInit](#) (MCAgency_t attr, [int](#) id, [int](#) num_procs)
Initialize a MobileC Barrier.
- EXPORTMC [int MC_CallAgentFunc](#) (MCAgent_t agent, const char *funcName, void *returnVal, [int](#) numArgs,...)
Use custom ChOptions_t type for internal Ch interpreter.
- EXPORTMC [int MC_CallAgentFuncArg](#) (MCAgent_t agent, const char *funcName, void *returnVal, void *arg)
Calls a function defined in an agent.
- EXPORTMC [int MC_CallAgentFuncV](#) (MCAgent_t agent, const char *funcName, void *returnVal, va_list ap)
Calls a function defined in an agent.
- EXPORTMC [int MC_CallAgentFuncVar](#) (MCAgent_t agent, const char *funcName, void *returnVal, ChVaList_t arglist)
- EXPORTMC [MCAgent_t MC_ComposeAgent](#) (const char *name, const char *home, const char *owner, const char *code, const char *return_var_name, const char *server, [int](#) persistent)
Compose a new agent dynamically without using a prewritten XML file.
- EXPORTMC [MCAgent_t MC_ComposeAgentS](#) (const char *name, const char *home, const char *owner, const char *code, const char *return_var_name, const char *server, [int](#) persistent, const char *workgroup_code)
Compose a new agent dynamically without using a prewritten XML file.
- EXPORTMC [MCAgent_t MC_ComposeAgentFromFile](#) (const char *name, const char *home, const char *owner, const char *filename, const char *return_var_name, const char *server, [int](#) persistent)
Compose a new agent dynamically from a source code file.
- EXPORTMC [MCAgent_t MC_ComposeAgentFromFileS](#) (const char *name, const char *home, const char *owner, const char *filename, const char *return_var_name, const char *server, [int](#) persistent, const char *workgroup_code)
Compose a new agent dynamically from a source code file.
- EXPORTMC [int MC_CondBroadcast](#) (MCAgency_t attr, [int](#) id)
Wakes up all agents/threads waiting on a condition variable.
- EXPORTMC [int MC_CondSignal](#) (MCAgency_t attr, [int](#) id)
Wakes up at least one thread waiting on a condition variable.
- EXPORTMC [int MC_CondReset](#) (MCAgency_t attr, [int](#) id)
Reset a previously signalled MobileC condition variable.
- EXPORTMC [int MC_CondWait](#) (MCAgency_t attr, [int](#) id)
Wait on a MobileC synchronization variable.
- [int MC_CopyAgent](#) (MCAgent_t *agent_out, const MCAgent_t agent_in)

Performs a deep-copy of an agent structure.

- EXPORTMC [int MC_DeleteAgent](#) ([MCAgent_t](#) agent)
Stop and remove an agent.
- EXPORTMC [int MC_DeleteAgentWG](#) ([MCAgent_t](#) calling_agent, [MCAgent_t](#) agent)
Stop and remove an agent in the same workgroup.
- EXPORTMC [int MC_End](#) ([MCAgency_t](#) attr)
End an agency.
- [int MC_DestroyServiceSearchResult](#) (char **agentName, char **serviceName, [int](#) *agentID, [int](#) numResult)
Free memory allocated by a Service Search operation.
- EXPORTMC [MCAgent_t MC_FindAgentByName](#) ([MCAgency_t](#) attr, const char *name)
Find an agent by its name.
- EXPORTMC [MCAgent_t MC_FindAgentByID](#) ([MCAgency_t](#) attr, [int](#) ID)
Find an agent by its id.
- EXPORTMC void * [MC_GetAgentExecEngine](#) ([MCAgent_t](#) agent)
Retrieve an agent's Ch interpreter.
- EXPORTMC [int MC_GetAgentID](#) ([MCAgent_t](#) agent)
Retrieve an agent's id.
- EXPORTMC char * [MC_GetAgentName](#) ([MCAgent_t](#) agent)
- EXPORTMC [int MC_GetAgentNumTasks](#) ([MCAgent_t](#) agent)
Retrive the number of tasks an agent has.
- EXPORTMC [int MC_GetAgentReturnData](#) ([MCAgent_t](#) agent, [int](#) task_num, void **data, [int](#) *dim, [int](#) **extent)
Get an agent's return data.
- EXPORTMC [int MC_GetAgentStatus](#) ([MCAgent_t](#) agent)
Get an agent's current status.
- EXPORTMC enum [MC_AgentType_e MC_GetAgentType](#) ([MCAgent_t](#) agent)
Get an agent's type.
- EXPORTMC char * [MC_GetAgentXMLString](#) ([MCAgent_t](#) agent)
Get an agent's xml string.
- EXPORTMC [int MC_HaltAgency](#) ([MCAgency_t](#) agency)
Halt an agency: Do not process new entries in queues.
- EXPORTMC [MCAgency_t MC_Initialize](#) ([int](#) port, [MCAgencyOptions_t](#) *options)
Initialize and start a MobileC agency.

- EXPORTMC [int MC_InitializeAgencyOptions](#) (struct [MCAgencyOptions_s](#) *options)
Initialize MobileC options.
- EXPORTMC [int MC_MainLoop](#) ([MCAgency_t](#) attr)
Wait indefinitely.
- EXPORTMC [int MC_LoadAgentFromFile](#) ([MCAgency_t](#) attr, const char *filename)
Load an agent from a file into an agency.
- EXPORTMC [int MC_MigrateAgent](#) ([MCAgent_t](#) agent, const char *hostname, [int](#) port)
Migrates a running agent to another host.
- EXPORTMC [int MC_MutexLock](#) ([MCAgency_t](#) attr, [int](#) id)
Locks a MobileC synchronization variable as a mutex.
- EXPORTMC [int MC_MutexUnlock](#) ([MCAgency_t](#) attr, [int](#) id)
- EXPORTMC [int MC_RegisterService](#) ([MCAgency_t](#) agency, [MCAgent_t](#) agent, [int](#) agentID, const char *agentName, char **serviceNames, [int](#) numServices)
Register a new service with the Directory Facilitator.
- EXPORTMC [int MC_ResumeAgency](#) ([MCAgency_t](#) agency)
Resumes a halted agency.
- EXPORTMC [MCAgent_t MC_RetrieveAgent](#) ([MCAgency_t](#) attr)
Retrieves the oldest agent from an agency.
- EXPORTMC [int MC_SemaphorePost](#) ([MCAgency_t](#) attr, [int](#) id)
Post to a MobileC synchronization variable semaphore.
- EXPORTMC [int MC_SemaphoreWait](#) ([MCAgency_t](#) attr, [int](#) id)
Decreases a MobileC synchronization variable semaphore count by one.
- EXPORTMC [int MC_SetDefaultAgentStatus](#) ([MCAgency_t](#) agency, enum [MC_AgentStatus_e](#) status)
Sets default incoming agent status.
- EXPORTMC [int MC_SetThreadOn](#) ([MCAgencyOptions_t](#) *options, enum [MC_ThreadIndex_e](#) index)
Sets a MobileC thread to "on" status.
- EXPORTMC [int MC_SetThreadsAllOn](#) ([MCAgencyOptions_t](#) *options)
Set all Mobile-C threads on.
- EXPORTMC [int MC_SetThreadOff](#) ([MCAgencyOptions_t](#) *options, enum [MC_ThreadIndex_e](#) index)
Sets a MobileC thread to "off" status.
- EXPORTMC [int MC_SetThreadsAllOff](#) ([MCAgencyOptions_t](#) *options)
Set all MobileC threads to 'off' status.

- EXPORTMC [int MC_PrintAgentCode](#) ([MCAgent_t](#) agent)
Prints an agents code to stdout.
- EXPORTMC [char * MC_RetrieveAgentCode](#) ([MCAgent_t](#) agent)
Retrieves an agent's Ch code.
- EXPORTMC [int MC_ResetSignal](#) ([MCAgency_t](#) attr)
Reset a MobileC signal.
- EXPORTMC [int MC_SearchForService](#) ([MCAgency_t](#) attr, [const char *searchString](#), [char ***agentNames](#), [char ***serviceNames](#), [int **agentIDs](#), [int *numResults](#))
Search the directory facilitator for a service.
- EXPORTMC [int MC_SendAgentMigrationMessage](#) ([MCAgency_t](#) attr, [const char *message](#), [const char *hostname](#), [int port](#))
Sends an agent migration message.
- EXPORTMC [int MC_SendAgentMigrationMessageFile](#) ([MCAgency_t](#) attr, [const char *filename](#), [const char *hostname](#), [int port](#))
Sends an agent migration message.
- EXPORTMC [int MC_SetAgentStatus](#) ([MCAgent_t](#) agent, [int status](#))
Set an agent's status.
- EXPORTMC [int MC_Steer](#) ([MCAgency_t](#) attr, [int\(*funcptr\)\(void *data\)](#), [void *arg](#))
Set up a steerable algorithm.
- EXPORTMC [enum MC_SteerCommand_e MC_SteerControl](#) ([void](#))
The MobileC user-algorithm steering function.
- EXPORTMC [int MC_SyncDelete](#) ([MCAgency_t](#) attr, [int id](#))
Deletes a previously initialized synchronization variable.
- EXPORTMC [int MC_SyncInit](#) ([MCAgency_t](#) attr, [int id](#))
Initializes a new MobileC synchronization variable.
- EXPORTMC [int MC_TerminateAgent](#) ([MCAgent_t](#) agent)
Halt a running agent.
- EXPORTMC [int MC_TerminateAgentWG](#) ([MCAgent_t](#) calling_agent, [MCAgent_t](#) agent)
- EXPORTMC [int MC_WaitAgent](#) ([MCAgency_t](#) attr)
Wait indefinitely.
- EXPORTMC [MCAgent_t MC_WaitRetrieveAgent](#) ([MCAgency_t](#) attr)
Wait and retrieve an agent.
- EXPORTMC [int MC_WaitSignal](#) ([MCAgency_t](#) attr, [int signals](#))
Wait for a MobileC signal.

13.38.1 Detailed Description

MobileC api header file.

Definition in file [libmc.h](#).

13.38.2 Define Documentation

13.38.2.1 `#define EXPORTMC`

Definition at line 147 of file libmc.h.

13.38.2.2 `#define MC_Wait(arg1) MC_MainLoop(arg1)`

Definition at line 152 of file libmc.h.

13.38.2.3 `#define THREAD_T pthread_t`

Definition at line 133 of file libmc.h.

Referenced by `acc_Thread()`, and `message_Send()`.

13.38.3 Typedef Documentation

13.38.3.1 `typedef agency_t* agency_p`

Definition at line 242 of file libmc.h.

13.38.3.2 `typedef struct agency_s agency_t`

The agency handle.

13.38.3.3 `typedef agent_t* agent_p`

Definition at line 275 of file libmc.h.

13.38.3.4 `typedef struct agent_s agent_t`

Definition at line 273 of file libmc.h.

13.38.3.5 `typedef struct agent_thread_arg_s agent_thread_arg_t`

13.38.3.6 `typedef enum error_code_e error_code_t`

13.38.3.7 `typedef enum MC_SteerCommand_e MC_SteerCommand_t`

Available commands for MC_Steer.

13.38.3.8 typedef agency_p MCAgency_t

Definition at line 243 of file libmc.h.

13.38.3.9 typedef struct MCAgencyOptions_s MCAgencyOptions_t

User modifiable agency [options](#).

13.38.3.10 typedef agent_t* MCAgent_t

Definition at line 274 of file libmc.h.

13.38.4 Enumeration Type Documentation

13.38.4.1 enum error_code_e

Enumerator:

MC_SUCCESS
MC_ERR
MC_ERR_CONNECT
MC_ERR_PARSE
MC_ERR_EMPTY
MC_ERR_INVALID
MC_ERR_INVALID_ARGS
MC_ERR_NOT_FOUND
MC_ERR_MEMORY
MC_ERR_SEND
MC_WARN_DUPLICATE
MC_SUCCESS
ERR
MC_ERR_CONNECT
MC_ERR_PARSE
MC_ERR_EMPTY
MC_ERR_INVALID
MC_ERR_INVALID_ARGS
MC_ERR_NOT_FOUND
MC_ERR_MEMORY
MC_ERR_SEND
MC_WARN_DUPLICATE

Definition at line 166 of file libmc.h.

13.38.4.2 enum fipa_performative_e

Enumerator:

FIPA_ERROR
FIPA_ZERO
FIPA_ACCEPT_PROPOSAL
FIPA_AGREE
FIPA_CANCEL
FIPA_CALL_FOR_PROPOSAL
FIPA_CONFIRM
FIPA_DISCONFIRM
FIPA_FAILURE
FIPA_INFORM
FIPA_INFORM_IF
FIPA_INFORM_REF
FIPA_NOT_UNDERSTOOD
FIPA_PROPOGATE
FIPA_PROPOSE
FIPA_PROXY
FIPA_QUERY_IF
FIPA_QUERY_REF
FIPA_REFUSE
FIPA_REJECT_PROPOSAL
FIPA_REQUEST
FIPA_REQUEST_WHEN
FIPA_REQUEST_WHENEVER
FIPA_SUBSCRIBE
FIPA_ERROR
FIPA_ZERO
FIPA_ACCEPT_PROPOSAL
FIPA_AGREE
FIPA_CANCEL
FIPA_CALL_FOR_PROPOSAL
FIPA_CONFIRM
FIPA_DISCONFIRM
FIPA_FAILURE
FIPA_INFORM
FIPA_INFORM_IF
FIPA_INFORM_REF
FIPA_NOT_UNDERSTOOD
FIPA_PROPOGATE
FIPA_PROPOSE

FIPA_PROXY
FIPA_QUERY_IF
FIPA_QUERY_REF
FIPA_REFUSE
FIPA_REJECT_PROPOSAL
FIPA_REQUEST
FIPA_REQUEST_WHEN
FIPA_REQUEST_WHENEVER
FIPA_SUBSCRIBE

Definition at line 297 of file libmc.h.

13.38.4.3 enum MC_AgentStatus_e

An agent's current execution state.

Enumerator:

MC_NO_STATUS
MC_WAIT_CH Waiting to be started
MC_WAIT_MESSGSEND Finished, waiting to migrate
MC_AGENT_ACTIVE Running
MC_AGENT_NEUTRAL Not running, but do not flush
MC_AGENT_SUSPENDED Unused
MC_WAIT_FINISHED Finished, waiting to be flushed

Definition at line 283 of file libmc.h.

13.38.4.4 enum MC_AgentType_e

Enumerator:

MC_NONE
MC_REMOTE_AGENT
MC_LOCAL_AGENT
MC_RETURN_AGENT

Definition at line 278 of file libmc.h.

13.38.4.5 enum MC_Signal_e

MobileC system signals.

Note:

Each signal is activated after the corresponding action. i.e. The 'MC_RECV_MESSAGE' signal is activated after a message is received.

See also:

[MC_WaitSignal\(\)](#), [MC_ResetSignal](#)

Enumerator:

MC_NO_SIGNAL No signal has been received
MC_RECV_CONNECTION A Connection has been received
MC_RECV_MESSAGE A Message has been received from a connection
MC_RECV_AGENT An normal agent has been parsed from a message
MC_RECV_RETURN A return agent has been parsed from a message
MC_EXEC_AGENT A normal agent has been executed.
MC_ALL_SIGNALS Catch any of the above signals

Definition at line 211 of file libmc.h.

13.38.4.6 enum MC_SteerCommand_e

Available commands for MC_Steer.

Enumerator:

MC_RUN Continue the algorithm
MC_SUSPEND Suspend/pause the algorithm
MC_RESTART Restart the algorithm from the beginning
MC_STOP Stop the algorithm

Definition at line 196 of file libmc.h.

13.38.4.7 enum MC_ThreadIndex_e

MobileC Module indices.

Enumerator:

MC_THREAD_DF Directory Facilitator
MC_THREAD_AMS Agent Managment system
MC_THREAD_ACC Agency communications
MC_THREAD_CP Command Prompt
MC_THREAD_AGENT Agent threads
MC_THREAD_ALL

Definition at line 184 of file libmc.h.

13.38.5 Function Documentation

13.38.5.1 EXPORTMC int MC_AclAddReceiver (struct fipa_acl_message_s * *acl*, const char * *name*, const char * *address*)

Definition at line 278 of file libmc.c.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_New()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_url_New()`, `fipa_url_sequence_New()`, `fipa_agent_identifier_s::name`, `fipa_url_sequence_s::num`, `fipa_agent_identifier_set_s::num`, `fipa_acl_message_s::receiver`, `fipa_acl_message_s::receiver_num`, `fipa_url_s::str`, and `fipa_url_sequence_s::urls`.

Referenced by `MC_AclAddReceiver_chdl()`.

13.38.5.2 EXPORTMC int MC_AclAddReplyTo (struct fipa_acl_message_s * *acl*, const char * *name*, const char * *address*)

Definition at line 314 of file `libmc.c`.

References `fipa_agent_identifier_s::addresses`, `fipa_agent_identifier_New()`, `fipa_agent_identifier_set_New()`, `fipa_agent_identifier_set_s::fipa_agent_identifiers`, `fipa_url_New()`, `fipa_url_sequence_New()`, `fipa_agent_identifier_s::name`, `fipa_url_sequence_s::num`, `fipa_agent_identifier_set_s::num`, `fipa_acl_message_s::reply_to`, `fipa_url_s::str`, and `fipa_url_sequence_s::urls`.

Referenced by `MC_AclAddReplyTo_chdl()`.

13.38.5.3 EXPORTMC int MC_AclDestroy (struct fipa_acl_message_s * *message*)

Destroy a FIPA ACL message.

Parameters:

message The ACL message to destroy

Returns:

0 on success, error code on failure.

Definition at line 78 of file `libmc.c`.

References `fipa_acl_message_Destroy()`.

Referenced by `MC_AclDestroy_chdl()`.

13.38.5.4 EXPORTMC struct fipa_acl_message_s* MC_AclNew (void) [read]

Allocate a new ACL Message.

Returns:

A newly allocated and empty ACL message.

Definition at line 84 of file `libmc.c`.

References `fipa_acl_message_New()`.

Referenced by `MC_AclNew_chdl()`.

13.38.5.5 EXPORTMC int MC_AclPost (MCAgent_t *agent*, struct fipa_acl_message_s * *message*)

Post ACL message to agent.

Parameters:

agent The agent to post the message to

message The message to post

Returns:

0 if successful, or error_code_t type.

Definition at line 89 of file libmc.c.

References agent_mailbox_Post(), and agent_s::mailbox.

Referenced by MC_AclPost_chdl(), and MC_AclSend().

13.38.5.6 EXPORTMC struct fipa_acl_message_s* MC_AclReply (struct fipa_acl_message_s * *acl_message*) [read]

Reply to an ACL message.

Parameters:

acl_message The incoming acl message to reply to

Returns:

A newly allocated ACL message

Note:

This function simply generates a new ACL message with the 'receiver' field automatically set to the 'sender' field of the incoming message.

Definition at line 95 of file libmc.c.

References fipa_Reply().

Referenced by MC_AclReply_chdl().

13.38.5.7 EXPORTMC struct fipa_acl_message_s* MC_AclRetrieve (MCAgent_t *agent*) [read]

Retrieve an ACL message.

Parameters:

agent Agent to retrieve message from.

Returns:

an ACL message struct on success or NULL on failure

Definition at line 101 of file libmc.c.

References agent_mailbox_Retrieve(), and agent_s::mailbox.

Referenced by MC_AclRetrieve_chdl().

13.38.5.8 EXPORTMC int MC_AclSend (MC_Agency_t *attr*, struct fipa_acl_message_s * *acl*)

Send a composed ACL Message.

Parameters:

- attr* An initialized and running MobileC agency
- acl* An allocated and fully composed ACL message.

Returns:

- 0 if successful, error code on failure.

Definition at line 107 of file libmc.c.

References fipa_agent_identifier_s::addresses, mtp_http_s::content, mtp_http_content_s::content_type, mtp_http_content_s::data, dynstring_Destroy(), FIPA_ACL, fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_Compose(), mtp_http_s::host, http_to_hostport(), MC_AclPost(), MC_FindAgentByName(), agency_s::mc_platform, dynstring_s::message, mtp_http_s::message_parts, message_Send(), message_s::message_type, mtp_http_CreateMessage(), mtp_http_Destroy(), mtp_http_New(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, fipa_agent_identifier_set_s::num, port, mc_platform_s::private_key, fipa_acl_message_s::receiver, fipa_url_s::str, mtp_http_s::target, message_s::target, and fipa_url_sequence_s::urls.

Referenced by MC_AclSend_chdl().

13.38.5.9 EXPORTMC int MC_AclSetContent (struct fipa_acl_message_s * *acl*, const char * *content*)

Definition at line 349 of file libmc.c.

References fipa_string_s::content, fipa_acl_message_s::content, fipa_string_Destroy(), and fipa_string_New().

Referenced by MC_AclSetContent_chdl().

13.38.5.10 EXPORTMC int MC_AclSetConversationID (struct fipa_acl_message_s * *acl*, char * *id*)

Definition at line 228 of file libmc.c.

References fipa_string_s::content, fipa_expression_s::content, fipa_acl_message_s::conversation_id, FIPA_EXPR_STRING, fipa_expression_New(), fipa_string_New(), fipa_expression_s::content_u::string, and fipa_expression_s::type.

Referenced by MC_AclSetConversationID_chdl().

13.38.5.11 EXPORTMC int MC_AclSetPerformative (struct fipa_acl_message_s * *acl*, enum fipa_performative_e *performative*)

Definition at line 247 of file libmc.c.

References fipa_acl_message_s::performative.

Referenced by MC_AclSetPerformative_chdl().

13.38.5.12 EXPORTMC int MC_AclSetProtocol (struct fipa_acl_message_s * *acl*, enum fipa_protocol_e *performative*)

Definition at line 212 of file libmc.c.

References FIPA_PROTOCOL_END, FIPA_PROTOCOL_ERROR, and fipa_acl_message_s::protocol.

Referenced by MC_AclSetProtocol_chdl().

13.38.5.13 EXPORTMC int MC_AclSetSender (struct fipa_acl_message_s * *acl*, const char * *name*, const char * *address*)

Definition at line 255 of file libmc.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_Destroy(), fipa_agent_identifier_New(), fipa_url_New(), fipa_url_sequence_New(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, fipa_acl_message_s::sender, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by MC_AclSetSender_chdl().

13.38.5.14 EXPORTMC struct fipa_acl_message_s* MC_AclWaitRetrieve (MCAgent_t *agent*) [read]

Wait for and retrieve an ACL message.

Parameters:

agent Agent to retrieve message from.

Returns:

an ACL message struct on success or NULL on failure

Definition at line 205 of file libmc.c.

References agent_mailbox_WaitRetrieve(), and agent_s::mailbox.

Referenced by MC_AclWaitRetrieve_chdl().

13.38.5.15 EXPORTMC int MC_AddAgent (MCAgency_t *attr*, MCAgent_t *agent*)

Add an agent to the agency '*attr*'.

Parameters:

attr a MobileC agency

agent An initialized MobileC agent

Returns:

0 if successful, or error_code_t type

Definition at line 366 of file libmc.c.

References mc_platform_s::agent_queue, mc_platform_s::ams, COND_SIGNAL, agency_s::mc_platform, agent_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by handler_COMPOSE_SEND(), and MC_AddAgent_chdl().

13.38.5.16 `int MC_AddStationaryAgent (MCAgency_t agency, void (*)(struct agent_thread_arg_s *) agent_thread, const char * name, void * agent_args)`

Definition at line 380 of file libmc.c.

References `agent_thread_arg_s::agent`, `agent_NewBinary()`, `mc_platform_s::agent_queue`, `agent_thread_arg_s::args`, `agent_thread_arg_s::attr`, `agency_s::mc_platform`, `agent_s::name`, `agent_thread_arg_s::thread`, and `THREAD_CREATE`.

13.38.5.17 `const void* MC_AgentVariableRetrieve (MCAgent_t agent, const char * var_name, int task_num)`

Retrieve a pointer to a previously saved variable.

Parameters:

agent A MobileC agent.

var_name The name of the saved variable that has previously been saved.

task_num The previous completed task from which to retrieve the saved variable.

Returns:

A pointer to the data on success or NULL on failure.

13.38.6 Examples

The following example demonstrates usage of [MC_AgentVariableRetrieve\(\)](#) from agent space.

Definition at line 401 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::data`, `agent_s::datastate`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_AgentVariableRetrieve_chdl()`.

13.38.6.1 `int MC_AgentVariableRetrieveInfo (MCAgent_t agent, const char * var_name, int task_num, const void ** data, int * dim, const int ** extent)`

Retrieve a info about a previously saved variable.

Parameters:

agent A MobileC agent.

var_name The name of the saved variable that has previously been saved.

task_num The previous completed task from which to retrieve the saved variable.

data (Output) The Variable Data

dim (Output) The dimension of the data array

extent (Output) The extents of the output array

Returns:

Error code.

Definition at line 419 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `agent_s::datastate`, `MC_ERR_NOT_FOUND`, `MC_SUCCESS`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

13.38.6.2 `int MC_AgentVariableSave (MCAgent_t agent, const char * var_name)`

Mark an agent variable for saving.

Parameters:

agent A MobileC agent.

var_name The name of the variable to mark for saving.

Returns:

0 on success, non-zero on failure.

See also:

`test1.xml`

13.38.7 Examples

See `agent_saved_variables_example/test1.xml` for an example of usage of this api function.

13.38.8 Examples

The following example demonstrates usage of `MC_AgentVariableSave()` from agent space.

Definition at line 440 of file libmc.c.

References `agent_s::datastate`, `MC_ERR_MEMORY`, `agent_task_s::num_saved_variables`, `agent_task_s::saved_variables`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_AgentVariableSave_chdl()`.

13.38.8.1 `EXPORTMC int MC_BarrierDelete (MCAgency_t attr, int id)`

Find and delete an initialized MobileC Barrier.

Parameters:

attr A running MobileC agency

id The id of the barrier node to delete

Returns:

returns 0 on success, error if the node is not found or other failure.

Definition at line 501 of file libmc.c.

References `mc_platform_s::barrier_queue`, `barrier_queue_Delete()`, and `agency_s::mc_platform`.

Referenced by `MC_BarrierDelete_chdl()`.

13.38.8.2 EXPORTMC int MC_BarrierInit (MCAgency_t *attr*, int *id*, int *num_procs*)

Initialize a MobileC Barrier.

Parameters:

attr A running MobileC agency
id The requested barrier id
num_procs The number of agents/threads/processes that will wait on the barrier

Returns:

The allocated barrier id. May differ from the requested id if it is already in use.

13.38.9 Examples

The following example demonstrates an agent which sets up an MC_Barrier.

Definition at line 487 of file libmc.c.

References barrier_node_Initialize(), mc_platform_s::barrier_queue, barrier_queue_Add(), barrier_queue_Get(), MC_ERR, agency_s::mc_platform, MC_SUCCESS, and node.

Referenced by MC_BarrierInit_chdl().

13.38.9.1 EXPORTMC int MC_CallAgentFunc (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, int *numArgs*, ...)

Use custom ChOptions_t type for internal Ch interpreter.

Parameters:

attr A running MobileC agency
options Initialized Ch *options* structure

Returns:

0 on success, error_code_t type on failure Calls a function defined in an agent

Parameters:

agent An initialized and executed MobileC agent
funcName The name of the function to call
returnVal The agent function's return value
numArgs The number of arguments supplied to the agent function
... A variable argument list to be supplied to the agent function

Returns:

0 if successful, error_code_t type on failure

13.38.10 Example

Definition at line 507 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

13.38.10.1 EXPORTMC int MC_CallAgentFuncArg (MCAgent_t *agent*, const char **funcName*, void **returnVal*, void **arg*)

Calls a function defined in an agent.

Parameters:

agent An initialized and executed MobileC agent
funcName The name of the function to call
returnVal The agent function's return value
arg The agent functions argument

Note:

The agent function must be of the form 'void* func(void* arg);'

Returns:

0 if successful, error_code_t type on failure

13.38.11 Example

Definition at line 528 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

13.38.11.1 EXPORTMC int MC_CallAgentFuncV (MCAgent_t *agent*, const char **funcName*, void **returnVal*, va_list *ap*)

Calls a function defined in an agent.

Parameters:

agent An initialized and executed MobileC agent
funcName The name of the function to call
returnVal The agent function's return value
ap A variable argument list

Returns:

0 if successful, error_code_t type on failure

Definition at line 547 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

13.38.11.2 EXPORTMC int MC_CallAgentFuncVar (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, ChVaList_t *arglist*)

Definition at line 567 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

Referenced by MC_CallAgentFunc_chdl().

13.38.11.3 EXPORTMC MCAgent_t MC_ComposeAgent (const char * *name*, const char * *home*, const char * *owner*, const char * *code*, const char * *return_var_name*, const char * *server*, int *persistent*)

Compose a new agent dynamically without using a prewritten XML file.

Parameters:

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

persitent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 607 of file libmc.c.

References MC_ComposeAgentS().

13.38.11.4 EXPORTMC MCAgent_t MC_ComposeAgentFromFile (const char * *name*, const char * *home*, const char * *owner*, const char * *filename*, const char * *return_var_name*, const char * *server*, int *persistent*)

Compose a new agent dynamically from a source code file.

Parameters:

filename The filename of the source file

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

persistent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 707 of file libmc.c.

References MC_ComposeAgentFromFileS().

Referenced by handler_COMPOSE_SEND().

13.38.11.5 EXPORTMC MCAgent_t MC_ComposeAgentFromFileS (const char * *name*, const char * *home*, const char * *owner*, const char * *filename*, const char * *return_var_name*, const char * *server*, int *persistent*, const char * *workgroup_code*)

Compose a new agent dynamically from a source code file.

Parameters:

filename The filename of the source file

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

workgroup_code The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

persistent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 729 of file libmc.c.

References agent_thread_arg_s::agent, and MC_ComposeAgentS().

Referenced by MC_ComposeAgentFromFile().

13.38.11.6 EXPORTMC MCAgent_t MC_ComposeAgentS (const char * *name*, const char * *home*, const char * *owner*, const char * *code*, const char * *return_var_name*, const char * *server*, int *persistent*, const char * *workgroup_code*)

Compose a new agent dynamically without using a prewritten XML file.

Parameters:

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

workgroup_code The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

persistent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid `MCAgent_t` structure on success or `NULL` on failure.

Definition at line 630 of file `libmc.c`.

References `agent_thread_arg_s::agent`, `agent_datastate_s::agent_code`, `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_datastate_New()`, `agent_New()`, `agent_s::agent_status`, `agent_task_New()`, `agent_s::agent_type`, `agent_s::datastate`, `agent_s::home`, `MC_LOCAL_AGENT`, `MC_WAIT_MESSGSEND`, `agent_s::name`, `agent_datastate_s::number_of_tasks`, `agent_s::orphan`, `agent_s::owner`, `agent_datastate_s::persistent`, `agent_task_s::server_name`, `agent_datastate_s::tasks`, `agent_task_s::var_name`, and `agent_s::wg_code`.

Referenced by `MC_ComposeAgent()`, `MC_ComposeAgent_chdl()`, `MC_ComposeAgentFromFileS()`, and `MC_ComposeAgentS_chdl()`.

13.38.11.7 EXPORTMC int MC_CondBroadcast (MCAgency_t attr, int id)

Wakes up all agents/threads waiting on a condition variable.

Parameters:

attr A MobileC agency

id Synchronization variable id to broadcast to

See also:

[MC_SyncInit\(\)](#), [MC_CondSignal\(\)](#)

Returns:

0 on success, `error_code_t` type on failure

Definition at line 772 of file `libmc.c`.

References `syncListNode_s::cond`, `COND_BROADCAST`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondBroadcast_chdl()`.

13.38.11.8 EXPORTMC int MC_CondReset (MCAgency_t *attr*, int *id*)

Reset a previously signalled MobileC condition variable.

Parameters:

attr A MobileC Agency

id The synchronization variable id to reset

See also:

[MC_SyncInit\(\)](#)

Returns:

0 on success, error_code_t type on failure

Definition at line 824 of file libmc.c.

References `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondReset_chdl()`.

13.38.11.9 EXPORTMC int MC_CondSignal (MCAgency_t *attr*, int *id*)

Wakes up at least one thread waiting on a condition variable.

Parameters:

attr A MobileC agency

id synchronization variable id

See also:

[MC_SyncInit\(\)](#)

Returns:

0 on success, error_code_t type on failure

13.38.12 Example

The following example demonstrates the agent-space version of the function, which is nearly identical to the binary space api function.

Definition at line 787 of file libmc.c.

References `syncListNode_s::cond`, `COND_SIGNAL`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondSignal_chdl()`.

13.38.12.1 EXPORTMC int MC_CondWait (MCAgency_t *attr*, int *id*)

Wait on a MobileC synchronization variable.

Parameters:

attr A MobileC agency
id a synchronization variable id

See also:

[MC_SyncInit\(\)](#)

Returns:

0 on success, error_code_t type on failure

13.38.13 Example

The following example demonstrates the agent-space version of this function.

Definition at line 802 of file libmc.c.

References syncListNode_s::cond, COND_WAIT, syncListNode_s::lock, MC_ERR_NOT_FOUND, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, syncListNode_s::signalled, mc_platform_s::syncList, and syncListFind().

Referenced by MC_CondWait_chdl().

13.38.13.1 int MC_CopyAgent (MCAgent_t * *agent_out*, const MCAgent_t *agent_in*)

Performs a deep-copy of an agent structure.

Parameters:

agent_out A pointer to the agent to copy to.
agent_in The agent to copy

Returns:

0 on success, error_code_t type on failure.

Definition at line 842 of file libmc.c.

References agent_Copy(), and MC_SUCCESS.

13.38.13.2 EXPORTMC int MC_DeleteAgent (MCAgent_t *agent*)

Stop and remove an agent.

Parameters:

agent An agent in any state (running, waiting, etc)

Returns:

0 on success, error_code_t type on failure

Definition at line 849 of file libmc.c.

References CHECK_NULL, MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_SetAgentStatus(), MC_SUCCESS, MC_TerminateAgent(), MC_WAIT_FINISHED, and agent_s::wg_code.

Referenced by MC_DeleteAgent_chdl().

13.38.13.3 EXPORTMC int MC_DeleteAgentWG (MCAgent_t calling_agent, MCAgent_t agent)

Stop and remove an agent in the same workgroup.

Parameters:

calling_agent The calling agent

agent An agent in any state (running, waiting, etc)

Note:

The agents must belong to the same workgroup.

Returns:

0 on success, error_code_t type on failure

Definition at line 869 of file libmc.c.

References CHECK_NULL, MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_SetAgentStatus(), MC_SUCCESS, MC_TerminateAgentWG(), MC_WAIT_FINISHED, and agent_s::wg_code.

Referenced by MC_DeleteAgentWG_chdl().

13.38.13.4 int MC_DestroyServiceSearchResult (char ** agentName, char ** serviceName, int * agentID, int numResult)

Free memory allocated by a Service Search operation.

Parameters:

agentName agent names returned by a search operation.

serviceName service names return by a search operation.

agentID list of agent id's returned by a search operation.

numResult The number of hits returned by a search operation.

Returns:

0 on success, error code on failure.

Definition at line 892 of file libmc.c.

Referenced by MC_DestroyServiceSearchResult_chdl().

13.38.13.5 EXPORTMC int MC_End (MCAgency_t *attr*)

End an agency.

Parameters:

attr A running agency

Returns:

0 on success, error_code_t type on failure

13.38.14 Example

Definition at line 941 of file libmc.c.

References mc_platform_s::acc, mc_platform_s::ams, mc_platform_s::cmd_prompt, COND_-BROADCAST, COND_SIGNAL, mc_platform_s::connection_queue, mc_platform_s::df, GET_-THREAD_MODE, agency_s::hostName, agency_s::mc_platform, mc_platform_Destroy(), MC_-THREAD_ACC, MC_THREAD_AMS, MC_THREAD_CP, MC_THREAD_DF, mc_platform_s::message_queue, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::quit, mc_platform_s::quit_cond, mc_platform_s::quit_lock, cmd_prompt_s::thread, THREAD_CANCEL, THREAD_JOIN, and agency_s::threads.

Referenced by MC_End_chdl().

13.38.14.1 EXPORTMC MCAgent_t MC_FindAgentByID (MCAgency_t *attr*, int *ID*)

Find an agent by its id.

Parameters:

attr the agency to search

ID the id to search for

Returns:

a valid agent on success, NULL on failure

Definition at line 1017 of file libmc.c.

References mc_platform_s::agent_queue, and agency_s::mc_platform.

Referenced by MC_FindAgentByID_chdl().

13.38.14.2 EXPORTMC MCAgent_t MC_FindAgentByName (MCAgency_t *attr*, const char * *name*)

Find an agent by its name.

Parameters:

attr a running agency

name name to search for

Returns:

a valid agent on success or NULL on failure

13.38.15 Example

Definition at line 1004 of file libmc.c.

References mc_platform_s::agent_queue, and agency_s::mc_platform.

Referenced by MC_AclSend(), MC_DeleteAgent_chdl(), MC_DeleteAgentWG_chdl(), MC_FindAgentByName_chdl(), MC_TerminateAgent_chdl(), and MC_TerminateAgentWG_chdl().

13.38.15.1 EXPORTMC void* MC_GetAgentExecEngine (MCAgent_t agent)

Retrieve an agent's Ch interpreter.

Parameters:

agent a valid agent

Returns:

a Ch interpreter of type 'ChInterp_t' on success, or NULL on failure.

Definition at line 1071 of file libmc.c.

References agent_s::agent_interp.

13.38.15.2 EXPORTMC int MC_GetAgentID (MCAgent_t agent)

Retrieve an agent's id.

Definition at line 1077 of file libmc.c.

References agent_s::id.

Referenced by MC_GetAgentID_chdl().

13.38.15.3 EXPORTMC char* MC_GetAgentName (MCAgent_t agent)

Definition at line 1085 of file libmc.c.

References agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::name.

Referenced by MC_GetAgentName_chdl().

13.38.15.4 EXPORTMC int MC_GetAgentNumTasks (MCAgent_t agent)

Retrive the number of tasks an agent has.

13.38.16 Example

Definition at line 1166 of file libmc.c.

References agent_s::datastate, and agent_datastate_s::number_of_tasks.

Referenced by MC_GetAgentNumTasks_chdl().

13.38.16.1 EXPORTMC int MC_GetAgentReturnData (MCAgent_t agent, int task_num, void ** data, int * dim, int ** extent)

Get an agent's return data.

Parameters:

agent a valid agent

task_num the task for which to retrieve the return data. The task must already be completed.

data the return data. May be multi dimensional array.

dim the number of dimensions of the return array.

extent the extent of each one of the array dimensions.

13.38.17 Example

This file demonstrates the retrieval of agent return data from an agent

This is the agent which gets the data

Definition at line 1103 of file libmc.c.

References agent_task_s::agent_return_data, interpreter_variable_data_s::array_dim, interpreter_variable_data_s::array_extent, CH_DATATYPE_SIZE, interpreter_variable_data_s::data_type, agent_s::datastate, agent_datastate_s::number_of_tasks, size, and agent_datastate_s::tasks.

13.38.17.1 EXPORTMC int MC_GetAgentStatus (MCAgent_t agent)

Get an agent's current status.

Returns:

returns type 'enum MC_AgentStatus_e'

Definition at line 1049 of file libmc.c.

References agent_s::agent_status, agent_s::lock, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_GetAgentStatus_chdl().

13.38.17.2 EXPORTMC enum MC_AgentType_e MC_GetAgentType (MCAgent_t *agent*)

Get an agent's type.

Returns:

returns type 'enum MC_AgentType_e'

Definition at line 1172 of file libmc.c.

References agent_s::agent_type.

13.38.17.3 EXPORTMC char* MC_GetAgentXMLString (MCAgent_t *agent*)

Get an agent's xml string.

Returns:

a malloc'd character string containing the agent's xml code

Definition at line 1059 of file libmc.c.

References agent_s::datastate, mxmlSaveAllocString(), and agent_datastate_s::xml_agent_root.

Referenced by MC_GetAgentXMLString_chdl().

13.38.17.4 EXPORTMC int MC_HaltAgency (MCAgency_t *agency*)

Halt an agency: Do not process new entries in queues.

Parameters:

agency A handle to a running MobileC agency.

Returns:

0 on success, non-zero on failure.

Definition at line 1218 of file libmc.c.

References mc_platform_s::giant, mc_platform_s::giant_lock, agency_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_GetAllAgents(), and MC_HaltAgency_chdl().

13.38.17.5 EXPORTMC MCAgency_t MC_Initialize (int *port*, MCAgencyOptions_t * *options*)

Initialize and start a MobileC agency.

Parameters:

port the TCP port the agency should bind to

options initialized MobileC [options](#) or NULL for default [options](#)

Returns:

a handle to a running MobileC agency or NULL on failure

13.38.18 Example

Definition at line 1227 of file libmc.c.

References `mc_platform_s::agency`, `buf`, `MCAgencyOptions_s::ch_options`, `agency_s::client`, `MCAgencyOptions_s::default_agent_status`, `agency_s::default_agentstatus`, `f`, `HOST_NAME_MAX`, `agency_s::hostName`, `MCAgencyOptions_s::initInterps`, `agency_s::initInterps`, `agency_s::known_host_filename`, `MCAgencyOptions_s::known_host_filename`, `MC_InitializeAgencyOptions()`, `mc_platform`, `agency_s::mc_platform`, `mc_platform_Initialize()`, `MC_THREAD_ALL`, `MCAgencyOptions_s::passphrase`, `agency_s::portno`, `agency_s::priv_key_filename`, `MCAgencyOptions_s::priv_key_filename`, `read_encrypted_file()`, `agency_s::server`, `MCAgencyOptions_s::stack_size`, `agency_s::stack_size`, `MCAgencyOptions_s::threads`, and `agency_s::threads`.

13.38.18.1 EXPORTMC int MC_InitializeAgencyOptions (struct MCAgencyOptions_s * *options*)

Initialize MobileC [options](#).

Parameters:

[options](#) [options](#) to initialize.

Returns:

0 on success, `error_code_t` on failure

Note:

MobileC [options](#) should be initialized with this function before any of its members are modified.

13.38.19 Example

Definition at line 1335 of file libmc.c.

References `MCAgencyOptions_s::default_agent_status`, `MCAgencyOptions_s::initInterps`, `MCAgencyOptions_s::known_host_filename`, `MC_THREAD_ALL`, `MC_WAIT_CH`, `MCAgencyOptions_s::modified`, `MCAgencyOptions_s::passphrase`, `MCAgencyOptions_s::priv_key_filename`, `MCAgencyOptions_s::stack_size`, and `MCAgencyOptions_s::threads`.

Referenced by `MC_Initialize()`.

13.38.19.1 EXPORTMC int MC_LoadAgentFromFile (MCAgency_t *attr*, const char * *filename*)

Load an agent from a file into an agency.

Parameters:

agency A valid and running Mobile-C agency

filename Filename containing the agent to load

Returns:

0 on success, non-zero on failure.

Definition at line 1356 of file libmc.c.

References `buf`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `mc_platform_s::message_queue`, `MXML_DESCEND`, `mxmlFindElement()`, `mxmlLoadString()`, `message_s::to_address`, `message_s::xml_payload`, and `message_s::xml_root`.

13.38.19.2 EXPORTMC int MC_MainLoop (MCAgency_t *attr*)

Wait indefinitely.

Note:

This function is intended to block the calling thread forever.

Definition at line 2010 of file libmc.c.

References `COND_WAIT`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_cond`, and `mc_platform_s::quit_lock`.

13.38.19.3 EXPORTMC int MC_MigrateAgent (MCAgent_t *agent*, const char * *hostname*, int *port*)

Migrates a running agent to another host.

Parameters:

agent The agent to migrate

hostname The new host to migrate the agent to

port The new port to migrate the agent to

Returns:

0 on success, `error_code_t` type on failure.

Definition at line 1426 of file libmc.c.

References `agent_s::datastate`, `agent_datastate_s::progress_modifier`, `agent_task_s::server_name`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_MigrateAgent_chdl()`.

13.38.19.4 EXPORTMC int MC_MutexLock (MCAgency_t *attr*, int *id*)

Locks a MobileC synchronization variable as a mutex.

Parameters:

attr a MobileC agency handle

id the synchronization variable id to lock

Returns:

0 on success, `error_code_t` type on failure

13.38.20 Example

Consider the following agents, which use the agent-space version of this api function. Note that the 'sleep' agent is sent first, followed by the 'wake' agent.

Definition at line 1448 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_LOCK, mc_platform_s::syncList, and syncListFind().

Referenced by MC_MutexLock_chdl().

13.38.20.1 EXPORTMC int MC_MutexUnlock (MCAgency_t attr, int id)

Definition at line 1460 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_UNLOCK, mc_platform_s::syncList, and syncListFind().

Referenced by MC_MutexUnlock_chdl().

13.38.20.2 EXPORTMC int MC_PrintAgentCode (MCAgent_t agent)

Prints an agents code to stdout.

Returns:

0 on success, error_code_t on failure

Definition at line 1472 of file libmc.c.

References agent_datastate_s::agent_code, agent_s::datastate, agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, agent_datastate_s::number_of_tasks, and agent_datastate_s::task_progress.

Referenced by MC_PrintAgentCode_chdl().

13.38.20.3 EXPORTMC int MC_RegisterService (MCAgency_t agency, MCAgent_t agent, int agentID, const char * agentName, char ** serviceNames, int numServices)

Register a new service with the Directory Facilitator.

Parameters:

agency a MobileC agency handle

agent (OPTIONAL: See note) a MobileC agent

agentID (OPTIONAL: See note) a MobileC agent id

agentName (OPTIONAL: See note) a MobileC agent name

serviceNames an array of character strings of service names

numServices the number of services described in 'serviceNames'

Returns:

0 on success, error_code_t type on failure

Note:

Three of the input arguments are optional. The function expects as input the arguments 'agent XOR (agentID AND agentName)'.

13.38.21 Example

Definition at line 1488 of file libmc.c.

References CHECK_NULL, mc_platform_s::df, df_AddRequest(), df_request_list_node_New(), agent_s::id, MC_ERR_INVALID_ARGS, MC_ERR_MEMORY, agency_s::mc_platform, MUTEX_INIT, MUTEX_T, and agent_s::name.

Referenced by MC_RegisterService_chdl().

13.38.21.1 EXPORTMC int MC_ResetSignal (MCAgency_t attr)

Reset a MobileC signal.

Returns:

0 on success, error_code_t on failure

See also:

[MC_WaitSignal\(\)](#)

Definition at line 1620 of file libmc.c.

References COND_SIGNAL, mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, MC_NO_SIGNAL, agency_s::mc_platform, mc_platform_s::MC_signal, MUTEX_LOCK, and MUTEX_UNLOCK.

13.38.21.2 EXPORTMC int MC_ResumeAgency (MCAgency_t agency)

Resumes a halted agency.

Parameters:

agency An agency previously halted with the [MC_HaltAgency\(\)](#) function.

Returns:

0 on success, non-zero on failure.

Definition at line 1567 of file libmc.c.

References mc_platform_s::giant, mc_platform_s::giant_lock, agency_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_GetAllAgents(), and MC_ResumeAgency_chdl().

13.38.21.3 EXPORTMC MCAgent_t MC_RetrieveAgent (MCAgency_t attr)

Retrieves the oldest agent from an agency.

Returns:

a valid agent or NULL on failure

Definition at line 1576 of file libmc.c.

References mc_platform_s::agent_queue, agent_s::agent_status, ListSearch(), MC_AGENT_NEUTRAL, agency_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_RetrieveAgent_chdl().

13.38.21.4 EXPORTMC char* MC_RetrieveAgentCode (MCAgent_t agent)

Retrieves an agent's Ch code.

Returns:

a malloc'd character string on success, NULL on failure

Definition at line 1604 of file libmc.c.

References agent_datastate_s::agent_code, buf, agent_s::datastate, agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and agent_datastate_s::task_progress.

Referenced by MC_RetrieveAgentCode_chdl().

13.38.21.5 EXPORTMC int MC_SearchForService (MCAgency_t attr, const char * searchString, char *** agentNames, char *** serviceNames, int ** agentIDs, int * numResults)

Search the directory facilitator for a service.

Returns:

0 on success, error_code_t on failure

Parameters:

attr (input) a MobileC agency handle

searchString (input) substring to search services for

agentNames (return) array of agent names with matching services

serviceNames (return) array of matching service names

agentIDs (return) array of matching agent IDs

numResults (return) number of matching results

13.38.22 Example

Definition at line 1631 of file libmc.c.

References CHECK_NULL, COND_SLEEP_ACTION, mc_platform_s::df, df_AddRequest(), df_request_list_node_Destroy(), df_request_list_node_New(), df_request_search_Destroy(), df_request_search_New(), MC_ERR_MEMORY, agency_s::mc_platform, MC_SUCCESS, and search.

Referenced by MC_SearchForService_chdl().

13.38.22.1 EXPORTMC int MC_SemaphorePost (MCAgency_t *attr*, int *id*)

Post to a MobileC synchronization variable semaphore.

Parameters:

attr a MobileC agency handle

id the synchronization variable id to post to

Returns:

0 on success, error_code_t type on failure

13.38.23 Example

Definition at line 1681 of file libmc.c.

References agency_s::mc_platform, syncListNode_s::sem, SEMAPHORE_POST, mc_platform_s::syncList, and syncListFind().

Referenced by MC_SemaphorePost_chdl().

13.38.23.1 EXPORTMC int MC_SemaphoreWait (MCAgency_t *attr*, int *id*)

Decreases a MobileC synchronization variable semaphore count by one.

Parameters:

attr a MobileC agency handle

id synchronization variable id to wait on

Returns:

0 on MC_SUCCESS, error_code_t type of failure

Note:

If the semaphore count is already zero, this function will block until another thread posts to the semaphore.

13.38.24 Example

Definition at line 1693 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_WAIT`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_SemaphoreWait_chdl()`.

13.38.24.1 EXPORTMC int MC_SendAgentMigrationMessage (MCAgency_t *attr*, const char * *message*, const char * *hostname*, int *port*)

Sends an agent migration message.

Parameters:

attr a MobileC agency handle
message a valid MobileC xml agent migration message
hostname host to send the message to
port port to send the message to

Definition at line 1715 of file `libmc.c`.

References `MC_ERR`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, and `mc_platform_s::message_queue`.

Referenced by `MC_SendAgentMigrationMessage_chdl()`.

13.38.24.2 EXPORTMC int MC_SendAgentMigrationMessageFile (MCAgency_t *attr*, const char * *filename*, const char * *hostname*, int *port*)

Sends an agent migration message.

Parameters:

attr a MobileC agency handle
filename file containing a valid MobileC xml agent migration message
hostname hostname to send the agent to
port port to send the agent to

Definition at line 1746 of file `libmc.c`.

References `agent_Initialize()`, `mc_platform_s::agent_queue`, `agent_s::agent_status`, `mc_platform_s::ams`, `buf`, `COND_BROADCAST`, `agency_s::mc_platform`, `MC_WAIT_MESSGSEND`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `handler_SEND()`, and `MC_SendAgentMigrationMessageFile_chdl()`.

13.38.24.3 EXPORTMC int MC_SetAgentStatus (MCAgent_t *agent*, int *status*)

Set an agent's status.

Parameters:

agent a MobileC agent
status agent status of type `'enum MC_AgentStatus_e'`

Returns:

0 on success, or `error_code_t` on failure

Definition at line 1824 of file `libmc.c`.

References `agent_s::agent_status`, `mc_platform_s::ams`, `COND_SIGNAL`, `agent_s::lock`, `agent_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_s::orphan`.

Referenced by `MC_DeleteAgent()`, `MC_DeleteAgentWG()`, and `MC_SetAgentStatus_chdl()`.

13.38.24.4 EXPORTMC int MC_SetDefaultAgentStatus (MCAgency_t *agency*, enum MC_AgentStatus_e *status*)

Sets default incoming agent status.

Parameters:

agency a MobileC agency handle

status the status to set all incoming agents

Returns:

0 on success, `error_type_t` on failure

Note:

using this function will override any status the incoming agent attempts to set for itself.

Definition at line 1839 of file `libmc.c`.

References `mc_platform_s::default_agentstatus`, and `agency_s::mc_platform`.

Referenced by `MC_SetDefaultAgentStatus_chdl()`.

13.38.24.5 EXPORTMC int MC_SetThreadOff (MCAgencyOptions_t * *options*, enum MC_ThreadIndex_e *index*)

Sets a MobileC thread to "off" status.

Parameters:

options MobileC [options](#) previously initialized with [MC_InitializeAgencyOptions\(\)](#)

index the thread to set

Returns:

0 on success, `error_code_t` on failure

Note:

This function must be called before [MC_Initialize\(\)](#). Once an agency is started with `MC_Initialize`, the `MC_SetThread` functions will have no effect.

Definition at line 1866 of file `libmc.c`.

References `SET_THREAD_OFF`, and `MCAgencyOptions_s::threads`.

13.38.24.6 EXPORTMC int MC_SetThreadOn (MCAgencyOptions_t * *options*, enum MC_ThreadIndex_e *index*)

Sets a MobileC thread to "on" status.

Parameters:

options MobileC *options* previously initialized with [MC_InitializeAgencyOptions\(\)](#)
index the thread to set

Returns:

0 on success, error_code_t on failure

Note:

This function must be called before [MC_Initialize\(\)](#). Once an agency is started with MC_Initialize, the MC_SetThread functions will have no effect.

Definition at line 1849 of file libmc.c.

References SET_THREAD_ON, and MCAgencyOptions_s::threads.

13.38.24.7 EXPORTMC int MC_SetThreadsAllOff (MCAgencyOptions_t * *options*)

Set all MobileC threads to 'off' status.

Parameters:

options a MobileC *options* structure initialized with with the [MC_InitializeAgencyOptions\(\)](#) function.

Returns:

0 on success, error code on failure.

Definition at line 1873 of file libmc.c.

References MC_THREAD_ALL, SET_THREAD_OFF, and MCAgencyOptions_s::threads.

13.38.24.8 EXPORTMC int MC_SetThreadsAllOn (MCAgencyOptions_t * *options*)

Set all Mobile-C threads on.

Parameters:

options MobileC *options* structure, initialized with [MC_InitializeAgencyOptions\(\)](#)

Returns:

0 on success, error code on failure.

Definition at line 1856 of file libmc.c.

References MC_THREAD_ALL, SET_THREAD_ON, and MCAgencyOptions_s::threads.

13.38.24.9 EXPORTMC int MC_Steer (MCAgency_t *attr*, int(*) (void *data) *funcptr*, void * *arg*)

Set up a steerable algorithm.

Parameters:

attr a MobileC agency handle
funcptr a function pointer to the algorithm
arg an argument for the algorithm function

Returns:

0 on success, error_code_t on failure

Note:

The algorithm function must contain a call to MC_SteerControl in order for the algorithm to be steerable.

13.38.25 Example

Definition at line 1883 of file libmc.c.

References agency_s::mc_platform, MC_RESTART, MC_RUN, mc_platform_s::MC_steer_command, mc_platform_s::MC_steer_lock, MUTEX_LOCK, and MUTEX_UNLOCK.

13.38.25.1 EXPORTMC enum MC_SteerCommand_e MC_SteerControl (void)

The MobileC user-algorithm steering function.

Returns:

The current steering command

Note:

This function belongs inside a user's steerable algorithm.

See also:

[MC_Steer\(\)](#)

Definition at line 1902 of file libmc.c.

References COND_WAIT, mc_platform_s::MC_steer_command, mc_platform_s::MC_steer_cond, mc_platform_s::MC_steer_lock, MC_SUSPEND, MUTEX_LOCK, and MUTEX_UNLOCK.

13.38.25.2 EXPORTMC int MC_SyncDelete (MCAgency_t *attr*, int *id*)

Deletes a previously initialized synchronization variable.

Parameters:

attr a MobileC agency handle

id the sync variable id to delete

Returns:

0 on success, or `error_code_t` on failure

Definition at line 1918 of file `libmc.c`.

References `syncList_s::giant_lock`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::syncList`, `syncListFind()`, `syncListNodeDestroy()`, and `syncListRemove()`.

Referenced by `MC_SyncDelete_chdl()`.

13.38.25.3 EXPORTMC int MC_SyncInit (MCAgency_t *attr*, int *id*)

Initializes a new MobileC synchronization variable.

Parameters:

attr a MobileC agency handle

id the requested sync variable id

Returns:

new sync variable's id. May be different than the requested id.

Note:

Each synchronization variable may be used as a mutex, condition variable, or semaphore. However, it should only be used as one type of synchronization variable per instance, or undefined behaviour may result.

See also:

[MC_MutexLock\(\)](#), [MC_MutexUnlock\(\)](#), [MC_CondWait\(\)](#), [MC_CondSignal\(\)](#), [MC_CondBroadcast](#), [MC_SemaphorePost\(\)](#), [MC_SemaphoreWait\(\)](#)

Definition at line 1948 of file `libmc.c`.

References `syncList_s::giant_lock`, `syncListNode_s::id`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, `mc_platform_s::syncList`, `syncListAddNode()`, `syncListFind()`, and `syncListNodeNew()`.

Referenced by `MC_SyncInit_chdl()`.

13.38.25.4 EXPORTMC int MC_TerminateAgent (MCAgent_t *agent*)

Halt a running agent.

Returns:

0 on success, `error_code_t` on failure

Definition at line 1973 of file `libmc.c`.

References `agent_s::agent_interp`.

Referenced by `ams_ManageAgentList()`, `MC_DeleteAgent()`, and `MC_TerminateAgent_chdl()`.

13.38.25.5 EXPORTMC int MC_TerminateAgentWG (MCAgent_t *calling_agent*, MCAgent_t *agent*)

Definition at line 1983 of file libmc.c.

References agent_s::agent_interp, MC_ERR_INVALID_ARGS, and agent_s::wg_code.

Referenced by MC_DeleteAgentWG(), and MC_TerminateAgentWG_chdl().

13.38.25.6 EXPORTMC int MC_WaitAgent (MCAgency_t *attr*)

Wait indefinitely.

Note:

This function is intended to block the calling thread forever. Wait for an agent arrival event
This function blocks until an agent arrival signal is triggered, at which point it unblocks.

Definition at line 2022 of file libmc.c.

References mc_platform_s::agent_queue, COND_WAIT, mc_platform, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, and size.

13.38.25.7 EXPORTMC MCAgent_t MC_WaitRetrieveAgent (MCAgency_t *attr*)

Wait and retrieve an agent.

Returns:

a valid MobileC agent on success, or NULL on failure

Note:

This function blocks until the arrival of an agent. The agent is retrieved after it is initialized, but before it is executed.

Definition at line 2042 of file libmc.c.

References mc_platform_s::agent_queue, ListSearch(), agency_s::mc_platform, MC_RECV_AGENT, MC_WaitSignal(), MUTEX_LOCK, and MUTEX_UNLOCK.

13.38.25.8 EXPORTMC int MC_WaitSignal (MCAgency_t *attr*, int *signals*)

Wait for a MobileC signal.

Parameters:

attr a MobileC agency handle

signals a flag of signals to wait for, of type 'enum MC_Signal_e'

Returns:

0 on success, error_code_t on failure

Note:

the parameter 'signals' may be something like 'MC_RECV_MESSAGE | MC_RECV_AGENT', etc.

13.38.26 Example

Definition at line 2060 of file libmc.c.

References COND_WAIT, mc_platform_s::giant, mc_platform_s::giant_lock, agency_s::mc_platform, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_WaitRetrieveAgent().

13.39 /home/dko/Projects/mobilec/trunk/src/include/macros.h File Reference

```
#include <pthread.h>
#include <semaphore.h>
#include "config.h"
#include <errno.h>
```

Defines

- #define [GET_THREAD_MODE](#)(a, b) ((a & (1<<b)) / (1<<b))
- #define [SET_THREAD_ON](#)(a, b) a = (a | (1<<b))
- #define [SET_THREAD_OFF](#)(a, b) a = (a & (~(1<<b)))
- #define [STRUCT](#)(name, members)
- #define [SOCKET_ERROR](#)()
- #define [PTHREAD_STACK_SIZE](#) 131072
- #define [THREAD_CREATE](#)(thread_handle, function, arg)
- #define [THREAD_CANCEL](#)(thread_handle) pthread_cancel(thread_handle)
- #define [THREAD_JOIN](#)(thread_handle) pthread_join(thread_handle, NULL)
- #define [THREAD_DETACH](#)(thread_handle)
- #define [THREAD_EXIT](#)() pthread_exit(NULL)
- #define [MUTEX_T](#) pthread_mutex_t
- #define [MUTEX_INIT](#)(mutex) pthread_mutex_init(mutex, NULL)
- #define [MUTEX_DESTROY](#)(mutex) pthread_mutex_destroy(mutex)
- #define [MUTEX_LOCK](#)(mutex)
- #define [MUTEX_UNLOCK](#)(mutex) pthread_mutex_unlock(mutex)
- #define [MUTEX_NEW](#)(mutex)
- #define [COND_T](#) pthread_cond_t
- #define [COND_INIT](#)(cond) pthread_cond_init(cond, NULL)
- #define [COND_DESTROY](#)(cond) pthread_cond_destroy(cond)
- #define [COND_WAIT](#)(cond, mutex) pthread_cond_wait(cond, mutex)
- #define [COND_SLEEP](#)(cond, mutex, test)
- #define [COND_RESET](#)(cond, mutex) pthread_mutex_unlock(mutex);
- #define [COND_SLEEP_ACTION](#)(cond, mutex, action)
- #define [SIGNAL](#)(cond, mutex, action)
- #define [COND_BROADCAST](#)(cond) pthread_cond_broadcast(cond)
- #define [COND_SIGNAL](#)(cond) pthread_cond_signal(cond)
- #define [SEMAPHORE_T](#) sem_t
- #define [SEMAPHORE_INIT](#)(sem) sem_init(sem, 0, 0)
- #define [SEMAPHORE_DESTROY](#)(sem) sem_destroy(sem)
- #define [SEMAPHORE_WAIT](#)(sem) sem_wait(sem)
- #define [SEMAPHORE_POST](#)(sem) sem_post(sem)
- #define [RWLOCK_T](#) mc_rwlock_t
- #define [RWLOCK_INIT](#)(rwlock) mc_rwlock_init(rwlock)
- #define [RWLOCK_DESTROY](#)(rwlock) mc_rwlock_destroy(rwlock)
- #define [RWLOCK_RDLOCK](#)(rwlock) mc_rwlock_rdlock(rwlock)
- #define [RWLOCK_RDUNLOCK](#)(rwlock) mc_rwlock_rdunlock(rwlock)
- #define [RWLOCK_WRLOCK](#)(rwlock) mc_rwlock_wrlock(rwlock)

- #define [RWLOCK_WRUNLOCK](#)(rwlock) mc_rwlock_wrunlock(rwlock)
- #define [WAKE_QUEUE](#)(queue, action)
- #define [SLEEP_QUEUE](#)(queue)
- #define [SLEEP_RESET](#)(queue) pthread_mutex_unlock(queue->thread_mutex)
- #define [CHECK_NULL](#)(var, action)
- #define [WARN](#)(message)
- #define [CH_DATATYPE_SIZE](#)(type, size)
- #define [CH_DATATYPE_STRING](#)(type, string)
- #define [CH_DATATYPE_VALUE_STRING](#)(type, string, p)
- #define [CH_STRING_DATATYPE](#)(string, type)
- #define [CH_DATATYPE_STR_TO_VAL](#)(type, string, val)

13.39.1 Define Documentation

13.39.1.1 #define CH_DATATYPE_SIZE(type, size)

Value:

```

switch(type) {
    case CH_CHARTYPE:
        size = sizeof(char);
        break;
    case CH_INTTYPE:
        size = sizeof(int);
        break;
    case CH_UINTTYPE:
        size = sizeof(unsigned int);
        break;
    case CH_SHORTTYPE:
        size = sizeof(short);
        break;
    case CH_USHORTTYPE:
        size = sizeof(unsigned short);
        break;
    case CH_FLOATTYPE:
        size = sizeof(float);
        break;
    case CH_DOUBLETTYPE:
        size = sizeof(double);
        break;
    default:
        fprintf(stderr, "Unknown data type: %d at %s:%d",
            type, __FILE__, __LINE__);
        size=0;
}

```

Definition at line 491 of file macros.h.

Referenced by [agent_AddPersistentVariable\(\)](#), [agent_xml_compose__create_row_nodes\(\)](#), [agent_xml_parse__data\(\)](#), [agent_xml_parse__fill_row_data\(\)](#), [agent_xml_parse__row\(\)](#), [interpreter_variable_data_Initialize\(\)](#), [interpreter_variable_data_InitializeFromAgent\(\)](#), and [MC_GetAgentReturnData\(\)](#).

13.39.1.2 #define CH_DATATYPE_STR_TO_VAL(type, string, val)

Value:

```

switch (type) { \
    case CH_INTTYPE: \

```

```

        *(int*)val = atoi(string); \
break; \
case CH_UINTTYPE: \
        *(unsigned int*)val = atoi(string); \
break; \
case CH_SHORTTYPE: \
        *(short*)val = (short)atoi(string); /*FIXME*/ \
break; \
case CH_USHORTTYPE: \
        *(unsigned short*)val = (unsigned short)atoi(string); /*F
        IXME*/ \
break; \
case CH_FLOATTYPE: \
        *(float*)val = strtod(string, NULL); \
break; \
case CH_DOUBLETTYPE: \
        *(double*)val = strtod(string, NULL); \
break; \
default: \
        fprintf(stderr, \
                "Unsupported data type: %d %s:%d\n", \
                type, __FILE__, __LINE__ ); \
}

```

Definition at line 605 of file macros.h.

Referenced by agent_xml_parse__data().

13.39.1.3 #define CH_DATATYPE_STRING(type, string)

Value:

```

switch(type) {
case CH_CHARTYPE:
    strcpy(string, "char");
break;
case CH_INTTYPE:
    strcpy(string, "int");
break;
case CH_UINTTYPE:
    strcpy(string, "unsigned int");
break;
case CH_SHORTTYPE:
    strcpy(string, "short");
break;
case CH_USHORTTYPE:
    strcpy(string, "unsigned short");
break;
case CH_FLOATTYPE:
    strcpy(string, "float");
break;
case CH_DOUBLETTYPE:
    strcpy(string, "double");
break;
default:
    fprintf(stderr,
            "Unsupported data type: %d %s:%d\n",
            type, __FILE__, __LINE__ );
}

```

Definition at line 522 of file macros.h.

Referenced by agent_xml_compose__data().

13.39.1.4 #define CH_DATATYPE_VALUE_STRING(type, string, p)**Value:**

```

switch(type) {
    case CH_CHARTYPE:
        sprintf(string, "%c", *((char*)p));
        break;
    case CH_INTTYPE:
        sprintf(string, "%d", *((int*)p));
        break;
    case CH_UINTTYPE:
        sprintf(string, "%d", *((unsigned int*)p));
        break;
    case CH_SHORTTYPE:
        sprintf(string, "%d", *((short*)p));
        break;
    case CH_USHORTTYPE:
        sprintf(string, "%d", *((unsigned short*)p));
        break;
    case CH_FLOATTYPE:
        sprintf(string, "%f", *((float*)p));
        break;
    case CH_DOUBLETTYPE:
        sprintf(string, "%f", *((double*)p));
        break;
    default:
        fprintf(stderr,
            "Unsupported data type: %d %s:%d\n",
            type, __FILE__, __LINE__);
}

```

Definition at line 554 of file macros.h.

Referenced by agent_xml_compose__create_row_nodes(), and agent_xml_compose__data().

13.39.1.5 #define CH_STRING_DATATYPE(string, type)**Value:**

```

if (!strcmp(string, "int")) {
    type = CH_INTTYPE;
} else if (!strcmp(string, "float")) {
    type = CH_FLOATTYPE;
} else if (!strcmp(string, "double")) {
    type = CH_DOUBLETTYPE;
} else if (!strcmp(string, "unsigned int")) {
    type = CH_UINTTYPE;
} else if (!strcmp(string, "short")) {
    type = CH_SHORTTYPE;
} else if (!strcmp(string, "unsigned short")) {
    type = CH_USHORTTYPE;
} else if (!strcmp(string, "char")) {
    type = CH_CHARTYPE;
} else {
    fprintf(stderr,
        "Unsupported data type: %d %s:%d\n",
        type, __FILE__, __LINE__);
}

```

Definition at line 583 of file macros.h.

Referenced by agent_xml_parse__data().

13.39.1.6 #define CHECK_NULL(var, action)**Value:**

```

if ( var == NULL ) {
    fprintf(stderr, "Pointer var is null: expected otherwise.\n");
    fprintf(stderr, "Error occured at %s:%d", __FILE__, __LINE__);
    action;
}

```

Definition at line 474 of file macros.h.

Referenced by agent_AddPersistentVariable(), agent_datastate_New(), agent_xml_parse_home(), agent_xml_parse_name(), agent_xml_parse_owner(), agent_xml_parse_sender(), agent_xml_parse_task(), ams_Initialize(), barrier_node_Initialize(), barrier_queue_New(), df_request_list_New(), df_request_list_node_New(), df_request_search_New(), fipa_agent_identifier_Parse(), fipa_word_Parse(), http_ParseExpression(), interpreter_variable_data_Initialize(), interpreter_variable_data_InitializeFromAgent(), interpreter_variable_data_New(), MC_AclSend_chdl(), MC_AddAgent_chdl(), MC_Barrier_chdl(), MC_BarrierDelete_chdl(), MC_BarrierInit_chdl(), MC_CondBroadcast_chdl(), MC_CondReset_chdl(), MC_CondSignal_chdl(), MC_CondWait_chdl(), MC_DeleteAgent(), MC_DeleteAgentWG(), MC_DeregisterService_chdl(), MC_End_chdl(), MC_FindAgentByID_chdl(), MC_FindAgentByName_chdl(), MC_HaltAgency_chdl(), MC_MutexLock_chdl(), MC_MutexUnlock_chdl(), mc_platform_Initialize(), MC_RegisterService(), MC_RegisterService_chdl(), MC_ResumeAgency_chdl(), MC_RetrieveAgent_chdl(), mc_rwlock_init(), MC_SearchForService(), MC_SearchForService_chdl(), MC_SemaphorePost_chdl(), MC_SemaphoreWait_chdl(), MC_SendAgentMigrationMessage_chdl(), MC_SendSteerCommand_chdl(), MC_SetDefaultAgentStatus_chdl(), MC_SyncDelete_chdl(), MC_SyncInit_chdl(), message_InitializeFromAgent(), message_InitializeFromConnection(), message_InitializeFromString(), message_New(), message_queue_SendOutgoing(), message_send_Thread(), message_xml_parse_message(), mtp_http_InitializeFromConnection(), mtp_http_New(), syncListNodeInit(), syncListNodeNew(), xml_get_cdata(), xml_get_text(), and xml_new_cdata().

13.39.1.7 #define COND_BROADCAST(cond) pthread_cond_broadcast(cond)

Definition at line 226 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), ams_Thread(), df_Thread(), handler_QUIT(), listen_Thread(), MC_Barrier(), MC_CondBroadcast(), MC_End(), MC_SendAgentMigrationMessageFile(), and MC_SendSteerCommand().

13.39.1.8 #define COND_DESTROY(cond) pthread_cond_destroy(cond)

Definition at line 202 of file macros.h.

Referenced by ams_Destroy(), barrier_node_Destroy(), df_Destroy(), df_request_list_node_Destroy(), df_request_search_Destroy(), mc_platform_Destroy(), mc_rwlock_destroy(), and syncListNodeDestroy().

13.39.1.9 #define COND_INIT(cond) pthread_cond_init(cond, NULL)

Definition at line 199 of file macros.h.

Referenced by acc_Initialize(), ams_Initialize(), barrier_node_Initialize(), df_Initialize(), df_request_list_New(), df_request_list_node_New(), df_request_search_New(), mc_platform_Initialize(), mc_rwlock_init(), syncListNodeInit(), and syncListNodeNew().

13.39.1.10 #define COND_RESET(cond, mutex) pthread_mutex_unlock(mutex);

Definition at line 214 of file macros.h.

13.39.1.11 #define COND_SIGNAL(cond) pthread_cond_signal(cond)

Definition at line 228 of file macros.h.

Referenced by agent_RunChScriptThread(), AP_QUEUE_STD_DEFN_TEMPLATE(), MC_AddAgent(), MC_CondSignal(), MC_End(), MC_ResetSignal(), mc_rwlock_rdunlock(), mc_rwlock_wrunlock(), and MC_SetAgentStatus().

13.39.1.12 #define COND_SLEEP(cond, mutex, test)

Value:

```
if (pthread_mutex_lock( mutex )) \
    printf("pthread lock error: %s:%d\n", __FILE__, __LINE__); \
if (!test) { \
    pthread_cond_wait( cond, mutex ); \
}
```

Definition at line 208 of file macros.h.

13.39.1.13 #define COND_SLEEP_ACTION(cond, mutex, action)

Value:

```
if (pthread_mutex_lock( mutex )) \
    printf("pthread lock error: %s:%d\n", __FILE__, __LINE__); \
action; \
pthread_cond_wait( cond, mutex );
```

Definition at line 216 of file macros.h.

Referenced by MC_SearchForService().

13.39.1.14 #define COND_T pthread_cond_t

Definition at line 197 of file macros.h.

Referenced by acc_Initialize(), ams_Initialize(), barrier_node_Initialize(), df_Initialize(), df_request_list_New(), df_request_list_node_New(), df_request_search_New(), mc_platform_Initialize(), mc_rwlock_init(), syncListNodeInit(), and syncListNodeNew().

13.39.1.15 #define COND_WAIT(cond, mutex) pthread_cond_wait(cond, mutex)

Definition at line 205 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_mailbox_WaitRetrieve(), ams_Thread(), df_Thread(), MC_Barrier(), MC_CondWait(), MC_MainLoop(), mc_platform_Initialize(), mc_rwlock_rdlock(), mc_rwlock_wrlock(), MC_SteerControl(), MC_WaitAgent(), MC_WaitSignal(), and message_Send().

13.39.1.16 #define GET_THREAD_MODE(a, b) ((a & (1<<b)) / (1<<b))

Definition at line 112 of file macros.h.

Referenced by MC_End(), and mc_platform_Initialize().

13.39.1.17 #define MUTEX_DESTROY(mutex) pthread_mutex_destroy(mutex)

Definition at line 180 of file macros.h.

Referenced by agent_Destroy(), agent_Initialize(), ams_Destroy(), barrier_node_Destroy(), df_Destroy(), df_request_list_node_Destroy(), df_request_search_Destroy(), mc_platform_Destroy(), mc_rwlock_destroy(), and syncListNodeDestroy().

13.39.1.18 #define MUTEX_INIT(mutex) pthread_mutex_init(mutex, NULL)

Definition at line 177 of file macros.h.

Referenced by acc_Initialize(), agent_Copy(), agent_Initialize(), agent_New(), agent_NewBinary(), ams_Initialize(), barrier_node_Initialize(), df_Initialize(), df_request_list_New(), df_request_list_node_New(), df_request_search_New(), mc_platform_Initialize(), MC_RegisterService(), mc_rwlock_init(), syncListInit(), syncListNodeInit(), and syncListNodeNew().

13.39.1.19 #define MUTEX_LOCK(mutex)

Value:

```
if (pthread_mutex_lock( mutex )) \
    fprintf(stderr, "pthread lock error: %s:%d\n", __FILE__, __LINE__)
```

Definition at line 183 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_Copy(), agent_Destroy(), agent_mailbox_WaitRetrieve(), agent_queue_Flush(), agent_RunChScriptThread(), ams_ManageAgentList(), ams_Print(), ams_Thread(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), df_Destroy(), df_node_Destroy(), df_request_list_Pop(), df_SearchForService(), df_Thread(), handler_QUIT(), interpreter_variable_data_Initialize(), listen_Thread(), MC_AddAgent(), MC_Barrier(), MC_CallAgentFunc(), MC_CallAgentFuncArg(), MC_CallAgentFuncV(), MC_CallAgentFuncVar(), MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), MC_End(), MC_GetAgentName(), MC_GetAgentStatus(), MC_GetAllAgents(), MC_HaltAgency(), MC_MainLoop(), MC_MutexLock(), mc_platform_Initialize(), MC_PrintAgentCode(), MC_ResetSignal(), MC_ResumeAgency(), MC_RetrieveAgent(), MC_RetrieveAgentCode(), mc_rwlock_rdlock(), mc_rwlock_rdunlock(), mc_rwlock_wrlock(), mc_rwlock_wrunlock(), MC_SendAgentMigrationMessageFile(), MC_SendSteerCommand(), MC_SetAgentStatus(), MC_Steer(), MC_SteerControl(), MC_SyncDelete(), MC_SyncInit(), MC_WaitAgent(), MC_WaitRetrieveAgent(), MC_WaitSignal(), message_queue_SendOutgoing(), message_Send(), and request_handler_DEREGISTER().

13.39.1.20 #define MUTEX_NEW(mutex)

Value:

```
mutex = (pthread_mutex_t*)malloc(sizeof(pthread_mutex_t)); \
    if (mutex == NULL) \
        fprintf(stderr, "Memory Error. %s:%d\n", __FILE__, __LINE__); \
```

Definition at line 188 of file macros.h.

Referenced by agent_New().

13.39.1.21 **#define MUTEX_T pthread_mutex_t**

Definition at line 175 of file macros.h.

Referenced by acc_Initialize(), agent_Copy(), agent_Initialize(), agent_NewBinary(), ams_Initialize(), barrier_node_Initialize(), df_Initialize(), df_request_list_New(), df_request_list_node_New(), df_request_search_New(), mc_platform_Initialize(), MC_RegisterService(), mc_rwlock_init(), syncListInit(), syncListNodeInit(), and syncListNodeNew().

13.39.1.22 **#define MUTEX_UNLOCK(mutex) pthread_mutex_unlock(mutex)**

Definition at line 186 of file macros.h.

Referenced by acc_MessageHandlerThread(), acc_Thread(), agent_mailbox_WaitRetrieve(), agent_queue_Flush(), agent_RunChScriptThread(), ams_ManageAgentList(), ams_Print(), ams_Thread(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), df_request_list_Pop(), df_SearchForService(), df_Thread(), handler_QUIT(), interpreter_variable_data_Initialize(), listen_Thread(), MC_AddAgent(), MC_Barrier(), MC_CallAgentFunc(), MC_CallAgentFuncArg(), MC_CallAgentFuncV(), MC_CallAgentFuncVar(), MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), MC_End(), MC_GetAgentName(), MC_GetAgentStatus(), MC_GetAllAgents(), MC_HaltAgency(), MC_MainLoop(), MC_MutexUnlock(), mc_platform_Initialize(), MC_PrintAgentCode(), MC_ResetSignal(), MC_ResumeAgency(), MC_RetrieveAgent(), MC_RetrieveAgentCode(), mc_rwlock_rdlock(), mc_rwlock_rdunlock(), mc_rwlock_wrlock(), mc_rwlock_wrunlock(), MC_SendAgentMigrationMessageFile(), MC_SendSteerCommand(), MC_SetAgentStatus(), MC_Steer(), MC_SteerControl(), MC_SyncDelete(), MC_SyncInit(), MC_WaitAgent(), MC_WaitRetrieveAgent(), MC_WaitSignal(), message_queue_SendOutgoing(), message_Send(), and request_handler_DEREGISTER().

13.39.1.23 **#define PTHREAD_STACK_SIZE 131072**

Definition at line 139 of file macros.h.

13.39.1.24 **#define RWLOCK_DESTROY(rwlock) mc_rwlock_destroy(rwlock)**

Definition at line 270 of file macros.h.

Referenced by barrier_queue_Destroy().

13.39.1.25 **#define RWLOCK_INIT(rwlock) mc_rwlock_init(rwlock)**

Definition at line 262 of file macros.h.

Referenced by barrier_queue_New(), and syncListInit().

13.39.1.26 **#define RWLOCK_RDLOCK(rwlock) mc_rwlock_rdlock(rwlock)**

Definition at line 288 of file macros.h.

Referenced by barrier_queue_Get(), and syncListFind().

13.39.1.27 #define RWLOCK_RDUNLOCK(rwlock) mc_rwlock_rdunlock(rwlock)

Definition at line 290 of file macros.h.

Referenced by barrier_queue_Get(), and syncListFind().

13.39.1.28 #define RWLOCK_T mc_rwlock_t

Definition at line 255 of file macros.h.

Referenced by barrier_queue_New(), and syncListInit().

13.39.1.29 #define RWLOCK_WRLOCK(rwlock) mc_rwlock_wrlock(rwlock)

Definition at line 292 of file macros.h.

Referenced by barrier_queue_Add(), barrier_queue_Delete(), syncListAddNode(), syncListDelete(), and syncListRemove().

13.39.1.30 #define RWLOCK_WRUNLOCK(rwlock) mc_rwlock_wrunlock(rwlock)

Definition at line 294 of file macros.h.

Referenced by barrier_queue_Add(), barrier_queue_Delete(), syncListAddNode(), syncListDelete(), and syncListRemove().

13.39.1.31 #define SEMAPHORE_DESTROY(sem) sem_destroy(sem)

Definition at line 240 of file macros.h.

Referenced by syncListNodeDestroy().

13.39.1.32 #define SEMAPHORE_INIT(sem) sem_init(sem, 0, 0)

Definition at line 237 of file macros.h.

Referenced by syncListNodeInit(), and syncListNodeNew().

13.39.1.33 #define SEMAPHORE_POST(sem) sem_post(sem)

Definition at line 245 of file macros.h.

Referenced by MC_SemaphorePost().

13.39.1.34 #define SEMAPHORE_T sem_t

Definition at line 235 of file macros.h.

Referenced by syncListNodeInit(), and syncListNodeNew().

13.39.1.35 #define SEMAPHORE_WAIT(sem) sem_wait(sem)

Definition at line 243 of file macros.h.

Referenced by MC_SemaphoreWait().

13.39.1.36 #define SET_THREAD_OFF(a, b) a = (a & ~(1<b))

Definition at line 116 of file macros.h.

Referenced by MC_SetThreadOff(), and MC_SetThreadsAllOff().

13.39.1.37 #define SET_THREAD_ON(a, b) a = (a | (1<b))

Definition at line 115 of file macros.h.

Referenced by MC_SetThreadOn(), and MC_SetThreadsAllOn().

13.39.1.38 #define SIGNAL(cond, mutex, action)

Value:

```
pthread_mutex_lock( mutex ); \
action; \
pthread_cond_signal( cond ); \
pthread_mutex_unlock( mutex )
```

Definition at line 221 of file macros.h.

Referenced by agent_RunChScriptThread(), df_Add(), df_AddRequest(), and request_handler_SEARCH().

13.39.1.39 #define SLEEP_QUEUE(queue)

Value:

```
if (pthread_mutex_lock( queue->thread_mutex )) \
printf("pthread lock error: %s:%d\n", __FILE__, __LINE__); \
pthread_cond_wait( queue->touched_signal, queue->thread_mutex )
```

Definition at line 308 of file macros.h.

13.39.1.40 #define SLEEP_RESET(queue) pthread_mutex_unlock(queue->thread_mutex)

Definition at line 312 of file macros.h.

13.39.1.41 #define SOCKET_ERROR()

Value:

```
printf("Socket error. %s:%d\nerrno:%d", __FILE__, __LINE__, errno); \
sleep(500)
```

Definition at line 132 of file macros.h.

Referenced by listen_Thread(), mc_platform_Destroy(), message_InitializeFromConnection(), message_send_Thread(), mtp_http_InitializeFromConnection(), net_bind(), and net_connect().

13.39.1.42 #define STRUCT(name, members)

Value:

```
typedef struct name##_s { \
    members \
} name##_t; \
typedef name##_t* name##_p;
```

Definition at line 120 of file macros.h.

13.39.1.43 #define THREAD_CANCEL(thread_handle) pthread_cancel(thread_handle)

Definition at line 156 of file macros.h.

Referenced by MC_End().

13.39.1.44 #define THREAD_CREATE(thread_handle, function, arg)

Value:

```
while(pthread_create( \
    thread_handle, \
    &attr, \
    function, \
    (void*) arg \
) < 0) { \
    printf("pthread_create failed. Trying again...\n"); \
    usleep(100000); \
}
```

Definition at line 145 of file macros.h.

Referenced by acc_Start(), acc_Thread(), agent_RunChScript(), ams_Start(), cmd_prompt_Start(), df_Start(), MC_AddStationaryAgent(), and message_Send().

13.39.1.45 #define THREAD_DETACH(thread_handle)

Value:

```
if(pthread_detach(thread_handle) < 0) { \
    printf("pthread_detach failed. %s:%d\n", __FILE__, __LINE__); \
}
```

Definition at line 162 of file macros.h.

Referenced by acc_Thread(), and message_Send().

13.39.1.46 #define THREAD_EXIT() pthread_exit(NULL)

Definition at line 167 of file macros.h.

Referenced by `acc_MessageHandlerThread()`, `acc_Thread()`, `ams_Thread()`, `df_Thread()`, and `listen_Thread()`.

13.39.1.47 `#define THREAD_JOIN(thread_handle) pthread_join(thread_handle, NULL)`

Definition at line 159 of file `macros.h`.

Referenced by `MC_End()`.

13.39.1.48 `#define WAKE_QUEUE(queue, action)`

Value:

```
if (pthread_mutex_trylock( queue->lock ) == 0) {    \
    action;                                          \
    pthread_cond_signal( queue->cond);              \
    pthread_mutex_unlock( queue->lock);              \
}
```

Definition at line 302 of file `macros.h`.

13.39.1.49 `#define WARN(message)`

Value:

```
fprintf(stderr, "WARNING: "); \
    fprintf(stderr, message ); \
    fprintf(stderr, " %s:%d\n", __FILE__, __LINE__ )
```

Definition at line 481 of file `macros.h`.

Referenced by `message_InitializeFromAgent()`.

13.40 /home/dko/Projects/mobilec/trunk/src/include/mc_error.h File Reference

Typedefs

- typedef enum [error_code_e](#) [error_code_t](#)

Enumerations

- enum [error_code_e](#) {
 [MC_SUCCESS](#) = 0, [MC_ERR](#), [MC_ERR_CONNECT](#), [MC_ERR_PARSE](#),
 [MC_ERR_EMPTY](#), [MC_ERR_INVALID](#), [MC_ERR_INVALID_ARGS](#), [MC_ERR_NOT_FOUND](#),
 [MC_ERR_MEMORY](#), [MC_ERR_SEND](#), [MC_WARN_DUPLICATE](#), [MC_SUCCESS](#) = 0,
 [ERR](#), [MC_ERR_CONNECT](#), [MC_ERR_PARSE](#), [MC_ERR_EMPTY](#),
 [MC_ERR_INVALID](#), [MC_ERR_INVALID_ARGS](#), [MC_ERR_NOT_FOUND](#), [MC_ERR_MEMORY](#),
 [MC_ERR_SEND](#), [MC_WARN_DUPLICATE](#) }

13.40.1 Typedef Documentation

13.40.1.1 typedef enum [error_code_e](#) [error_code_t](#)

13.40.2 Enumeration Type Documentation

13.40.2.1 enum [error_code_e](#)

Enumerator:

[MC_SUCCESS](#)
[MC_ERR](#)
[MC_ERR_CONNECT](#)
[MC_ERR_PARSE](#)
[MC_ERR_EMPTY](#)
[MC_ERR_INVALID](#)
[MC_ERR_INVALID_ARGS](#)
[MC_ERR_NOT_FOUND](#)
[MC_ERR_MEMORY](#)
[MC_ERR_SEND](#)
[MC_WARN_DUPLICATE](#)
[MC_SUCCESS](#)
[ERR](#)
[MC_ERR_CONNECT](#)
[MC_ERR_PARSE](#)
[MC_ERR_EMPTY](#)

MC_ERR_INVALID
MC_ERR_INVALID_ARGS
MC_ERR_NOT_FOUND
MC_ERR_MEMORY
MC_ERR_SEND
MC_WARN_DUPLICATE

Definition at line 41 of file mc_error.h.

13.41 /home/dko/Projects/mobilec/trunk/src/include/mc_platform.h File Reference

```
#include "acc.h"
#include "ams.h"
#include "barrier.h"
#include "cmd_prompt.h"
#include "config.h"
#include "connection.h"
#include "data_structures.h"
#include "df.h"
#include "libmc.h"
#include "../mc_sync/sync_list.h"
#include "../security/asm.h"
```

Data Structures

- struct [mc_platform_s](#)

Typedefs

- typedef struct [mc_platform_s](#) [mc_platform_t](#)
- typedef [mc_platform_t](#) * [mc_platform_p](#)

Functions

- [mc_platform_p](#) [mc_platform_Initialize](#) ([MCAgency_t](#) agency, [ChOptions_t](#) *ch_options)
- [int](#) [mc_platform_Destroy](#) ([mc_platform_p](#) mc_platform)

13.41.1 Typedef Documentation

13.41.1.1 typedef [mc_platform_t](#)* [mc_platform_p](#)

Definition at line 121 of file [mc_platform.h](#).

13.41.1.2 typedef struct [mc_platform_s](#) [mc_platform_t](#)

Definition at line 120 of file [mc_platform.h](#).

13.41.2 Function Documentation

13.41.2.1 [int](#) [mc_platform_Destroy](#) ([mc_platform_p](#) mc_platform)

Definition at line 244 of file [mc_platform.c](#).

References `mc_platform_s::acc`, `acc_Destroy()`, `mc_platform_s::agent_queue`, `mc_platform_s::ams`, `ams_Destroy()`, `mc_platform_s::barrier_queue`, `barrier_queue_Destroy()`, `mc_platform_s::cmd_prompt`, `cmd_prompt_Destroy()`, `COND_DESTROY`, `mc_platform_s::connection_queue`, `mc_platform_s::df`, `df_Destroy()`, `mc_platform_s::giant_cond`, `mc_platform_s::giant_lock`, `mc_platform_s::interp_options`, `mc_platform_s::MC_signal_cond`, `mc_platform_s::MC_signal_lock`, `mc_platform_s::MC_steer_cond`, `mc_platform_s::MC_steer_lock`, `MC_SUCCESS`, `mc_platform_s::MC_sync_cond`, `mc_platform_s::MC_sync_lock`, `mc_platform_s::message_queue`, `MUTEX_DESTROY`, `mc_platform_s::quit_lock`, `SOCKET_ERROR`, and `mc_platform_s::sockfd`.

Referenced by `MC_End()`.

13.41.2.2 `mc_platform_p mc_platform_Initialize (MCAgency_t agency, ChOptions_t * ch_options)`

Definition at line 53 of file `mc_platform.c`.

References `acc_Initialize()`, `acc_Start()`, `agent_ChScriptInitVar()`, `ams_Initialize()`, `ams_Start()`, `barrier_queue_New()`, `CHECK_NULL`, `cmd_prompt_Initialize()`, `cmd_prompt_Start()`, `COND_INIT`, `COND_T`, `COND_WAIT`, `agency_s::default_agentstatus`, `DEFAULT_HOSTNAME_LENGTH`, `df_Initialize()`, `df_Start()`, `GET_THREAD_MODE`, `agency_s::initInterps`, `agency_s::last_error`, `MC_ERR_MEMORY`, `MC_NO_SIGNAL`, `mc_platform`, `MC_THREAD_ACC`, `MC_THREAD_ALL`, `MC_THREAD_AMS`, `MC_THREAD_CP`, `MC_THREAD_DF`, `MUTEX_INIT`, `MUTEX_LOCK`, `MUTEX_T`, `MUTEX_UNLOCK`, `agency_s::portno`, `agency_s::stack_size`, `syncListInit()`, and `agency_s::threads`.

Referenced by `MC_Initialize()`.

13.42 /home/dko/Projects/mobilec/trunk/src/include/mc_rwlock.h File Reference

```
#include "macros.h"
```

Data Structures

- struct [mc_rwlock_s](#)

Typedefs

- typedef struct [mc_rwlock_s](#) [mc_rwlock_t](#)
- typedef [mc_rwlock_t](#) * [mc_rwlock_p](#)

Functions

- [int mc_rwlock_init](#) ([mc_rwlock_p](#) *rwlock*)
- [int mc_rwlock_destroy](#) ([mc_rwlock_p](#) *rwlock*)
- [int mc_rwlock_rdlock](#) ([mc_rwlock_p](#) *rwlock*)
- [int mc_rwlock_rdunlock](#) ([mc_rwlock_p](#) *rwlock*)
- [int mc_rwlock_wrlock](#) ([mc_rwlock_p](#) *rwlock*)
- [int mc_rwlock_wrunlock](#) ([mc_rwlock_p](#) *rwlock*)

13.42.1 Typedef Documentation

13.42.1.1 typedef [mc_rwlock_t](#)* [mc_rwlock_p](#)

Definition at line 46 of file [mc_rwlock.h](#).

13.42.1.2 typedef struct [mc_rwlock_s](#) [mc_rwlock_t](#)

13.42.2 Function Documentation

13.42.2.1 [int mc_rwlock_destroy](#) ([mc_rwlock_p](#) *rwlock*)

Definition at line 66 of file [mc_rwlock.c](#).

References [mc_rwlock_s::cond](#), [COND_DESTROY](#), [mc_rwlock_s::lock](#), and [MUTEX_DESTROY](#).

13.42.2.2 [int mc_rwlock_init](#) ([mc_rwlock_p](#) *rwlock*)

Definition at line 46 of file [mc_rwlock.c](#).

References [CHECK_NULL](#), [mc_rwlock_s::cond](#), [COND_INIT](#), [COND_T](#), [mc_rwlock_s::lock](#), [MC_ERR_MEMORY](#), [MUTEX_INIT](#), [MUTEX_T](#), [mc_rwlock_s::num_readers](#), [mc_rwlock_s::write_flag](#), and [mc_rwlock_s::write_request](#).

13.42.2.3 int mc_rwlock_rdlock (mc_rwlock_p *rwlock*)

Definition at line 79 of file mc_rwlock.c.

References mc_rwlock_s::cond, COND_WAIT, mc_rwlock_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, mc_rwlock_s::num_readers, mc_rwlock_s::write_flag, and mc_rwlock_s::write_request.

13.42.2.4 int mc_rwlock_rdunlock (mc_rwlock_p *rwlock*)

Definition at line 94 of file mc_rwlock.c.

References mc_rwlock_s::cond, COND_SIGNAL, mc_rwlock_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and mc_rwlock_s::num_readers.

13.42.2.5 int mc_rwlock_wrlock (mc_rwlock_p *rwlock*)

Definition at line 107 of file mc_rwlock.c.

References mc_rwlock_s::cond, COND_WAIT, mc_rwlock_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, mc_rwlock_s::num_readers, mc_rwlock_s::write_flag, and mc_rwlock_s::write_request.

13.42.2.6 int mc_rwlock_wrunlock (mc_rwlock_p *rwlock*)

Definition at line 126 of file mc_rwlock.c.

References mc_rwlock_s::cond, COND_SIGNAL, mc_rwlock_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and mc_rwlock_s::write_flag.

13.43 /home/dko/Projects/mobilec/trunk/src/include/message.h File Reference

```
#include "config.h"
#include "mtp_http.h"
#include <mxml.h>
#include <netinet/in.h>
#include "security/interface.h"
```

Data Structures

- struct [message_s](#)
- struct [message_send_arg_s](#)

Typedefs

- typedef enum [message_type_e](#) [message_type_t](#)
- typedef struct [message_s](#) [message_t](#)
- typedef [message_t](#) * [message_p](#)
- typedef struct [message_send_arg_s](#) [message_send_arg_t](#)

Enumerations

- enum [message_type_e](#) {
 [RELAY](#), [REQUEST](#), [SUBSCRIBE](#), [CANCEL](#),
 [N_UNDRSTD](#), [MOBILE_AGENT](#), [QUER_IF](#), [QUER_REF](#),
 [AGENT_UPDATE](#), [RETURN_MSG](#), [FIPA_ACL](#), [ENCRYPTED_DATA](#),
 [ENCRYPTION_INITIALIZE](#), [REQUEST_ENCRYPTION_INITIALIZE](#), [NUM_MESSAGE_-](#)
 [TYPE](#), [MSG_SEC](#) }

Functions

- [message_p](#) [message_New](#) (void)
- [message_p](#) [message_Copy](#) ([message_p](#) src)
- int [message_InitializeFromAgent](#) (struct [mc_platform_s](#) *[mc_platform](#), [message_p](#) message, struct [agent_s](#) *[agent](#))
- int [message_InitializeFromConnection](#) (struct [mc_platform_s](#) *[mc_platform](#), [message_p](#) message, struct [connection_s](#) *[connection](#))
- int [message_InitializeFromString](#) (struct [mc_platform_s](#) *[mc_platform](#), [message_p](#) message, const char *[string](#), const char *[destination_host](#), int [destination_port](#), const char *[target](#))
- int [message_Destroy](#) ([message_p](#) message)
- int [auth_rece_send_msg](#) (int [sockfd](#), char *[hostname](#), char *[message](#), char *[privkey](#), char *[known_-](#)
 [host_filename](#))
- int [message_Send](#) (struct [mc_platform_s](#) *[mc_platform](#), [message_p](#) message, char *[privatekey](#))
- void * [message_send_Thread](#) (void *[arg](#))
- int [http_to_hostport](#) (const char *[http_str](#), char **[host](#), int *[port](#), char **[target](#))

13.43.1 Typedef Documentation

13.43.1.1 typedef message_t* message_p

Definition at line 114 of file message.h.

13.43.1.2 typedef struct message_send_arg_s message_send_arg_t

13.43.1.3 typedef struct message_s message_t

13.43.1.4 typedef enum message_type_e message_type_t

13.43.2 Enumeration Type Documentation

13.43.2.1 enum message_type_e

Enumerator:

RELAY
REQUEST
SUBSCRIBE
CANCEL
N_UNDRSTD
MOBILE_AGENT
QUER_IF
QUER_REF
AGENT_UPDATE
RETURN_MSG
FIPA_ACL
ENCRYPTED_DATA
ENCRYPTION_INITIALIZE
REQUEST_ENCRYPTION_INITIALIZE
NUM_MESSAGE_TYPE
MSG_SEC

Definition at line 51 of file message.h.

13.43.3 Function Documentation

13.43.3.1 int auth_rece_send_msg (int sockfd, char * hostname, char * message, char * privkey, char * known_host_filename)

Definition at line 437 of file message.c.

References `aes_en_de()`, `initiate_migration_process()`, `read_known_host_file()`, and `send_AES_en_MA()`.

Referenced by `message_send_Thread()`.

13.43.3.2 int http_to_hostport (const char * *http_str*, char ** *host*, int * *port*, char ** *target*)

Definition at line 287 of file message.c.

References MC_ERR_PARSE.

Referenced by MC_AclSend().

13.43.3.3 message_p message_Copy (message_p *src*)

Definition at line 88 of file message.c.

13.43.3.4 int message_Destroy (message_p *message*)

Definition at line 398 of file message.c.

References message_s::addr, message_s::agent_xml_flag, message_s::from_address, MC_SUCCESS, message_s::message_body, mxmlDelete(), message_s::target, message_s::to_address, message_s::update_name, and message_s::xml_root.

Referenced by acc_connection_Thread(), acc_MessageHandlerThread(), ams_ManageAgentList(), MC_LoadAgentFromFile(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessageFile(), message_InitializeFromConnection(), and message_InitializeFromString().

13.43.3.5 int message_InitializeFromAgent (struct mc_platform_s * *mc_platform*, message_p *message*, struct agent_s * *agent*)

Referenced by ams_ManageAgentList().

13.43.3.6 int message_InitializeFromConnection (struct mc_platform_s * *mc_platform*, message_p *message*, struct connection_s * *connection*)**13.43.3.7 int message_InitializeFromString (struct mc_platform_s * *mc_platform*, message_p *message*, const char * *string*, const char * *destination_host*, int *destination_port*, const char * *target*)**

Referenced by MC_LoadAgentFromFile(), MC_SendAgentMigrationMessage(), and MC_SendAgentMigrationMessageFile().

13.43.3.8 message_p message_New (void)

Definition at line 64 of file message.c.

References message_s::addr, message_s::agent_xml_flag, CHECK_NULL, message_s::connect_id, message_s::from_address, message_s::http_type, message_s::isHTTP, message_s::message_body, message_s::message_id, message_s::message_type, message_s::target, message_s::to_address, message_s::update_name, message_s::update_num, message_s::xml_payload, and message_s::xml_root.

Referenced by acc_connection_Thread(), ams_ManageAgentList(), MC_LoadAgentFromFile(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessageFile(), and mtp_http_CreateMessage().

13.43.3.9 `int message_Send (struct mc_platform_s * mc_platform, message_p message, char * privatekey)`

Definition at line 562 of file message.c.

References `mc_platform_s::acc`, `COND_WAIT`, `message_send_arg_s::mc_platform`, `message_send_arg_s::message`, `message_send_Thread()`, `MSG_THREADS`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `message_send_arg_s::privatekey`, `THREAD_CREATE`, `THREAD_DETACH`, and `THREAD_T`.

Referenced by `acc_MessageHandlerThread()`, `MC_AclSend()`, and `message_queue_SendOutgoing()`.

13.43.3.10 `void* message_send_Thread (void * arg)`

Definition at line 601 of file message.c.

References `mc_platform_s::agency`, `auth_rece_send_msg()`, `buf`, `CHECK_NULL`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `agency_s::known_host_filename`, `mc_platform`, `dynstring_s::message`, `message_s::message_body`, `MSG_THREAD_EXIT`, `mtp_http_ComposeMessage()`, `mtp_http_Destroy()`, `mtp_http_New()`, `mtp_http_Parse()`, `port`, `send`, `SOCKET_ERROR`, `SOCKET_INPUT_SIZE`, `strtok_r`, and `message_s::to_address`.

Referenced by `message_Send()`.

13.44 /home/dko/Projects/mobilec/trunk/src/include/mobilec.h File Reference

```
#include <dynstring.h>
#include <fipa_acl.h>
#include <fipa_comm.h>
```

13.45 /home/dko/Projects/mobilec/trunk/src/include/mtp_http.h File Reference

Data Structures

- struct [mtp_http_content_s](#)
- struct [mtp_http_s](#)

Defines

- #define [SOCKET_INPUT_SIZE](#) 4096

Typedefs

- typedef struct [mtp_http_content_s](#) [mtp_http_content_t](#)
- typedef struct [mtp_http_s](#) [mtp_http_t](#)
- typedef [mtp_http_t](#) * [mtp_http_p](#)

Enumerations

- enum [http_status_code_e](#) {
 [CONTINUE](#) = 100, [SWITCHING_PROTOCOLS](#), [PROCESSING](#), [OK](#) = 200,
 [CREATED](#), [ACCEPTED](#), [NON_AUTHORITATIVE_INFORMATION](#), [NO_CONTENT](#),
 [RESET_CONTENT](#), [PARTIAL_CONTENT](#), [MULTI_STATUS](#), [BAD_REQUEST](#) = 400,
 [UNAUTHORIZED](#), [PAYMENT_REQUIRED](#), [FORBIDDEN](#), [NOT_FOUND](#),
 [METHOD_NOT_ALLOWED](#), [NOT_ACCEPTABLE](#), [PROXY_AUTHENTICATION_REQUIRED](#), [REQUEST_TIMEOUT](#),
 [CONFLICT](#), [GONE](#), [LENGTH_REQUIRED](#), [PRECONDITION_FAILED](#),
 [REQUEST_ENTITY_TOO_LARGE](#), [REQUEST_URI_TOO_LONG](#), [UNSUPPORTED_MEDIA_TYPE](#), [REQUESTED_RANGE_NOT_SATISFIABLE](#),
 [EXPECTATION_FAILED](#), [UNPROCESSABLE_ENTITY](#), [LOCKED](#), [FAILED_DEPENDANCY](#),
 [UNORDERED_COLLECTION](#), [UPGRADE_REQUIRED](#), [RETRY_WITH](#) }
 http return status codes
- enum [http_performative_e](#) {
 [HTTP_PERFORMATIVE_UNDEF](#) = -1, [HTTP_PERFORMATIVE_ZERO](#) = 0, [HTTP_HEAD](#),
 [HTTP_GET](#),
 [HTTP_POST](#), [HTTP_PUT](#), [HTTP_DELETE](#), [HTTP_TRACE](#),
 [HTTP_OPTIONS](#), [HTTP_CONNECT](#), [HTTP_RESPONSE](#), [HTTP_NUM_PERFORMATIVES](#) }
 http 'verbs'

Functions

- const char * [http_GetExpression](#) (const char *string, char **expr)
Parse an html expression.
- int [http_ParseExpression](#) (const char *expression_string, char **name, char **value)
Parse an expression into its name and value.
- const char * [http_ParseRequest](#) (mtp_http_p http, const char *string)
- const char * [http_GetToken](#) (const char *string, char **token)
- int [mtp_http_Destroy](#) (mtp_http_p http)
- int [mtp_http_InitializeFromConnection](#) (struct mtp_http_s *http, struct connection_s *connection, char *privatekey)
- mtp_http_p [mtp_http_New](#) (void)
- const char * [mtp_http_ParseHeader](#) (struct mtp_http_s *http, const char *string)
- int [mtp_http_Parse](#) (struct mtp_http_s *http, const char *string)
- int [mtp_http_ComposeMessage](#) (struct message_s *message)
- struct message_s * [mtp_http_CreateMessage](#) (mtp_http_t *mtp_http, char *hostname, int port)

13.45.1 Define Documentation

13.45.1.1 #define SOCKET_INPUT_SIZE 4096

Definition at line 147 of file mtp_http.h.

Referenced by `message_InitializeFromConnection()`, `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

13.45.2 Typedef Documentation

13.45.2.1 typedef struct mtp_http_content_s mtp_http_content_t

13.45.2.2 typedef mtp_http_t* mtp_http_p

Definition at line 145 of file mtp_http.h.

13.45.2.3 typedef struct mtp_http_s mtp_http_t

13.45.3 Enumeration Type Documentation

13.45.3.1 enum http_performative_e

http 'verbs'

Enumerator:

HTTP_PERFORMATIVE_UNDEF
HTTP_PERFORMATIVE_ZERO
HTTP_HEAD
HTTP_GET

HTTP_POST
HTTP_PUT
HTTP_DELETE
HTTP_TRACE
HTTP_OPTIONS
HTTP_CONNECT
HTTP_RESPONSE
HTTP_NUM_PERFORMATIVES

Definition at line 90 of file mtp_http.h.

13.45.3.2 enum http_status_code_e

http return status codes

Enumerator:

CONTINUE
SWITCHING_PROTOCOLS
PROCESSING
OK
CREATED
ACCEPTED
NON_AUTHORITATIVE_INFORMATION
NO_CONTENT
RESET_CONTENT
PARTIAL_CONTENT
MULTI_STATUS
BAD_REQUEST
UNAUTHORIZED
PAYMENT_REQUIRED
FORBIDDEN
NOT_FOUND
METHOD_NOT_ALLOWED
NOT_ACCEPTABLE
PROXY_AUTHENTICATION_REQUIRED
REQUEST_TIMEOUT
CONFLICT
GONE
LENGTH_REQUIRED
PRECONDITION_FAILED
REQUEST_ENTITY_TOO_LARGE
REQUEST_URI_TOO_LONG

UNSUPPORTED_MEDIA_TYPE
REQUESTED_RANGE_NOT_SATISFIABLE
EXPECTATION_FAILED
UNPROCESSABLE_ENTITY
LOCKED
FAILED_DEPENDANCY
UNORDERED_COLLECTION
UPGRADE_REQUIRED
RETRY_WITH

Definition at line 45 of file mtp_http.h.

13.45.4 Function Documentation

13.45.4.1 `const char* http_GetExpression (const char * string, char ** expr)`

Parse an html expression.

Parameters:

string (input) The html block of text: Will parse the first expression pointed to by 'string'.
expr (output) The allocated expression

Returns:

A pointer to the next expression segment of the string block, or NULL.

Definition at line 347 of file mtp_http.c.

Referenced by mtp_http_Parse(), and mtp_http_ParseHeader().

13.45.4.2 `const char* http_GetToken (const char * string, char ** token)`

Definition at line 788 of file mtp_http.c.

References cur.

Referenced by http_ParseRequest().

13.45.4.3 `int http_ParseExpression (const char * expression_string, char ** name, char ** value)`

Parse an expression into its name and value.

Parameters:

expression_string (input) The expression
name (output) An allocated name string or NULL
value (output) An allocated value string or NULL

Returns:

error_code_t type

Note:

an http expression is something like 'Date: Mon, 23 May 2005 22:38:34 GMT'
 ' where 'Date' is the name and the remainder of the string is the value

Definition at line 406 of file mtp_http.c.

References CHECK_NULL, MC_ERR_PARSE, and MC_SUCCESS.

Referenced by mtp_http_Parse(), and mtp_http_ParseHeader().

13.45.4.4 const char* http_ParseRequest (mtp_http_p *http*, const char * *string*)

Definition at line 699 of file mtp_http.c.

References cur, HTTP_CONNECT, HTTP_DELETE, HTTP_GET, http_GetToken(), HTTP_HEAD, HTTP_OPTIONS, mtp_http_s::http_performative, HTTP_PERFORMATIVE_UNDEF, HTTP_POST, HTTP_PUT, HTTP_RESPONSE, HTTP_TRACE, mtp_http_s::response_code, mtp_http_s::response_string, and mtp_http_s::target.

Referenced by mtp_http_ParseHeader().

13.45.4.5 int mtp_http_ComposeMessage (struct message_s * *message*)

Referenced by message_send_Thread().

13.45.4.6 struct message_s* mtp_http_CreateMessage (mtp_http_t * *mtp_http*, char * *hostname*, int *port*) [read]

Definition at line 873 of file mtp_http.c.

References buf, mtp_http_s::content, mtp_http_content_s::content_type, mtp_http_content_s::data, dynstring_Append(), dynstring_Destroy(), dynstring_New(), mtp_http_s::host, message_s::isHTTP, dynstring_s::len, dynstring_s::message, message_s::message_body, message_New(), mtp_http_s::message_parts, PACKAGE_VERSION, mtp_http_s::target, and message_s::to_address.

Referenced by MC_AclSend().

13.45.4.7 int mtp_http_Destroy (mtp_http_p *http*)

Definition at line 54 of file mtp_http.c.

References mtp_http_s::accept_ranges, mtp_http_s::boundary, mtp_http_s::connection, mtp_http_s::content, mtp_http_s::content_length, mtp_http_content_s::content_type, mtp_http_s::content_type, mtp_http_content_s::data, mtp_http_s::date, mtp_http_s::host, mtp_http_s::http_version, mtp_http_s::message_parts, mtp_http_s::response_string, mtp_http_s::return_code, SAFE_FREE, mtp_http_s::server, mtp_http_s::target, and mtp_http_s::user_agent.

Referenced by acc_connection_Thread(), MC_AclSend(), message_send_Thread(), and mtp_http_InitializeFromConnection().

13.45.4.8 int mtp_http_InitializeFromConnection (struct mtp_http_s * *http*, struct connection_s * *connection*, char * *privatekey*)

Referenced by acc_connection_Thread().

13.45.4.9 mtp_http_p mtp_http_New (void)

Definition at line 87 of file mtp_http.c.

References CHECK_NULL, and mtp_http_s::content.

Referenced by acc_connection_Thread(), MC_AclSend(), message_send_Thread(), and mtp_http_InitializeFromConnection().

13.45.4.10 int mtp_http_Parse (struct mtp_http_s * *http*, const char * *string*)

Definition at line 549 of file mtp_http.c.

References mtp_http_s::boundary, mtp_http_s::content, mtp_http_s::content_length, mtp_http_content_s::content_type, mtp_http_s::content_type, mtp_http_content_s::data, http_GetExpression(), HTTP_HEAD, http_ParseExpression(), mtp_http_s::http_performative, HTTP_POST, HTTP_PUT, HTTP_RESPONSE, MC_SUCCESS, mtp_http_s::message_parts, and mtp_http_ParseHeader().

Referenced by message_send_Thread(), and mtp_http_InitializeFromConnection().

13.45.4.11 const char* mtp_http_ParseHeader (struct mtp_http_s * *http*, const char * *string*)

Definition at line 465 of file mtp_http.c.

References mtp_http_s::header_length, http_GetExpression(), HTTP_PARSE_EXPR, http_ParseExpression(), http_ParseRequest(), MC_SUCCESS, and SAFE_FREE.

Referenced by mtp_http_InitializeFromConnection(), and mtp_http_Parse().

13.46 /home/dko/Projects/mobilec/trunk/src/include/xml_compose.h File Reference

```
#include "agent.h"
```

Functions

- [mxml_node_t * agent_xml_compose \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__gaf_message \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__message \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__mobile_agent \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__agent_data \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__name \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__owner \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__home \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__wg_code \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__tasks \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__task \(agent_p agent, int index\)](#)
- [mxml_node_t * agent_xml_compose__data \(agent_p agent, int index, interpreter_variable_data_t *interp_variable\)](#)
- [mxml_node_t * agent_xml_compose__agent_code \(agent_p agent, int index\)](#)
- [mxml_node_t * agent_xml_compose__row \(interpreter_variable_data_t *interp_variable, int index\)](#)
- [mxml_node_t * agent_xml_compose__create_row_nodes \(void *data, int index, int *extent, ChType_t type, int dim, int extent_index\)](#)

13.46.1 Function Documentation

13.46.1.1 mxml_node_t* agent_xml_compose (agent_p agent)

Definition at line 46 of file `xml_compose.c`.

References `agent_xml_compose__gaf_message()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmlAdd()`, `mxmlLoadString()`, and `node`.

Referenced by `message_InitializeFromAgent()`.

13.46.1.2 mxml_node_t* agent_xml_compose__agent_code (agent_p agent, int index)

Definition at line 521 of file `xml_compose.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_s::datastate`, `MXML_NO_PARENT`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, and `xml_new_cdata()`.

Referenced by `agent_xml_compose__tasks()`.

13.46.1.3 mxml_node_t* agent_xml_compose__agent_data (agent_p agent)

Definition at line 150 of file `xml_compose.c`.

References `agent_xml_compose__home()`, `agent_xml_compose__name()`, `agent_xml_compose__owner()`, `agent_xml_compose__tasks()`, `agent_xml_compose__wg_code()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by agent_xml_compose__mobile_agent().

13.46.1.4 **mxml_node_t* agent_xml_compose__create_row_nodes** (void * *data*, int *index*, int * *extent*, ChType_t *type*, int *dim*, int *extent_index*)

Definition at line 572 of file xml_compose.c.

References agent_xml_compose__create_row_nodes(), buf, CH_DATATYPE_SIZE, CH_DATATYPE_VALUE_STRING, MXML_ADD_AFTER, MXML_ADD_TO_PARENT, MXML_NO_PARENT, mxm-
lAdd(), mxmlElementSetAttr(), mxmlNewElement(), mxmlNewText(), node, and size.

Referenced by agent_xml_compose__create_row_nodes(), and agent_xml_compose__row().

13.46.1.5 **mxml_node_t* agent_xml_compose__data** (agent_p *agent*, int *index*, interpreter_variable_data_t * *interp_variable*)

Definition at line 447 of file xml_compose.c.

References agent_xml_compose__row(), interpreter_variable_data_s::array_dim, buf, CH_DATATYPE_STRING, CH_DATATYPE_VALUE_STRING, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, MXML_ADD_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), interpreter_variable_data_s::name, and node.

Referenced by agent_xml_compose__task().

13.46.1.6 **mxml_node_t* agent_xml_compose__gaf_message** (agent_p *agent*)

Definition at line 66 of file xml_compose.c.

References agent_xml_compose__message(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose().

13.46.1.7 **mxml_node_t* agent_xml_compose__home** (agent_p *agent*)

Definition at line 253 of file xml_compose.c.

References agent_s::home, mxmlNewElement(), mxmlNewText(), and node.

Referenced by agent_xml_compose__agent_data().

13.46.1.8 **mxml_node_t* agent_xml_compose__message** (agent_p *agent*)

Definition at line 85 of file xml_compose.c.

References agent_s::agent_type, agent_xml_compose__mobile_agent(), MC_LOCAL_AGENT, MC_REMOTE_AGENT, MC_RETURN_AGENT, MXML_ADD_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), and node.

Referenced by agent_xml_compose__gaf_message().

13.46.1.9 **mxml_node_t* agent_xml_compose__mobile_agent** (agent_p *agent*)

Definition at line 129 of file xml_compose.c.

References `agent_xml_compose__agent_data()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__message()`.

13.46.1.10 `mxml_node_t* agent_xml_compose__name (agent_p agent)`

Definition at line 221 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `agent_s::name`, and `node`.

Referenced by `agent_xml_compose__agent_data()`.

13.46.1.11 `mxml_node_t* agent_xml_compose__owner (agent_p agent)`

Definition at line 237 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::owner`.

Referenced by `agent_xml_compose__agent_data()`.

13.46.1.12 `mxml_node_t* agent_xml_compose__row (interpreter_variable_data_t * interp_variable, int index)`

Definition at line 549 of file `xml_compose.c`.

References `agent_xml_compose__create_row_nodes()`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, and `node`.

Referenced by `agent_xml_compose__data()`.

13.46.1.13 `mxml_node_t* agent_xml_compose__task (agent_p agent, int index)`

Definition at line 342 of file `xml_compose.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_compose__data()`, `buf`, `agent_task_s::code_id`, `agent_s::datastate`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, `agent_task_s::persistent`, `agent_datastate_s::persistent`, `agent_task_s::server_name`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_xml_compose__tasks()`.

13.46.1.14 `mxml_node_t* agent_xml_compose__tasks (agent_p agent)`

Definition at line 285 of file `xml_compose.c`.

References `agent_xml_compose__agent_code()`, `agent_xml_compose__task()`, `buf`, `agent_s::datastate`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, `agent_datastate_s::number_of_tasks`, and `agent_datastate_s::task_progress`.

Referenced by `agent_xml_compose__agent_data()`.

13.46.1.15 `mxml_node_t* agent_xml_compose__wg_code (agent_p agent)`

Definition at line 269 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::wg_code`.

Referenced by `agent_xml_compose__agent_data()`.

13.47 /home/dko/Projects/mobilec/trunk/src/include/xml_helper.h

File Reference

```
#include <mxml.h>
```

Functions

- [mxml_node_t * xml_find_sibling](#) (const [mxml_node_t](#) **node*, const char **sibling_name*)
- char * [xml_get_cdata](#) (const [mxml_node_t](#) **node*)
- [mxml_node_t * xml_get_child](#) (const [mxml_node_t](#) **node*, const char **child_name*, [int](#) *descend*)
- [mxml_node_t * xml_get_deep_child](#) (const [mxml_node_t](#) **parent*, const char ***child_path*)
- [mxml_node_t * xml_get_next_element](#) (const [mxml_node_t](#) **node*)
- char * [xml_get_text](#) (const [mxml_node_t](#) **node*)
- const char * [xml_get_element_name](#) (const [mxml_node_t](#) **node*)
- [mxml_node_t * xml_new_cdata](#) ([mxml_node_t](#) **parent*, const char **text*)
- const char * [whitespace_cb](#) ([mxml_node_t](#) **node*, [int](#) *where*)

13.47.1 Function Documentation

13.47.1.1 const char * whitespace_cb (mxml_node_t * *node*, int *where*)

Definition at line 571 of file testmxml.c.

References [mxml_node_s::child](#), [mxml_value_u::element](#), [MXML_WS_AFTER_CLOSE](#), [MXML_WS_AFTER_OPEN](#), [MXML_WS_BEFORE_CLOSE](#), [MXML_WS_BEFORE_OPEN](#), [mxml_value_s::name](#), [mxml_node_s::parent](#), and [mxml_node_s::value](#).

Referenced by [main\(\)](#).

13.47.1.2 mxml_node_t* xml_find_sibling (const mxml_node_t * *node*, const char * *sibling_name*)

Definition at line 54 of file xml_helper.c.

References [MXML_NO_DESCEND](#), [mxmlFindElement\(\)](#), [mxml_node_s::parent](#), and [xml_get_element_name\(\)](#).

13.47.1.3 char* xml_get_cdata (const mxml_node_t * *node*)

Definition at line 76 of file xml_helper.c.

References [buf](#), [CHECK_NULL](#), [MXML_ELEMENT](#), [mxml_node_s::type](#), and [xml_get_element_name\(\)](#).

Referenced by [xml_get_text\(\)](#).

13.47.1.4 mxml_node_t* xml_get_child (const mxml_node_t * *node*, const char * *child_name*, int *descend*)

Definition at line 109 of file xml_helper.c.

References [mxmlFindElement\(\)](#).

Referenced by `agent_return_xml_parse()`, `agent_xml_parse__agent_data()`, `agent_xml_parse__data()`, `agent_xml_parse__mobile_agent()`, `message_xml_parse()`, `message_xml_parse__message()`, and `xml_get_deep_child()`.

13.47.1.5 `mxml_node_t* xml_get_deep_child (const mxml_node_t * parent, const char ** child_path)`

Definition at line 128 of file `xml_helper.c`.

References `MXML_NO_DESCEND`, `node`, and `xml_get_child()`.

13.47.1.6 `const char* xml_get_element_name (const mxml_node_t * node)`

Definition at line 222 of file `xml_helper.c`.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_value_s::name`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `agent_xml_parse__data()`, `agent_xml_parse__mobile_agent()`, `agent_xml_parse__row()`, `message_xml_parse()`, `xml_find_sibling()`, and `xml_get_cdata()`.

13.47.1.7 `mxml_node_t* xml_get_next_element (const mxml_node_t * node)`

Definition at line 142 of file `xml_helper.c`.

References `MXML_ELEMENT`, `mxml_node_s::next`, and `mxml_node_s::type`.

13.47.1.8 `char* xml_get_text (const mxml_node_t * node)`

Definition at line 160 of file `xml_helper.c`.

References `CHECK_NULL`, `mxml_node_s::child`, `mxml_value_u::element`, `MXML_ELEMENT`, `MXML_TEXT`, `mxml_value_s::name`, `mxml_node_s::next`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, `mxml_node_s::value`, and `xml_get_cdata()`.

Referenced by `agent_xml_parse__agent_code()`, `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__sender()`, and `agent_xml_parse__wg_code()`.

13.47.1.9 `mxml_node_t* xml_new_cdata (mxml_node_t * parent, const char * text)`

Definition at line 235 of file `xml_helper.c`.

References `CHECK_NULL`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__agent_code()`.

13.48 /home/dko/Projects/mobilec/trunk/src/include/xml_parser.h File Reference

```
#include <mxml.h>
#include "macros.h"
#include "agent.h"
```

Functions

- **STRUCT** (xml_parser, const [mxml_node_t](#) *root; const [mxml_node_t](#) *node;)
- [error_code_t agent_xml_parse](#) ([agent_p](#) agent)
- [error_code_t agent_xml_parse__mobile_agent](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__agent_data](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__name](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__owner](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__home](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__sender](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__wg_code](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__tasks](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser)
- [error_code_t agent_xml_parse__task](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser, [int](#) index)
- [error_code_t agent_xml_parse__data](#) ([agent_p](#) agent, [xml_parser_p](#) xml_parser, [int](#) index)
- [error_code_t agent_xml_parse__row](#) ([interpreter_variable_data_t](#) *interp_variable, [xml_parser_p](#) xml_parser, [int](#) index)
- void [agent_xml_parse__fill_row_data](#) (void *data, [ChType_t](#) type, [int](#) *extent, const [mxml_node_t](#) *node, [int](#) *index)
- [error_code_t agent_xml_parse__agent_code](#) ([agent_p](#) agent, [int](#) index, [xml_parser_p](#) xml_parser)
- [error_code_t message_xml_parse](#) ([message_p](#) message)
- [error_code_t message_xml_parse__message](#) ([message_p](#) message, [xml_parser_p](#) xml_parser)

13.48.1 Function Documentation

13.48.1.1 [error_code_t agent_xml_parse](#) ([agent_p](#) agent)

Definition at line 52 of file [xml_parser.c](#).

References [agent_xml_parse__mobile_agent\(\)](#), [agent_s::datastate](#), [MC_SUCCESS](#), and [agent_datastate_s::xml_agent_root](#).

Referenced by [agent_Initialize\(\)](#).

13.48.1.2 [error_code_t agent_xml_parse__agent_code](#) ([agent_p](#) agent, [int](#) index, [xml_parser_p](#) xml_parser)

Definition at line 872 of file [xml_parser.c](#).

References [agent_datastate_s::agent_code](#), [agent_datastate_s::agent_code_ids](#), [agent_datastate_s::agent_codes](#), [agent_task_s::code_id](#), [agent_s::datastate](#), [MC_SUCCESS](#), [mxmlElementGetAttr\(\)](#), [agent_datastate_s::number_of_tasks](#), [agent_datastate_s::task_progress](#), [agent_datastate_s::tasks](#), and [xml_get_text\(\)](#).

Referenced by [agent_xml_parse__tasks\(\)](#).

13.48.1.3 error_code_t agent_xml_parse__agent_data (agent_p agent, xml_parser_p xml_parser)

Definition at line 93 of file xml_parser.c.

References agent_xml_parse__home(), agent_xml_parse__name(), agent_xml_parse__owner(), agent_xml_parse__sender(), agent_xml_parse__tasks(), agent_xml_parse__wg_code(), MC_ERR_PARSE, MC_SUCCESS, and xml_get_child().

Referenced by agent_xml_parse__mobile_agent().

13.48.1.4 error_code_t agent_xml_parse__data (agent_p agent, xml_parser_p xml_parser, int index)

Definition at line 538 of file xml_parser.c.

References agent_task_s::agent_return_data, agent_task_s::agent_variable_list, agent_xml_parse__row(), interpreter_variable_data_s::array_dim, CH_DATATYPE_SIZE, CH_DATATYPE_STR_TO_VAL, CH_STRING_DATATYPE, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, agent_s::datastate, interpreter_variable_data_New(), MC_ERR_PARSE, MC_SUCCESS, mxmlElementGetAttr(), interpreter_variable_data_s::name, mxml_node_s::parent, agent_task_s::persistent, agent_datastate_s::tasks, xml_get_child(), and xml_get_element_name().

Referenced by agent_xml_parse__task().

13.48.1.5 void agent_xml_parse__fill_row_data (void * data, ChType_t type, int * extent, const mxml_node_t * node, int * index)

Definition at line 751 of file xml_parser.c.

References agent_xml_parse__fill_row_data(), buf, CH_DATATYPE_SIZE, mxml_node_s::child, MXML_DESCEND_FIRST, MXML_ELEMENT, MXML_TEXT, mxmlFindElement(), mxml_text_s::string, strtok_r, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by agent_xml_parse__fill_row_data(), and agent_xml_parse__row().

13.48.1.6 error_code_t agent_xml_parse__home (agent_p agent, xml_parser_p xml_parser)

Definition at line 219 of file xml_parser.c.

References CHECK_NULL, agent_s::home, MC_SUCCESS, and xml_get_text().

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.48.1.7 error_code_t agent_xml_parse__mobile_agent (agent_p agent, xml_parser_p xml_parser)

Definition at line 65 of file xml_parser.c.

References agent_xml_parse__agent_data(), MC_ERR_PARSE, xml_get_child(), and xml_get_element_name().

Referenced by agent_xml_parse().

13.48.1.8 error_code_t agent_xml_parse__name (agent_p agent, xml_parser_p xml_parser)

Definition at line 166 of file xml_parser.c.

References CHECK_NULL, MC_ERR_PARSE, MC_SUCCESS, agent_s::name, and xml_get_text().

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.48.1.9 error_code_t agent_xml_parse__owner (agent_p agent, xml_parser_p xml_parser)

Definition at line 192 of file xml_parser.c.

References CHECK_NULL, MC_SUCCESS, agent_s::owner, and xml_get_text().

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.48.1.10 error_code_t agent_xml_parse__row (interpreter_variable_data_t * interp_variable, xml_parser_p xml_parser, int index)

Definition at line 684 of file xml_parser.c.

References agent_xml_parse__fill_row_data(), interpreter_variable_data_s::array_dim, interpreter_variable_data_s::array_extent, CH_DATATYPE_SIZE, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, MC_SUCCESS, and xml_get_element_name().

Referenced by agent_xml_parse__data().

13.48.1.11 error_code_t agent_xml_parse__sender (agent_p agent, xml_parser_p xml_parser)

Definition at line 245 of file xml_parser.c.

References CHECK_NULL, MC_SUCCESS, agent_s::sender, and xml_get_text().

Referenced by agent_xml_parse__agent_data().

13.48.1.12 error_code_t agent_xml_parse__task (agent_p agent, xml_parser_p xml_parser, int index)

Definition at line 442 of file xml_parser.c.

References agent_xml_parse__data(), CHECK_NULL, agent_task_s::code_id, agent_s::datastate, MC_ERR_PARSE, MC_SUCCESS, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent_task_s::persistent, agent_task_s::server_name, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by agent_xml_parse__tasks().

13.48.1.13 error_code_t agent_xml_parse__tasks (agent_p agent, xml_parser_p xml_parser)

Definition at line 304 of file xml_parser.c.

References agent_datastate_s::agent_code, agent_datastate_s::agent_code_ids, agent_datastate_s::agent_codes, agent_task_New(), agent_xml_parse__agent_code(), agent_xml_parse__task(), buf, agent_s::datastate, MC_ERR_PARSE, MXML_DESCEND, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent_datastate_s::number_of_tasks, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.48.1.14 **error_code_t agent_xml_parse__wg_code (agent_p *agent*, xml_parser_p *xml_parser*)**

Definition at line 273 of file xml_parser.c.

References MC_SUCCESS, agent_s::wg_code, and xml_get_text().

Referenced by agent_xml_parse__agent_data().

13.48.1.15 **error_code_t message_xml_parse (message_p *message*)**

Definition at line 948 of file xml_parser.c.

References MC_ERR_PARSE, message_xml_parse__message(), MXML_DESCEND, MXML_NO_DESCEND, mxmlFindElement(), xml_get_child(), xml_get_element_name(), and message_s::xml_root.

Referenced by acc_connection_Thread(), message_InitializeFromConnection(), and message_InitializeFromString().

13.48.1.16 **error_code_t message_xml_parse__message (message_p *message*, xml_parser_p *xml_parser*)**

Definition at line 1003 of file xml_parser.c.

References message_s::addr, buf, CHECK_NULL, ENCRYPTED_DATA, ENCRYPTION_INITIALIZE, FIPA_ACL, message_s::from_address, MC_ERR_PARSE, MC_SUCCESS, message_s::message_type, MOBILE_AGENT, mxmlElementGetAttr(), port, REQUEST_ENCRYPTION_INITIALIZE, RETURN_MSG, strtok_r, xml_get_child(), and message_s::xml_payload.

Referenced by message_xml_parse().

13.48.1.17 **STRUCT (xml_parser, const mxml_node_t *root;const mxml_node_t *node;)**

13.49 /home/dko/Projects/mobilec/trunk/src/libmc.c File Reference

```
#include "config.h"
#include <unistd.h>
#include <pthread.h>
#include <embedch.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/time.h>
#include "include/libmc.h"
#include "include/macros.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/data_structures.h"
#include "include/fipa_acl_envelope.h"
#include "include/fipa_acl.h"
#include "include/agent.h"
#include "include/agent_task.h"
```

Defines

- #define [HOST_NAME_MAX](#) 255

Functions

- [int MC_AclDestroy](#) (struct [fipa_acl_message_s](#) *message)
Destroy a FIPA ACL message.
- [EXPORTMC fipa_acl_message_t * MC_AclNew](#) (void)
Allocate a new ACL Message.
- [EXPORTMC int MC_AclPost](#) ([MCAgent_t](#) agent, struct [fipa_acl_message_s](#) *message)
Post ACL message to agent.
- [EXPORTMC fipa_acl_message_t * MC_AclReply](#) ([fipa_acl_message_t](#) *acl_message)
Reply to an ACL message.
- [EXPORTMC fipa_acl_message_t * MC_AclRetrieve](#) ([MCAgent_t](#) agent)
Retrieve an ACL message.
- [EXPORTMC int MC_AclSend](#) ([MCAgency_t](#) attr, [fipa_acl_message_t](#) *acl)
Send a composed ACL Message.

- EXPORTMC [fipa_acl_message_t](#) * [MC_AclWaitRetrieve](#) ([MCAgent_t](#) agent)
Wait for and retrieve an ACL message.
- [int](#) [MC_AclSetProtocol](#) ([fipa_acl_message_t](#) *acl, enum [fipa_protocol_e](#) protocol)
- [int](#) [MC_AclSetConversationID](#) ([fipa_acl_message_t](#) *acl, char *id)
- [int](#) [MC_AclSetPerformative](#) ([fipa_acl_message_t](#) *acl, enum [fipa_performative_e](#) performative)
- [int](#) [MC_AclSetSender](#) ([fipa_acl_message_t](#) *acl, const char *name, const char *address)
- [int](#) [MC_AclAddReceiver](#) ([fipa_acl_message_t](#) *acl, const char *name, const char *address)
- [int](#) [MC_AclAddReplyTo](#) ([fipa_acl_message_t](#) *acl, const char *name, const char *address)
- [int](#) [MC_AclSetContent](#) ([fipa_acl_message_t](#) *acl, const char *content)
- EXPORTMC [int](#) [MC_AddAgent](#) ([MCAgency_t](#) attr, [MCAgent_t](#) agent)
Add an agent to the agency 'attr'.
- [int](#) [MC_AddStationaryAgent](#) ([MCAgency_t](#) agency, void *(*agent_thread)(struct [agent_thread_arg_s](#) *), const char *name, void *agent_args)
- const void * [MC_AgentVariableRetrieve](#) ([MCAgent_t](#) agent, const char *var_name, [int](#) task_num)
Retrieve a pointer to a previously saved variable.
- [int](#) [MC_AgentVariableRetrieveInfo](#) ([MCAgent_t](#) agent, const char *var_name, [int](#) task_num, const void **data, [int](#) *dim, const [int](#) **extent)
Retrieve a info about a previously saved variable.
- [int](#) [MC_AgentVariableSave](#) ([MCAgent_t](#) agent, const char *var_name)
Mark an agent variable for saving.
- [int](#) [MC_Barrier](#) ([MCAgency_t](#) attr, [int](#) id)
- EXPORTMC [int](#) [MC_BarrierInit](#) ([MCAgency_t](#) attr, [int](#) id, [int](#) num_procs)
Initialize a MobileC Barrier.
- EXPORTMC [int](#) [MC_BarrierDelete](#) ([MCAgency_t](#) attr, [int](#) id)
Find and delete an initialized MobileC Barrier.
- EXPORTMC [int](#) [MC_CallAgentFunc](#) ([MCAgent_t](#) agent, const char *funcName, void *returnVal, [int](#) numArgs,...)
Use custom ChOptions_t type for internal Ch interpreter.
- EXPORTMC [int](#) [MC_CallAgentFuncArg](#) ([MCAgent_t](#) agent, const char *funcName, void *returnVal, void *arg)
Calls a function defined in an agent.
- EXPORTMC [int](#) [MC_CallAgentFuncV](#) ([MCAgent_t](#) agent, const char *funcName, void *returnVal, va_list ap)
Calls a function defined in an agent.
- EXPORTMC [int](#) [MC_CallAgentFuncVar](#) ([MCAgent_t](#) agent, const char *funcName, void *returnVal, ChVaList_t varg)
- EXPORTMC [MCAgent_t](#) [MC_ComposeAgent](#) (const char *name, const char *home, const char *owner, const char *code, const char *return_var_name, const char *server, [int](#) persistent)
Compose a new agent dynamically without using a prewritten XML file.

- EXPORTMC [MCAgent_t MC_ComposeAgentS](#) (const char *name, const char *home, const char *owner, const char *code, const char *return_var_name, const char *server, [int](#) persistent, const char *workgroup_code)
Compose a new agent dynamically without using a prewritten XML file.
- EXPORTMC [MCAgent_t MC_ComposeAgentFromFile](#) (const char *name, const char *home, const char *owner, const char *filename, const char *return_var_name, const char *server, [int](#) persistent)
Compose a new agent dynamically from a source code file.
- EXPORTMC [MCAgent_t MC_ComposeAgentFromFileS](#) (const char *name, const char *home, const char *owner, const char *filename, const char *return_var_name, const char *server, [int](#) persistent, const char *workgroup_code)
Compose a new agent dynamically from a source code file.
- EXPORTMC [int MC_CondBroadcast](#) ([MCAgency_t](#) attr, [int](#) id)
Wakes up all agents/threads waiting on a condition variable.
- EXPORTMC [int MC_CondSignal](#) ([MCAgency_t](#) attr, [int](#) id)
Wakes up at least one thread waiting on a condition variable.
- EXPORTMC [int MC_CondWait](#) ([MCAgency_t](#) attr, [int](#) id)
Wait on a MobileC synchronization variable.
- EXPORTMC [int MC_CondReset](#) ([MCAgency_t](#) attr, [int](#) id)
Reset a previously signalled MobileC condition variable.
- [int MC_CopyAgent](#) ([MCAgent_t](#) *agent_out, const [MCAgent_t](#) agent_in)
Performs a deep-copy of an agent structure.
- EXPORTMC [int MC_DeleteAgent](#) ([MCAgent_t](#) agent)
Stop and remove an agent.
- EXPORTMC [int MC_DeleteAgentWG](#) ([MCAgent_t](#) calling_agent, [MCAgent_t](#) agent)
Stop and remove an agent in the same workgroup.
- [int MC_DestroyServiceSearchResult](#) (char **agentName, char **serviceName, [int](#) *agentID, [int](#) numResult)
Free memory allocated by a Service Search operation.
- [int MC_DeregisterService](#) ([MCAgency_t](#) agency, [int](#) agentID, const char *serviceName)
- EXPORTMC [int MC_End](#) ([MCAgency_t](#) agency)
End an agency.
- EXPORTMC [MCAgent_t MC_FindAgentByName](#) ([MCAgency_t](#) attr, const char *name)
Find an agent by its name.
- EXPORTMC [MCAgent_t MC_FindAgentByID](#) ([MCAgency_t](#) attr, [int](#) ID)
Find an agent by its id.
- time_t [MC_GetAgentArrivalTime](#) ([MCAgent_t](#) agent)

- EXPORTMC [int MC_GetAgentStatus](#) ([MCAgent_t](#) agent)
Get an agent's current status.
- EXPORTMC [char * MC_GetAgentXMLString](#) ([MCAgent_t](#) agent)
Get an agent's xml string.
- EXPORTMC [void * MC_GetAgentExecEngine](#) ([MCAgent_t](#) agent)
Retrieve an agent's Ch interpreter.
- EXPORTMC [int MC_GetAgentID](#) ([MCAgent_t](#) agent)
Retrieve an agent's id.
- EXPORTMC [char * MC_GetAgentName](#) ([MCAgent_t](#) agent)
- EXPORTMC [int MC_GetAgentReturnData](#) ([MCAgent_t](#) agent, [int](#) task_num, [void **data](#), [int](#) *dim, [int **extent](#))
Get an agent's return data.
- EXPORTMC [int MC_GetAgentNumTasks](#) ([MCAgent_t](#) agent)
Retrive the number of tasks an agent has.
- EXPORTMC [enum MC_AgentType_e MC_GetAgentType](#) ([MCAgent_t](#) agent)
Get an agent's type.
- [int MC_GetAllAgents](#) ([MCAgency_t](#) attr, [MCAgent_t **agents](#), [int](#) *num_agents)
- EXPORTMC [int MC_HaltAgency](#) ([MCAgency_t](#) attr)
Halt an agency: Do not process new entries in queues.
- EXPORTMC [MCAgency_t MC_Initialize](#) ([int](#) port, [MCAgencyOptions_t](#) *options)
Initialize and start a MobileC agency.
- EXPORTMC [int MC_InitializeAgencyOptions](#) ([struct MCAgencyOptions_s](#) *options)
Initialize MobileC options.
- EXPORTMC [int MC_LoadAgentFromFile](#) ([MCAgency_t](#) attr, [const char *filename](#))
Load an agent from a file into an agency.
- EXPORTMC [int MC_MigrateAgent](#) ([MCAgent_t](#) agent, [const char *hostname](#), [int](#) port)
Migrates a running agent to another host.
- EXPORTMC [int MC_MutexLock](#) ([MCAgency_t](#) attr, [int](#) id)
Locks a MobileC synchronization variable as a mutex.
- EXPORTMC [int MC_MutexUnlock](#) ([MCAgency_t](#) attr, [int](#) id)
- EXPORTMC [int MC_PrintAgentCode](#) ([MCAgent_t](#) agent)
Prints an agents code to stdout.
- EXPORTMC [int MC_RegisterService](#) ([MCAgency_t](#) agency, [MCAgent_t](#) agent, [int](#) agentID, [const char *agentName](#), [char **serviceNames](#), [int](#) numServices)
Register a new service with the Directory Facilitator.

- EXPORTMC [int MC_ResumeAgency](#) ([MCAgency_t](#) attr)
Resumes a halted agency.
- EXPORTMC [MCAgency_t MC_RetrieveAgent](#) ([MCAgency_t](#) attr)
Retrieves the oldest agent from an agency.
- EXPORTMC [char * MC_RetrieveAgentCode](#) ([MCAgent_t](#) agent)
Retrieves an agent's Ch code.
- EXPORTMC [int MC_ResetSignal](#) ([MCAgency_t](#) attr)
Reset a MobileC signal.
- EXPORTMC [int MC_SearchForService](#) ([MCAgency_t](#) attr, [const char *searchString](#), [char ***agentNames](#), [char ***serviceNames](#), [int **agentIDs](#), [int *numResults](#))
Search the directory facilitator for a service.
- EXPORTMC [int MC_SemaphorePost](#) ([MCAgency_t](#) attr, [int id](#))
Post to a MobileC synchronization variable semaphore.
- EXPORTMC [int MC_SemaphoreWait](#) ([MCAgency_t](#) attr, [int id](#))
Decreases a MobileC synchronization variable semaphore count by one.
- [int MC_SendCh](#) ([MCAgency_t](#) attr, [const char *filename](#), [const char *remotehost](#), [int port](#))
- EXPORTMC [int MC_SendAgentMigrationMessage](#) ([MCAgency_t](#) attr, [const char *string](#), [const char *hostname](#), [int port](#))
Sends an agent migration message.
- EXPORTMC [int MC_SendAgentMigrationMessageFile](#) ([MCAgency_t](#) attr, [const char *filename](#), [const char *hostname](#), [int port](#))
Sends an agent migration message.
- EXPORTMC [int MC_SendSteerCommand](#) ([MCAgency_t](#) attr, [enum MC_SteerCommand_e cmd](#))
- [int MC_SetAgentStatus](#) ([MCAgent_t](#) agent, [int status](#))
Set an agent's status.
- [int MC_SetDefaultAgentStatus](#) ([MCAgency_t](#) agency, [enum MC_AgentStatus_e status](#))
Sets default incoming agent status.
- EXPORTMC [int MC_SetThreadOn](#) ([MCAgencyOptions_t *options](#), [enum MC_ThreadIndex_e index](#))
Sets a MobileC thread to "on" status.
- EXPORTMC [int MC_SetThreadsAllOn](#) ([MCAgencyOptions_t *options](#))
Set all Mobile-C threads on.
- EXPORTMC [int MC_SetThreadOff](#) ([MCAgencyOptions_t *options](#), [enum MC_ThreadIndex_e index](#))
Sets a MobileC thread to "off" status.
- EXPORTMC [int MC_SetThreadsAllOff](#) ([MCAgencyOptions_t *options](#))

Set all MobileC threads to 'off' status.

- EXPORTMC [int MC_Steer](#) ([MCAgency_t](#) attr, [int](#)(*funcptr)(void *data), void *arg)

Set up a steerable algorithm.

- EXPORTMC enum [MC_SteerCommand_e](#) [MC_SteerControl](#) (void)

The MobileC user-algorithm steering function.

- EXPORTMC [int MC_SyncDelete](#) ([MCAgency_t](#) attr, [int](#) id)

Deletes a previously initialized synchronization variable.

- EXPORTMC [int MC_SyncInit](#) ([MCAgency_t](#) attr, [int](#) id)

Initializes a new MobileC synchronization variable.

- EXPORTMC [int MC_TerminateAgent](#) ([MCAgent_t](#) agent)

Halt a running agent.

- EXPORTMC [int MC_TerminateAgentWG](#) ([MCAgent_t](#) calling_agent, [MCAgent_t](#) agent)

- [int MC_MainLoop](#) ([MCAgency_t](#) attr)

Wait indefinitely.

- EXPORTMC [int MC_WaitAgent](#) ([MCAgency_t](#) attr)

Wait indefinitely.

- EXPORTMC [MCAgent_t MC_WaitRetrieveAgent](#) ([MCAgency_t](#) attr)

Wait and retrieve an agent.

- EXPORTMC [int MC_WaitSignal](#) ([MCAgency_t](#) attr, [int](#) signals)

Wait for a MobileC signal.

- [int MC_AclDestroy_chdl](#) (void *varg)
- void * [MC_AclNew_chdl](#) (void *varg)
- [int MC_AclPost_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclReply_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclRetrieve_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSend_chdl](#) (void *varg)
- EXPORTCH void * [MC_AclWaitRetrieve_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSetProtocol_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSetConversationID_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSetPerformative_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSetSender_chdl](#) (void *varg)
- EXPORTCH [int MC_AclAddReceiver_chdl](#) (void *varg)
- EXPORTCH [int MC_AclAddReplyTo_chdl](#) (void *varg)
- EXPORTCH [int MC_AclSetContent_chdl](#) (void *varg)
- EXPORTCH [int MC_AddAgent_chdl](#) (void *varg)
- EXPORTCH const void * [MC_AgentVariableRetrieve_chdl](#) (void *varg)
- EXPORTCH [int MC_AgentVariableSave_chdl](#) (void *varg)
- EXPORTCH [int MC_CallAgentFunc_chdl](#) (void *varg)
- EXPORTCH [int MC_Barrier_chdl](#) (void *varg)
- EXPORTCH [int MC_BarrierDelete_chdl](#) (void *varg)

- EXPORTCH [int MC_BarrierInit_chdl](#) (void *varg)
- EXPORTCH [int MC_CondBroadcast_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_ComposeAgent_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_ComposeAgentS_chdl](#) (void *varg)
- EXPORTCH [int MC_CondSignal_chdl](#) (void *varg)
- EXPORTCH [int MC_CondReset_chdl](#) (void *varg)
- EXPORTCH [int MC_CondWait_chdl](#) (void *varg)
- EXPORTCH [int MC_DeleteAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_DeleteAgentWG_chdl](#) (void *varg)
- EXPORTCH [int MC_DestroyServiceSearchResult_chdl](#) (void *varg)
- EXPORTCH [int MC_DeregisterService_chdl](#) (void *varg)
- EXPORTCH [int MC_End_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_FindAgentByID_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_FindAgentByName_chdl](#) (void *varg)
- EXPORTCH [time_t MC_GetAgentArrivalTime_chdl](#) (void *varg)
- EXPORTCH [int MC_GetAgentID_chdl](#) (void *varg)
- EXPORTCH [char * MC_GetAgentName_chdl](#) (void *varg)
- EXPORTCH [int MC_GetAgentNumTasks_chdl](#) (void *varg)
- EXPORTCH [int MC_GetAgentStatus_chdl](#) (void *varg)
- EXPORTCH [char * MC_GetAgentXMLString_chdl](#) (void *varg)
- EXPORTCH [int MC_GetTimeOfDay_chdl](#) (void *varg)
- EXPORTCH [int MC_HaltAgency_chdl](#) (void *varg)
- EXPORTCH [int MC_MigrateAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_MutexLock_chdl](#) (void *varg)
- EXPORTCH [int MC_MutexUnlock_chdl](#) (void *varg)
- EXPORTCH [int MC_PrintAgentCode_chdl](#) (void *varg)
- EXPORTCH [int MC_RegisterService_chdl](#) (void *varg)
- EXPORTCH [int MC_ResumeAgency_chdl](#) (void *varg)
- EXPORTCH [MCAgent_t MC_RetrieveAgent_chdl](#) (void *varg)
- EXPORTCH [char * MC_RetrieveAgentCode_chdl](#) (void *varg)
- EXPORTCH [int MC_SaveData_chdl](#) (void *varg)
- EXPORTCH [int MC_SearchForService_chdl](#) (void *varg)
- EXPORTCH [int MC_SemaphorePost_chdl](#) (void *varg)
- EXPORTCH [int MC_SemaphoreWait_chdl](#) (void *varg)
- EXPORTCH [int MC_SendAgentMigrationMessage_chdl](#) (void *varg)
- EXPORTCH [int MC_SendAgentMigrationMessageFile_chdl](#) (void *varg)
- EXPORTCH [int MC_SendSteerCommand_chdl](#) (void *varg)
- EXPORTCH [int MC_SetAgentStatus_chdl](#) (void *varg)
- EXPORTCH [int MC_SetDefaultAgentStatus_chdl](#) (void *varg)
- EXPORTCH [int MC_SyncDelete_chdl](#) (void *varg)
- EXPORTCH [int MC_SyncInit_chdl](#) (void *varg)
- EXPORTCH [int MC_TerminateAgent_chdl](#) (void *varg)
- EXPORTCH [int MC_TerminateAgentWG_chdl](#) (void *varg)

Variables

- [mc_platform_p g_mc_platform](#)

13.49.1 Define Documentation

13.49.1.1 #define HOST_NAME_MAX 255

Definition at line 67 of file libmc.c.

Referenced by MC_Initialize().

13.49.2 Function Documentation

13.49.2.1 int MC_AclAddReceiver (fipa_acl_message_t * *acl*, const char * *name*, const char * *address*)

Definition at line 278 of file libmc.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_New(), fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_url_New(), fipa_url_sequence_New(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, fipa_agent_identifier_set_s::num, fipa_acl_message_s::receiver, fipa_acl_message_s::receiver_num, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by MC_AclAddReceiver_chdl().

13.49.2.2 EXPORTCH int MC_AclAddReceiver_chdl (void * *varg*)

Definition at line 2271 of file libmc.c.

References MC_AclAddReceiver().

Referenced by agent_ChScriptInitVar().

13.49.2.3 int MC_AclAddReplyTo (fipa_acl_message_t * *acl*, const char * *name*, const char * *address*)

Definition at line 314 of file libmc.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_New(), fipa_agent_identifier_set_New(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_url_New(), fipa_url_sequence_New(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, fipa_agent_identifier_set_s::num, fipa_acl_message_s::reply_to, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by MC_AclAddReplyTo_chdl().

13.49.2.4 EXPORTCH int MC_AclAddReplyTo_chdl (void * *varg*)

Definition at line 2291 of file libmc.c.

References MC_AclAddReplyTo().

Referenced by agent_ChScriptInitVar().

13.49.2.5 int MC_AclDestroy (struct fipa_acl_message_s * *message*)

Destroy a FIPA ACL message.

Parameters:

message The ACL message to destroy

Returns:

0 on success, error code on failure.

Definition at line 78 of file libmc.c.

References fipa_acl_message_Destroy().

Referenced by MC_AclDestroy_chdl().

13.49.2.6 int MC_AclDestroy_chdl (void * *varg*)

Definition at line 2081 of file libmc.c.

References MC_AclDestroy().

Referenced by agent_ChScriptInitVar().

13.49.2.7 EXPORTMC fipa_acl_message_t* MC_AclNew (void) [read]

Allocate a new ACL Message.

Returns:

A newly allocated and empty ACL message.

Definition at line 84 of file libmc.c.

References fipa_acl_message_New().

Referenced by MC_AclNew_chdl().

13.49.2.8 void* MC_AclNew_chdl (void * *varg*)

Definition at line 2096 of file libmc.c.

References MC_AclNew().

Referenced by agent_ChScriptInitVar().

13.49.2.9 EXPORTMC int MC_AclPost (MCAgent_t *agent*, struct fipa_acl_message_s * *message*)

Post ACL message to agent.

Parameters:

agent The agent to post the message to

message The message to post

Returns:

0 if successful, or error_code_t type.

Definition at line 89 of file libmc.c.

References agent_mailbox_Post(), and agent_s::mailbox.

Referenced by MC_AclPost_chdl(), and MC_AclSend().

13.49.2.10 int MC_AclPost_chdl (void * *varg*)

Definition at line 2104 of file libmc.c.

References MC_AclPost().

Referenced by agent_ChScriptInitVar().

13.49.2.11 EXPORTMC fipa_acl_message_t* MC_AclReply (struct fipa_acl_message_s * *acl_message*) [read]

Reply to an ACL message.

Parameters:

acl_message The incoming acl message to reply to

Returns:

A newly allocated ACL message

Note:

This function simply generates a new ACL message with the 'receiver' field automatically set to the 'sender' field of the incoming message.

Definition at line 95 of file libmc.c.

References fipa_Reply().

Referenced by MC_AclReply_chdl().

13.49.2.12 EXPORTCH void* MC_AclReply_chdl (void * *varg*)

Definition at line 2122 of file libmc.c.

References MC_AclReply().

Referenced by agent_ChScriptInitVar().

13.49.2.13 EXPORTMC fipa_acl_message_t* MC_AclRetrieve (MCAgent_t *agent*) [read]

Retrieve an ACL message.

Parameters:

agent Agent to retrieve message from.

Returns:

an ACL message struct on success or NULL on failure

Definition at line 101 of file libmc.c.

References agent_mailbox_Retrieve(), and agent_s::mailbox.

Referenced by MC_AclRetrieve_chdl().

13.49.2.14 EXPORTCH void* MC_AclRetrieve_chdl (void * *varg*)

Definition at line 2138 of file libmc.c.

References MC_AclRetrieve().

Referenced by agent_ChScriptInitVar().

13.49.2.15 EXPORTMC int MC_AclSend (MCAgency_t *attr*, struct fipa_acl_message_s * *acl*)

Send a composed ACL Message.

Parameters:

attr An initialized and running MobileC agency

acl An allocated and fully composed ACL message.

Returns:

0 if successful, error code on failure.

Definition at line 107 of file libmc.c.

References fipa_agent_identifier_s::addresses, mtp_http_s::content, mtp_http_content_s::content_type, mtp_http_content_s::data, dynstring_Destroy(), FIPA_ACL, fipa_acl_Compose(), fipa_acl_message_Copy(), fipa_agent_identifier_set_s::fipa_agent_identifiers, fipa_envelope_Compose(), mtp_http_s::host, http_to_hostport(), MC_AclPost(), MC_FindAgentByName(), agency_s::mc_platform, dynstring_s::message, mtp_http_s::message_parts, message_Send(), message_s::message_type, mtp_http_CreateMessage(), mtp_http_Destroy(), mtp_http_New(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, fipa_agent_identifier_set_s::num, port, mc_platform_s::private_key, fipa_acl_message_s::receiver, fipa_url_s::str, mtp_http_s::target, message_s::target, and fipa_url_sequence_s::urls.

Referenced by MC_AclSend_chdl().

13.49.2.16 EXPORTCH int MC_AclSend_chdl (void * *varg*)

Definition at line 2154 of file libmc.c.

References CHECK_NULL, MC_AclSend(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.2.17 int MC_AclSetContent (fipa_acl_message_t * *acl*, const char * *content*)

Definition at line 349 of file libmc.c.

References fipa_string_s::content, fipa_acl_message_s::content, fipa_string_Destroy(), and fipa_string_New().

Referenced by MC_AclSetContent_chdl().

13.49.2.18 EXPORTCH int MC_AclSetContent_chdl (void * *varg*)

Definition at line 2311 of file libmc.c.

References fipa_acl_message_s::content, and MC_AclSetContent().

Referenced by agent_ChScriptInitVar().

13.49.2.19 int MC_AclSetConversationID (fipa_acl_message_t * *acl*, char * *id*)

Definition at line 228 of file libmc.c.

References fipa_string_s::content, fipa_expression_s::content, fipa_acl_message_s::conversation_id, FIPA_EXPR_STRING, fipa_expression_New(), fipa_string_New(), fipa_expression_s::content_u::string, and fipa_expression_s::type.

Referenced by MC_AclSetConversationID_chdl().

13.49.2.20 EXPORTCH int MC_AclSetConversationID_chdl (void * *varg*)

Definition at line 2215 of file libmc.c.

References MC_AclSetConversationID().

Referenced by agent_ChScriptInitVar().

13.49.2.21 int MC_AclSetPerformative (fipa_acl_message_t * *acl*, enum fipa_performative_e *performative*)

Definition at line 247 of file libmc.c.

References fipa_acl_message_s::performative.

Referenced by MC_AclSetPerformative_chdl().

13.49.2.22 EXPORTCH int MC_AclSetPerformative_chdl (void * *varg*)

Definition at line 2233 of file libmc.c.

References MC_AclSetPerformative(), and fipa_acl_message_s::performative.

Referenced by agent_ChScriptInitVar().

13.49.2.23 int MC_AclSetProtocol (fipa_acl_message_t * *acl*, enum fipa_protocol_e *protocol*)

Definition at line 212 of file libmc.c.

References FIPA_PROTOCOL_END, FIPA_PROTOCOL_ERROR, and fipa_acl_message_s::protocol.

Referenced by MC_AclSetProtocol_chdl().

13.49.2.24 EXPORTCH int MC_AclSetProtocol_chdl (void * *varg*)

Definition at line 2196 of file libmc.c.

References MC_AclSetProtocol(), and fipa_acl_message_s::protocol.

Referenced by agent_ChScriptInitVar().

13.49.2.25 int MC_AclSetSender (fipa_acl_message_t * *acl*, const char * *name*, const char * *address*)

Definition at line 255 of file libmc.c.

References fipa_agent_identifier_s::addresses, fipa_agent_identifier_Destroy(), fipa_agent_identifier_New(), fipa_url_New(), fipa_url_sequence_New(), fipa_agent_identifier_s::name, fipa_url_sequence_s::num, fipa_acl_message_s::sender, fipa_url_s::str, and fipa_url_sequence_s::urls.

Referenced by MC_AclSetSender_chdl().

13.49.2.26 EXPORTCH int MC_AclSetSender_chdl (void * *varg*)

Definition at line 2251 of file libmc.c.

References MC_AclSetSender().

Referenced by agent_ChScriptInitVar().

13.49.2.27 EXPORTMC fipa_acl_message_t* MC_AclWaitRetrieve (MCAgent_t *agent*) [read]

Wait for and retrieve an ACL message.

Parameters:

agent Agent to retrieve message from.

Returns:

an ACL message struct on success or NULL on failure

Definition at line 205 of file libmc.c.

References agent_mailbox_WaitRetrieve(), and agent_s::mailbox.

Referenced by MC_AclWaitRetrieve_chdl().

13.49.2.28 EXPORTCH void* MC_AclWaitRetrieve_chdl (void * *varg*)

Definition at line 2178 of file libmc.c.

References MC_AclWaitRetrieve().

Referenced by agent_ChScriptInitVar().

13.49.2.29 EXPORTMC int MC_AddAgent (MCAgency_t *attr*, MCAgent_t *agent*)

Add an agent to the agency 'attr'.

Parameters:

attr a MobileC agency

agent An initialized MobileC agent

Returns:

0 if successful, or error_code_t type

Definition at line 366 of file libmc.c.

References mc_platform_s::agent_queue, mc_platform_s::ams, COND_SIGNAL, agency_s::mc_platform, agent_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by handler_COMPOSE_SEND(), and MC_AddAgent_chdl().

13.49.2.30 EXPORTCH int MC_AddAgent_chdl (void * *varg*)

Definition at line 2331 of file libmc.c.

References CHECK_NULL, MC_AddAgent(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.2.31 int MC_AddStationaryAgent (MCAgency_t *agency*, void (*)(struct agent_thread_arg_s *) *agent_thread*, const char * *name*, void * *agent_args*)

Definition at line 380 of file libmc.c.

References agent_thread_arg_s::agent, agent_NewBinary(), mc_platform_s::agent_queue, agent_thread_arg_s::args, agent_thread_arg_s::attr, agency_s::mc_platform, agent_s::name, agent_thread_arg_s::thread, and THREAD_CREATE.

13.49.2.32 const void* MC_AgentVariableRetrieve (MCAgent_t *agent*, const char * *var_name*, int *task_num*)

Retrieve a pointer to a previously saved variable.

Parameters:

agent A MobileC agent.

var_name The name of the saved variable that has previously been saved.

task_num The previous completed task from which to retrieve the saved variable.

Returns:

A pointer to the data on success or NULL on failure.

13.49.3 Examples

The following example demonstrates usage of [MC_AgentVariableRetrieve\(\)](#) from agent space.

Definition at line 401 of file libmc.c.

References agent_task_s::agent_variable_list, interpreter_variable_data_s::data, agent_s::datastate, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by MC_AgentVariableRetrieve_chdl().

13.49.3.1 EXPORTCH const void* MC_AgentVariableRetrieve_chdl (void * *varg*)

Definition at line 2355 of file libmc.c.

References MC_AgentVariableRetrieve().

Referenced by agent_ChScriptInitVar().

13.49.3.2 int MC_AgentVariableRetrieveInfo (MCAgent_t *agent*, const char * *var_name*, int *task_num*, const void ** *data*, int * *dim*, const int ** *extent*)

Retrieve a info about a previously saved variable.

Parameters:

agent A MobileC agent.

var_name The name of the saved variable that has previously been saved.

task_num The previous completed task from which to retrieve the saved variable.

data (Output) The Variable Data

dim (Output) The dimension of the data array

extent (Output) The extents of the output array

Returns:

Error code.

Definition at line 419 of file libmc.c.

References agent_task_s::agent_variable_list, interpreter_variable_data_s::array_dim, interpreter_variable_data_s::array_extent, interpreter_variable_data_s::data, agent_s::datastate, MC_ERR_NOT_FOUND, MC_SUCCESS, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

13.49.3.3 int MC_AgentVariableSave (MCAgent_t *agent*, const char * *var_name*)

Mark an agent variable for saving.

Parameters:

agent A MobileC agent.

var_name The name of the variable to mark for saving.

Returns:

0 on success, non-zero on failure.

See also:

test1.xml

13.49.4 Examples

See agent_saved_variables_example/test1.xml for an example of usage of this api function.

13.49.5 Examples

The following example demonstrates usage of [MC_AgentVariableSave\(\)](#) from agent space.

Definition at line 440 of file libmc.c.

References `agent_s::datastate`, `MC_ERR_MEMORY`, `agent_task_s::num_saved_variables`, `agent_task_s::saved_variables`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `MC_AgentVariableSave_chdl()`.

13.49.5.1 EXPORTCH int MC_AgentVariableSave_chdl (void * *varg*)

Definition at line 2379 of file libmc.c.

References `MC_AgentVariableSave()`.

Referenced by `agent_ChScriptInitVar()`.

13.49.5.2 int MC_Barrier (MCAgency_t *attr*, int *id*)

Definition at line 461 of file libmc.c.

References `mc_platform_s::barrier_queue`, `barrier_queue_Get()`, `barrier_node_s::cond`, `COND_BROADCAST`, `COND_WAIT`, `barrier_node_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MC_SUCCESS`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `node`, `barrier_node_s::num_registered`, and `barrier_node_s::num_waiting`.

Referenced by `MC_Barrier_chdl()`.

13.49.5.3 EXPORTCH int MC_Barrier_chdl (void * *varg*)

Definition at line 2431 of file libmc.c.

References `CHECK_NULL`, `MC_Barrier()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

13.49.5.4 EXPORTMC int MC_BarrierDelete (MCAgency_t *attr*, int *id*)

Find and delete an initialized MobileC Barrier.

Parameters:

- attr* A running MobileC agency
- id* The id of the barrier node to delete

Returns:

returns 0 on success, error if the node is not found or other failure.

Definition at line 501 of file libmc.c.

References `mc_platform_s::barrier_queue`, `barrier_queue_Delete()`, and `agency_s::mc_platform`.

Referenced by `MC_BarrierDelete_chdl()`.

13.49.5.5 EXPORTCH int MC_BarrierDelete_chdl (void * *varg*)

Definition at line 2454 of file libmc.c.

References CHECK_NULL, MC_BarrierDelete(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.5.6 EXPORTMC int MC_BarrierInit (MCAgency_t *attr*, int *id*, int *num_procs*)

Initialize a MobileC Barrier.

Parameters:

attr A running MobileC agency

id The requested barrier id

num_procs The number of agents/threads/processes that will wait on the barrier

Returns:

The allocated barrier id. May differ from the requested id if it is already in use.

13.49.6 Examples

The following example demonstrates an agent which sets up an MC_Barrier.

Definition at line 487 of file libmc.c.

References barrier_node_Initialize(), mc_platform_s::barrier_queue, barrier_queue_Add(), barrier_queue_Get(), MC_ERR, agency_s::mc_platform, MC_SUCCESS, and node.

Referenced by MC_BarrierInit_chdl().

13.49.6.1 EXPORTCH int MC_BarrierInit_chdl (void * *varg*)

Definition at line 2477 of file libmc.c.

References CHECK_NULL, MC_BarrierInit(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.6.2 EXPORTMC int MC_CallAgentFunc (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, int *numArgs*, ...)

Use custom ChOptions_t type for internal Ch interpreter.

Parameters:

attr A running MobileC agency

options Initialized Ch [options](#) structure

Returns:

0 on success, error_code_t type on failure Calls a function defined in an agent

Parameters:

agent An initialized and executed MobileC agent
funcName The name of the function to call
returnVal The agent function's return value
numArgs The number of arguments supplied to the agent function
... A variable argument list to be supplied to the agent function

Returns:

0 if successful, error_code_t type on failure

13.49.7 Example

Definition at line 507 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

13.49.7.1 EXPORTCH int MC_CallAgentFunc_chdl (void * *varg*)

Definition at line 2401 of file libmc.c.

References MC_CallAgentFuncVar().

Referenced by agent_ChScriptInitVar().

13.49.7.2 EXPORTMC int MC_CallAgentFuncArg (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, void * *arg*)

Calls a function defined in an agent.

Parameters:

agent An initialized and executed MobileC agent
funcName The name of the function to call
returnVal The agent function's return value
arg The agent functions argument

Note:

The agent function must be of the form 'void* func(void* arg);'

Returns:

0 if successful, error_code_t type on failure

13.49.8 Example

Definition at line 528 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

13.49.8.1 EXPORTMC int MC_CallAgentFuncV (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, va_list *ap*)

Calls a function defined in an agent.

Parameters:

agent An initialized and executed MobileC agent
funcName The name of the function to call
returnVal The agent function's return value
ap A variable argument list

Returns:

0 if successful, error_code_t type on failure

Definition at line 547 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

13.49.8.2 EXPORTMC int MC_CallAgentFuncVar (MCAgent_t *agent*, const char * *funcName*, void * *returnVal*, ChVaList_t *varg*)

Definition at line 567 of file libmc.c.

References agent_s::agent_interp, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::run_lock.

Referenced by MC_CallAgentFunc_chdl().

13.49.8.3 EXPORTMC MCAgent_t MC_ComposeAgent (const char * *name*, const char * *home*, const char * *owner*, const char * *code*, const char * *return_var_name*, const char * *server*, int *persistent*)

Compose a new agent dynamically without using a prewritten XML file.

Parameters:

name The desired name of the new agent.
home The home of the new agent.
owner The owner of the new agent.
code The agent code
return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.
server The target destination of the agent.
persistent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 607 of file libmc.c.

References MC_ComposeAgentS().

13.49.8.4 EXPORTCH MCAgent_t MC_ComposeAgent_chdl (void * *varg*)

Definition at line 2525 of file libmc.c.

References MC_ComposeAgentS().

Referenced by agent_ChScriptInitVar().

13.49.8.5 EXPORTMC MCAgent_t MC_ComposeAgentFromFile (const char * *name*, const char * *home*, const char * *owner*, const char * *filename*, const char * *return_var_name*, const char * *server*, int *persistent*)

Compose a new agent dynamically from a source code file.

Parameters:

filename The filename of the source file

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

persistant A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 707 of file libmc.c.

References MC_ComposeAgentFromFileS().

Referenced by handler_COMPOSE_SEND().

13.49.8.6 EXPORTMC MCAgent_t MC_ComposeAgentFromFileS (const char * *name*, const char * *home*, const char * *owner*, const char * *filename*, const char * *return_var_name*, const char * *server*, int *persistent*, const char * *workgroup_code*)

Compose a new agent dynamically from a source code file.

Parameters:

filename The filename of the source file

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

workgroup_code The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

persitent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 729 of file libmc.c.

References agent_thread_arg_s::agent, and MC_ComposeAgentS().

Referenced by MC_ComposeAgentFromFile().

13.49.8.7 EXPORTMC MCAgent_t MC_ComposeAgentS (const char * name, const char * home, const char * owner, const char * code, const char * return_var_name, const char * server, int persistent, const char * workgroup_code)

Compose a new agent dynamically without using a prewritten XML file.

Parameters:

name The desired name of the new agent.

home The home of the new agent.

owner The owner of the new agent.

code The agent code

return_var_name The name of the agent's return variable. Set to "no-return" if no return variable is desired.

server The target destination of the agent.

workgroup_code The secret workgroup code of the agent. Only agents with the same workgroup code may perform certain interactions.

persitent A flag indicating whether or not the agent should be persistent. A value of '1' indicates persistence, while a value of '0' indicates default non-persistent behaviour.

Returns:

This function returns a valid MCAgent_t structure on success or NULL on failure.

Definition at line 630 of file libmc.c.

References agent_thread_arg_s::agent, agent_datastate_s::agent_code, agent_datastate_s::agent_code_ids, agent_datastate_s::agent_codes, agent_datastate_New(), agent_New(), agent_s::agent_status, agent_task_New(), agent_s::agent_type, agent_s::datastate, agent_s::home, MC_LOCAL_AGENT, MC_WAIT_MESSGSEND, agent_s::name, agent_datastate_s::number_of_tasks, agent_s::orphan, agent_s::owner, agent_datastate_s::persistent, agent_task_s::server_name, agent_datastate_s::tasks, agent_task_s::var_name, and agent_s::wg_code.

Referenced by MC_ComposeAgent(), MC_ComposeAgent_chdl(), MC_ComposeAgentFromFileS(), and MC_ComposeAgentS_chdl().

13.49.8.8 EXPORTCH MC_Agent_t MC_ComposeAgentS_chdl (void * *varg*)

Definition at line 2566 of file libmc.c.

References MC_ComposeAgentS().

Referenced by agent_ChScriptInitVar().

13.49.8.9 EXPORTMC int MC_CondBroadcast (MC_Agency_t *attr*, int *id*)

Wakes up all agents/threads waiting on a condition variable.

Parameters:

attr A MobileC agency

id Synchronization variable id to broadcast to

See also:

[MC_SyncInit\(\)](#), [MC_CondSignal\(\)](#)

Returns:

0 on success, error_code_t type on failure

Definition at line 772 of file libmc.c.

References syncListNode_s::cond, COND_BROADCAST, syncListNode_s::lock, MC_ERR_NOT_FOUND, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, syncListNode_s::signalled, mc_platform_s::syncList, and syncListFind().

Referenced by MC_CondBroadcast_chdl().

13.49.8.10 EXPORTCH int MC_CondBroadcast_chdl (void * *varg*)

Definition at line 2502 of file libmc.c.

References CHECK_NULL, MC_CondBroadcast(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.8.11 EXPORTMC int MC_CondReset (MC_Agency_t *attr*, int *id*)

Reset a previously signalled MobileC condition variable.

Parameters:

attr A MobileC Agency

id The synchronization variable id to reset

See also:

[MC_SyncInit\(\)](#)

Returns:

0 on success, error_code_t type on failure

Definition at line 824 of file libmc.c.

References `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondReset_chdl()`.

13.49.8.12 EXPORTCH int MC_CondReset_chdl (void * *varg*)

Definition at line 2632 of file libmc.c.

References `CHECK_NULL`, `MC_CondReset()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

13.49.8.13 EXPORTMC int MC_CondSignal (MCAgency_t *attr*, int *id*)

Wakes up at least one thread waiting on a condition variable.

Parameters:

attr A MobileC agency

id synchronization variable id

See also:

[MC_SyncInit\(\)](#)

Returns:

0 on success, `error_code_t` type on failure

13.49.9 Example

The following example demonstrates the agent-space version of the function, which is nearly identical to the binary space api function.

Definition at line 787 of file libmc.c.

References `syncListNode_s::cond`, `COND_SIGNAL`, `syncListNode_s::lock`, `MC_ERR_NOT_FOUND`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `syncListNode_s::signalled`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_CondSignal_chdl()`.

13.49.9.1 EXPORTCH int MC_CondSignal_chdl (void * *varg*)

Definition at line 2609 of file libmc.c.

References `CHECK_NULL`, `MC_CondSignal()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

13.49.9.2 EXPORTMC int MC_CondWait (MCAgency_t *attr*, int *id*)

Wait on a MobileC synchronization variable.

Parameters:

attr A MobileC agency
id a synchronization variable id

See also:

[MC_SyncInit\(\)](#)

Returns:

0 on success, error_code_t type on failure

13.49.10 Example

The following example demonstrates the agent-space version of this function.

Definition at line 802 of file libmc.c.

References syncListNode_s::cond, COND_WAIT, syncListNode_s::lock, MC_ERR_NOT_FOUND, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, syncListNode_s::signalled, mc_platform_s::syncList, and syncListFind().

Referenced by MC_CondWait_chdl().

13.49.10.1 EXPORTCH int MC_CondWait_chdl (void * *varg*)

Definition at line 2655 of file libmc.c.

References CHECK_NULL, MC_CondWait(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.10.2 int MC_CopyAgent (MCAgent_t * *agent_out*, const MCAgent_t *agent_in*)

Performs a deep-copy of an agent structure.

Parameters:

agent_out A pointer to the agent to copy to.
agent_in The agent to copy

Returns:

0 on success, error_code_t type on failure.

Definition at line 842 of file libmc.c.

References agent_Copy(), and MC_SUCCESS.

13.49.10.3 EXPORTMC int MC_DeleteAgent (MCAgent_t *agent*)

Stop and remove an agent.

Parameters:

agent An agent in any state (running, waiting, etc)

Returns:

0 on success, error_code_t type on failure

Definition at line 849 of file libmc.c.

References CHECK_NULL, MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_SetAgentStatus(), MC_SUCCESS, MC_TerminateAgent(), MC_WAIT_FINISHED, and agent_s::wg_code.

Referenced by MC_DeleteAgent_chdl().

13.49.10.4 EXPORTCH int MC_DeleteAgent_chdl (void * *varg*)

Definition at line 2677 of file libmc.c.

References MC_DeleteAgent(), MC_ERR_NOT_FOUND, and MC_FindAgentByName().

Referenced by agent_ChScriptInitVar().

13.49.10.5 EXPORTMC int MC_DeleteAgentWG (MCAgent_t *calling_agent*, MCAgent_t *agent*)

Stop and remove an agent in the same workgroup.

Parameters:

calling_agent The calling agent

agent An agent in any state (running, waiting, etc)

Note:

The agents must belong to the same workgroup.

Returns:

0 on success, error_code_t type on failure

Definition at line 869 of file libmc.c.

References CHECK_NULL, MC_ERR_INVALID, MC_ERR_INVALID_ARGS, MC_SetAgentStatus(), MC_SUCCESS, MC_TerminateAgentWG(), MC_WAIT_FINISHED, and agent_s::wg_code.

Referenced by MC_DeleteAgentWG_chdl().

13.49.10.6 EXPORTCH int MC_DeleteAgentWG_chdl (void * *varg*)

Definition at line 2697 of file libmc.c.

References MC_DeleteAgentWG(), MC_ERR_NOT_FOUND, and MC_FindAgentByName().

Referenced by agent_ChScriptInitVar().

13.49.10.7 int MC_DeregisterService (MCAgency_t *agency*, int *agentID*, const char * *serviceName*)

Definition at line 912 of file libmc.c.

References mc_platform_s::df, df_AddRequest(), df_request_list_node_New(), and agency_s::mc_platform.

Referenced by MC_DeregisterService_chdl().

13.49.10.8 EXPORTCH int MC_DeregisterService_chdl (void * *varg*)

Definition at line 2746 of file libmc.c.

References CHECK_NULL, MC_DeregisterService(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.10.9 int MC_DestroyServiceSearchResult (char ** *agentName*, char ** *serviceName*, int * *agentID*, int *numResult*)

Free memory allocated by a Service Search operation.

Parameters:

agentName agent names returned by a search operation.

serviceName service names return by a search operation.

agentID list of agent id's returned by a search operation.

numResult The number of hits returned by a search operation.

Returns:

0 on success, error code on failure.

Definition at line 892 of file libmc.c.

Referenced by MC_DestroyServiceSearchResult_chdl().

13.49.10.10 EXPORTCH int MC_DestroyServiceSearchResult_chdl (void * *varg*)

Definition at line 2719 of file libmc.c.

References MC_DestroyServiceSearchResult().

Referenced by agent_ChScriptInitVar().

13.49.10.11 EXPORTMC int MC_End (MCAgency_t *attr*)

End an agency.

Parameters:

attr A running agency

Returns:

0 on success, error_code_t type on failure

13.49.11 Example

Definition at line 941 of file libmc.c.

References `mc_platform_s::acc`, `mc_platform_s::ams`, `mc_platform_s::cmd_prompt`, `COND_BROADCAST`, `COND_SIGNAL`, `mc_platform_s::connection_queue`, `mc_platform_s::df`, `GET_THREAD_MODE`, `agency_s::hostName`, `agency_s::mc_platform`, `mc_platform_Destroy()`, `MC_THREAD_ACC`, `MC_THREAD_AMS`, `MC_THREAD_CP`, `MC_THREAD_DF`, `mc_platform_s::message_queue`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_cond`, `mc_platform_s::quit_lock`, `cmd_prompt_s::thread`, `THREAD_CANCEL`, `THREAD_JOIN`, and `agency_s::threads`.

Referenced by `MC_End_chdl()`.

13.49.11.1 EXPORTCH int MC_End_chdl (void * *varg*)

Definition at line 2773 of file libmc.c.

References `CHECK_NULL`, `MC_End()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

13.49.11.2 EXPORTMC MCAgent_t MC_FindAgentByID (MCAgency_t *attr*, int *ID*)

Find an agent by its id.

Parameters:

attr the agency to search

ID the id to search for

Returns:

a valid agent on success, NULL on failure

Definition at line 1017 of file libmc.c.

References `mc_platform_s::agent_queue`, and `agency_s::mc_platform`.

Referenced by `MC_FindAgentByID_chdl()`.

13.49.11.3 EXPORTCH MCAgent_t MC_FindAgentByID_chdl (void * *varg*)

Definition at line 2790 of file libmc.c.

References `CHECK_NULL`, `MC_FindAgentByID()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

13.49.11.4 EXPORTMC MCAgent_t MC_FindAgentByName (MCAgency_t *attr*, const char * *name*)

Find an agent by its name.

Parameters:

attr a running agency
name name to search for

Returns:

a valid agent on success or NULL on failure

13.49.12 Example

Definition at line 1004 of file libmc.c.

References mc_platform_s::agent_queue, and agency_s::mc_platform.

Referenced by MC_AclSend(), MC_DeleteAgent_chdl(), MC_DeleteAgentWG_chdl(), MC_FindAgentByName_chdl(), MC_TerminateAgent_chdl(), and MC_TerminateAgentWG_chdl().

13.49.12.1 EXPORTCH MCAgent_t MC_FindAgentByName_chdl (void * *varg*)

Definition at line 2813 of file libmc.c.

References CHECK_NULL, MC_FindAgentByName(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.12.2 time_t MC_GetAgentArrivalTime (MCAgent_t *agent*)

Definition at line 1034 of file libmc.c.

References agent_s::arrival_time.

Referenced by MC_GetAgentArrivalTime_chdl().

13.49.12.3 EXPORTCH time_t MC_GetAgentArrivalTime_chdl (void * *varg*)

Definition at line 2840 of file libmc.c.

References MC_GetAgentArrivalTime().

13.49.12.4 EXPORTMC void* MC_GetAgentExecEngine (MCAgent_t *agent*)

Retrieve an agent's Ch interpreter.

Parameters:

agent a valid agent

Returns:

a Ch interpreter of type 'ChInterp_t' on success, or NULL on failure.

Definition at line 1071 of file libmc.c.

References agent_s::agent_interp.

13.49.12.5 EXPORTMC int MC_GetAgentID (MCAgent_t *agent*)

Retrieve an agent's id.

Definition at line 1077 of file libmc.c.

References agent_s::id.

Referenced by MC_GetAgentID_chdl().

13.49.12.6 EXPORTCH int MC_GetAgentID_chdl (void * *varg*)

Definition at line 2860 of file libmc.c.

References MC_GetAgentID().

Referenced by agent_ChScriptInitVar().

13.49.12.7 EXPORTMC char* MC_GetAgentName (MCAgent_t *agent*)

Definition at line 1085 of file libmc.c.

References agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and agent_s::name.

Referenced by MC_GetAgentName_chdl().

13.49.12.8 EXPORTCH char* MC_GetAgentName_chdl (void * *varg*)

Definition at line 2876 of file libmc.c.

References MC_GetAgentName().

Referenced by agent_ChScriptInitVar().

13.49.12.9 EXPORTMC int MC_GetAgentNumTasks (MCAgent_t *agent*)

Retrive the number of tasks an agent has.

13.49.13 Example

Definition at line 1166 of file libmc.c.

References agent_s::datastate, and agent_datastate_s::number_of_tasks.

Referenced by MC_GetAgentNumTasks_chdl().

13.49.13.1 EXPORTCH int MC_GetAgentNumTasks_chdl (void * *varg*)

Definition at line 2892 of file libmc.c.

References MC_GetAgentNumTasks().

13.49.13.2 EXPORTMC int MC_GetAgentReturnData (MCAgent_t *agent*, int *task_num*, void ***data*, int **dim*, int ***extent*)

Get an agent's return data.

Parameters:

agent a valid agent

task_num the task for which to retrieve the return data. The task must already be completed.

data the return data. May be multi dimensional array.

dim the number of dimensions of the return array.

extent the extent of each one of the array dimensions.

13.49.14 Example

This file demonstrates the retrieval of agent return data from an agent

This is the agent which gets the data

Definition at line 1103 of file libmc.c.

References agent_task_s::agent_return_data, interpreter_variable_data_s::array_dim, interpreter_variable_data_s::array_extent, CH_DATATYPE_SIZE, interpreter_variable_data_s::data_type, agent_s::datastate, agent_datastate_s::number_of_tasks, size, and agent_datastate_s::tasks.

13.49.14.1 EXPORTMC int MC_GetAgentStatus (MCAgent_t *agent*)

Get an agent's current status.

Returns:

returns type 'enum MC_AgentStatus_e'

Definition at line 1049 of file libmc.c.

References agent_s::agent_status, agent_s::lock, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_GetAgentStatus_chdl().

13.49.14.2 EXPORTCH int MC_GetAgentStatus_chdl (void * *varg*)

Definition at line 2908 of file libmc.c.

References MC_GetAgentStatus().

Referenced by agent_ChScriptInitVar().

13.49.14.3 EXPORTMC enum MC_AgentType_e MC_GetAgentType (MC_Agent_t *agent*)

Get an agent's type.

Returns:

returns type 'enum MC_AgentType_e'

Definition at line 1172 of file libmc.c.

References agent_s::agent_type.

13.49.14.4 EXPORTMC char* MC_GetAgentXMLString (MC_Agent_t *agent*)

Get an agent's xml string.

Returns:

a malloc'd character string containing the agent's xml code

Definition at line 1059 of file libmc.c.

References agent_s::datastate, mxmlSaveAllocString(), and agent_datastate_s::xml_agent_root.

Referenced by MC_GetAgentXMLString_chdl().

13.49.14.5 EXPORTCH char* MC_GetAgentXMLString_chdl (void * *varg*)

Definition at line 2924 of file libmc.c.

References MC_GetAgentXMLString().

Referenced by agent_ChScriptInitVar().

13.49.14.6 int MC_GetAllAgents (MC_Agency_t *attr*, MC_Agent_t ** *agents*, int * *num_agents*)

Definition at line 1182 of file libmc.c.

References mc_platform_s::agent_queue, mc_platform_s::giant, mc_platform_s::giant_lock, MC_HaltAgency(), agency_s::mc_platform, MC_ResumeAgency(), MUTEX_LOCK, and MUTEX_UNLOCK.

13.49.14.7 EXPORTCH int MC_GetTimeOfDay_chdl (void * *varg*)

Definition at line 2940 of file libmc.c.

Referenced by agent_ChScriptInitVar().

13.49.14.8 EXPORTMC int MC_HaltAgency (MC_Agency_t *agency*)

Halt an agency: Do not process new entries in queues.

Parameters:

agency A handle to a running MobileC agency.

Returns:

0 on success, non-zero on failure.

Definition at line 1218 of file libmc.c.

References `mc_platform_s::giant`, `mc_platform_s::giant_lock`, `agency_s::mc_platform`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

Referenced by `MC_GetAllAgents()`, and `MC_HaltAgency_chdl()`.

13.49.14.9 EXPORTCH int MC_HaltAgency_chdl (void * *varg*)

Definition at line 2955 of file libmc.c.

References `CHECK_NULL`, `MC_HaltAgency()`, and `agency_s::mc_platform`.

Referenced by `agent_ChScriptInitVar()`.

13.49.14.10 EXPORTMC MCAgency_t MC_Initialize (int *port*, MCAgencyOptions_t * *options*)

Initialize and start a MobileC agency.

Parameters:

port the TCP port the agency should bind to

options initialized MobileC [options](#) or NULL for default [options](#)

Returns:

a handle to a running MobileC agency or NULL on failure

13.49.15 Example

Definition at line 1227 of file libmc.c.

References `mc_platform_s::agency`, `buf`, `MCAgencyOptions_s::ch_options`, `agency_s::client`, `MCAgencyOptions_s::default_agent_status`, `agency_s::default_agentstatus`, `f`, `HOST_NAME_MAX`, `agency_s::hostName`, `MCAgencyOptions_s::initInterps`, `agency_s::initInterps`, `agency_s::known_host_filename`, `MCAgencyOptions_s::known_host_filename`, `MC_InitializeAgencyOptions()`, `mc_platform`, `agency_s::mc_platform`, `mc_platform_Initialize()`, `MC_THREAD_ALL`, `MCAgencyOptions_s::passphrase`, `agency_s::portno`, `agency_s::priv_key_filename`, `MCAgencyOptions_s::priv_key_filename`, `read_encrypted_file()`, `agency_s::server`, `MCAgencyOptions_s::stack_size`, `agency_s::stack_size`, `MCAgencyOptions_s::threads`, and `agency_s::threads`.

13.49.15.1 EXPORTMC int MC_InitializeAgencyOptions (struct MCAgencyOptions_s * *options*)

Initialize MobileC [options](#).

Parameters:

options [options](#) to initialize.

Returns:

0 on success, `error_code_t` on failure

Note:

MobileC [options](#) should be initialized with this function before any of its members are modified.

13.49.16 Example

Definition at line 1335 of file `libmc.c`.

References `MCAgencyOptions_s::default_agent_status`, `MCAgencyOptions_s::initInterps`, `MCAgencyOptions_s::known_host_filename`, `MC_THREAD_ALL`, `MC_WAIT_CH`, `MCAgencyOptions_s::modified`, `MCAgencyOptions_s::passphrase`, `MCAgencyOptions_s::priv_key_filename`, `MCAgencyOptions_s::stack_size`, and `MCAgencyOptions_s::threads`.

Referenced by `MC_Initialize()`.

13.49.16.1 EXPORTMC int MC_LoadAgentFromFile (MCAgency_t attr, const char *filename)

Load an agent from a file into an agency.

Parameters:

agency A valid and running Mobile-C agency

filename Filename containing the agent to load

Returns:

0 on success, non-zero on failure.

Definition at line 1356 of file `libmc.c`.

References `buf`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, `mc_platform_s::message_queue`, `MXML_DESCEND`, `mxmlFindElement()`, `mxmlLoadString()`, `message_s::to_address`, `message_s::xml_payload`, and `message_s::xml_root`.

13.49.16.2 int MC_MainLoop (MCAgency_t attr)

Wait indefinitely.

Note:

This function is intended to block the calling thread forever.

Definition at line 2010 of file `libmc.c`.

References `COND_WAIT`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, `mc_platform_s::quit`, `mc_platform_s::quit_cond`, and `mc_platform_s::quit_lock`.

13.49.16.3 EXPORTMC int MC_MigrateAgent (MCAgent_t *agent*, const char * *hostname*, int *port*)

Migrates a running agent to another host.

Parameters:

agent The agent to migrate
hostname The new host to migrate the agent to
port The new port to migrate the agent to

Returns:

0 on success, error_code_t type on failure.

Definition at line 1426 of file libmc.c.

References agent_s::datastate, agent_datastate_s::progress_modifier, agent_task_s::server_name, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by MC_MigrateAgent_chdl().

13.49.16.4 EXPORTCH int MC_MigrateAgent_chdl (void * *varg*)

Definition at line 2972 of file libmc.c.

References MC_MigrateAgent(), and port.

Referenced by agent_ChScriptInitVar().

13.49.16.5 EXPORTMC int MC_MutexLock (MCAgency_t *attr*, int *id*)

Locks a MobileC synchronization variable as a mutex.

Parameters:

attr a MobileC agency handle
id the synchronization variable id to lock

Returns:

0 on success, error_code_t type on failure

13.49.17 Example

Consider the following agents, which use the agent-space version of this api function. Note that the 'sleep' agent is sent first, followed by the 'wake' agent.

Definition at line 1448 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_LOCK, mc_platform_s::syncList, and syncListFind().

Referenced by MC_MutexLock_chdl().

13.49.17.1 EXPORTCH int MC_MutexLock_chdl (void * *varg*)

Definition at line 2991 of file libmc.c.

References CHECK_NULL, MC_MutexLock(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.17.2 EXPORTMC int MC_MutexUnlock (MCAgency_t *attr*, int *id*)

Definition at line 1460 of file libmc.c.

References syncListNode_s::lock, agency_s::mc_platform, MUTEX_UNLOCK, mc_platform_s::syncList, and syncListFind().

Referenced by MC_MutexUnlock_chdl().

13.49.17.3 EXPORTCH int MC_MutexUnlock_chdl (void * *varg*)

Definition at line 3015 of file libmc.c.

References CHECK_NULL, MC_MutexUnlock(), and agency_s::mc_platform.

Referenced by agent_ChScriptInitVar().

13.49.17.4 EXPORTMC int MC_PrintAgentCode (MCAgent_t *agent*)

Prints an agents code to stdout.

Returns:

0 on success, error_code_t on failure

Definition at line 1472 of file libmc.c.

References agent_datastate_s::agent_code, agent_s::datastate, agent_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, agent_datastate_s::number_of_tasks, and agent_datastate_s::task_progress.

Referenced by MC_PrintAgentCode_chdl().

13.49.17.5 EXPORTCH int MC_PrintAgentCode_chdl (void * *varg*)

Definition at line 3039 of file libmc.c.

References MC_PrintAgentCode().

Referenced by agent_ChScriptInitVar().

13.49.17.6 EXPORTMC int MC_RegisterService (MCAgency_t *agency*, MCAgent_t *agent*, int *agentID*, const char * *agentName*, char ** *serviceNames*, int *numServices*)

Register a new service with the Directory Facilitator.

Parameters:

agency a MobileC agency handle

agent (OPTIONAL: See note) a MobileC agent
agentID (OPTIONAL: See note) a MobileC agent id
agentName (OPTIONAL: See note) a MobileC agent name
serviceNames an array of character strings of service names
numServices the number of services described in 'serviceNames'

Returns:

0 on success, error_code_t type on failure

Note:

Three of the input arguments are optional. The function expects as input the arguments 'agent XOR (agentID AND agentName)'.

13.49.18 Example

Definition at line 1488 of file libmc.c.

References CHECK_NULL, mc_platform_s::df, df_AddRequest(), df_request_list_node_New(), agent_s::id, MC_ERR_INVALID_ARGS, MC_ERR_MEMORY, agency_s::mc_platform, MUTEX_INIT, MUTEX_T, and agent_s::name.

Referenced by MC_RegisterService_chdl().

13.49.18.1 EXPORTCH int MC_RegisterService_chdl (void * varg)

Definition at line 3055 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_RegisterService().

Referenced by agent_ChScriptInitVar().

13.49.18.2 EXPORTMC int MC_ResetSignal (MCAgency_t attr)

Reset a MobileC signal.

Returns:

0 on success, error_code_t on failure

See also:

[MC_WaitSignal\(\)](#)

Definition at line 1620 of file libmc.c.

References COND_SIGNAL, mc_platform_s::giant, mc_platform_s::giant_cond, mc_platform_s::giant_lock, MC_NO_SIGNAL, agency_s::mc_platform, mc_platform_s::MC_signal, MUTEX_LOCK, and MUTEX_UNLOCK.

13.49.18.3 EXPORTMC int MC_ResumeAgency (MCAgency_t *agency*)

Resumes a halted agency.

Parameters:

agency An agency previously halted with the [MC_HaltAgency\(\)](#) function.

Returns:

0 on success, non-zero on failure.

Definition at line 1567 of file libmc.c.

References mc_platform_s::giant, mc_platform_s::giant_lock, agency_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_GetAllAgents(), and MC_ResumeAgency_chdl().

13.49.18.4 EXPORTCH int MC_ResumeAgency_chdl (void * *varg*)

Definition at line 3092 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_ResumeAgency().

Referenced by agent_ChScriptInitVar().

13.49.18.5 EXPORTMC MCAgent_t MC_RetrieveAgent (MCAgency_t *attr*)

Retrieves the oldest agent from an agency.

Returns:

a valid agent or NULL on failure

Definition at line 1576 of file libmc.c.

References mc_platform_s::agent_queue, agent_s::agent_status, ListSearch(), MC_AGENT_NEUTRAL, agency_s::mc_platform, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_RetrieveAgent_chdl().

13.49.18.6 EXPORTCH MCAgent_t MC_RetrieveAgent_chdl (void * *varg*)

Definition at line 3110 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_RetrieveAgent().

Referenced by agent_ChScriptInitVar().

13.49.18.7 EXPORTMC char* MC_RetrieveAgentCode (MCAgent_t *agent*)

Retrieves an agent's Ch code.

Returns:

a malloc'd character string on success, NULL on failure

Definition at line 1604 of file libmc.c.

References `agent_datastate_s::agent_code`, `buf`, `agent_s::datastate`, `agent_s::lock`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_datastate_s::task_progress`.

Referenced by `MC_RetrieveAgentCode_chdl()`.

13.49.18.8 EXPORTCH char* MC_RetrieveAgentCode_chdl (void * *varg*)

Definition at line 3127 of file libmc.c.

References `MC_RetrieveAgentCode()`.

Referenced by `agent_ChScriptInitVar()`.

13.49.18.9 EXPORTCH int MC_SaveData_chdl (void * *varg*)

Definition at line 3143 of file libmc.c.

References `agent_task_s::agent_variable_list`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `agent_s::datastate`, `interpreter_variable_data_New()`, `interpreter_variable_data_s::name`, `interpreter_variable_data_s::size`, `size`, `agent_datastate_s::task_progress`, and `agent_datastate_s::tasks`.

Referenced by `agent_ChScriptInitVar()`.

13.49.18.10 EXPORTMC int MC_SearchForService (MCAgency_t *attr*, const char * *searchString*, char *** *agentNames*, char *** *serviceNames*, int ** *agentIDs*, int * *numResults*)

Search the directory facilitator for a service.

Returns:

0 on success, `error_code_t` on failure

Parameters:

attr (input) a MobileC agency handle

searchString (input) substring to search services for

agentNames (return) array of agent names with matching services

serviceNames (return) array of matching service names

agentIDs (return) array of matching agent IDs

numResults (return) number of matching results

13.49.19 Example

Definition at line 1631 of file libmc.c.

References `CHECK_NULL`, `COND_SLEEP_ACTION`, `mc_platform_s::df`, `df_AddRequest()`, `df_request_list_node_Destroy()`, `df_request_list_node_New()`, `df_request_search_Destroy()`, `df_request_search_New()`, `MC_ERR_MEMORY`, `agency_s::mc_platform`, `MC_SUCCESS`, and `search`.

Referenced by `MC_SearchForService_chdl()`.

13.49.19.1 EXPORTCH int MC_SearchForService_chdl (void * *varg*)

Definition at line 3178 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SearchForService().

Referenced by agent_ChScriptInitVar().

13.49.19.2 EXPORTMC int MC_SemaphorePost (MCAgency_t *attr*, int *id*)

Post to a MobileC synchronization variable semaphore.

Parameters:

attr a MobileC agency handle

id the synchronization variable id to post to

Returns:

0 on success, error_code_t type on failure

13.49.20 Example

Definition at line 1681 of file libmc.c.

References agency_s::mc_platform, syncListNode_s::sem, SEMAPHORE_POST, mc_platform_s::syncList, and syncListFind().

Referenced by MC_SemaphorePost_chdl().

13.49.20.1 EXPORTCH int MC_SemaphorePost_chdl (void * *varg*)

Definition at line 3219 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SemaphorePost().

Referenced by agent_ChScriptInitVar().

13.49.20.2 EXPORTMC int MC_SemaphoreWait (MCAgency_t *attr*, int *id*)

Decreases a MobileC synchronization variable semaphore count by one.

Parameters:

attr a MobileC agency handle

id synchronization variable id to wait on

Returns:

0 on MC_SUCCESS, error_code_t type of failure

Note:

If the semaphore count is already zero, this function will block until another thread posts to the semaphore.

13.49.21 Example

Definition at line 1693 of file libmc.c.

References `agency_s::mc_platform`, `syncListNode_s::sem`, `SEMAPHORE_WAIT`, `mc_platform_s::syncList`, and `syncListFind()`.

Referenced by `MC_SemaphoreWait_chdl()`.

13.49.21.1 EXPORTCH int MC_SemaphoreWait_chdl (void * *varg*)

Definition at line 3243 of file libmc.c.

References `CHECK_NULL`, `agency_s::mc_platform`, and `MC_SemaphoreWait()`.

Referenced by `agent_ChScriptInitVar()`.

13.49.21.2 EXPORTMC int MC_SendAgentMigrationMessage (MCAgency_t *attr*, const char * *message*, const char * *hostname*, int *port*)

Sends an agent migration message.

Parameters:

- attr* a MobileC agency handle
- message* a valid MobileC xml agent migration message
- hostname* host to send the message to
- port* port to send the message to

Definition at line 1715 of file libmc.c.

References `MC_ERR`, `agency_s::mc_platform`, `message_Destroy()`, `message_InitializeFromString()`, `message_New()`, and `mc_platform_s::message_queue`.

Referenced by `MC_SendAgentMigrationMessage_chdl()`.

13.49.21.3 EXPORTCH int MC_SendAgentMigrationMessage_chdl (void * *varg*)

Definition at line 3267 of file libmc.c.

References `CHECK_NULL`, `agency_s::mc_platform`, `MC_SendAgentMigrationMessage()`, and `port`.

Referenced by `agent_ChScriptInitVar()`.

13.49.21.4 EXPORTMC int MC_SendAgentMigrationMessageFile (MCAgency_t *attr*, const char * *filename*, const char * *hostname*, int *port*)

Sends an agent migration message.

Parameters:

- attr* a MobileC agency handle
- filename* file containing a valid MobileC xml agent migration message

hostname hostname to send the agent to

port port to send the agent to

Definition at line 1746 of file libmc.c.

References agent_Initialize(), mc_platform_s::agent_queue, agent_s::agent_status, mc_platform_s::ams, buf, COND_BROADCAST, agency_s::mc_platform, MC_WAIT_MESSGSEND, message_Destroy(), message_InitializeFromString(), message_New(), MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by handler_SEND(), and MC_SendAgentMigrationMessageFile_chdl().

13.49.21.5 EXPORTCH int MC_SendAgentMigrationMessageFile_chdl (void * *varg*)

Definition at line 3292 of file libmc.c.

References MC_SendAgentMigrationMessageFile(), and port.

Referenced by agent_ChScriptInitVar().

13.49.21.6 int MC_SendCh (MCAgency_t *attr*, const char * *filename*, const char * *remotehost*, int *port*)

Definition at line 1705 of file libmc.c.

13.49.21.7 EXPORTMC int MC_SendSteerCommand (MCAgency_t *attr*, enum MC_SteerCommand_e *cmd*)

Definition at line 1814 of file libmc.c.

References COND_BROADCAST, agency_s::mc_platform, mc_platform_s::MC_steer_command, mc_platform_s::MC_steer_cond, mc_platform_s::MC_steer_lock, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_SendSteerCommand_chdl().

13.49.21.8 EXPORTCH int MC_SendSteerCommand_chdl (void * *varg*)

Definition at line 3312 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SendSteerCommand().

Referenced by agent_ChScriptInitVar().

13.49.21.9 int MC_SetAgentStatus (MCAgent_t *agent*, int *status*)

Set an agent's status.

Parameters:

agent a MobileC agent

status agent status of type 'enum MC_AgentStatus_e'

Returns:

0 on success, or error_code_t on failure

Definition at line 1824 of file libmc.c.

References `agent_s::agent_status`, `mc_platform_s::ams`, `COND_SIGNAL`, `agent_s::lock`, `agent_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `agent_s::orphan`.

Referenced by `MC_DeleteAgent()`, `MC_DeleteAgentWG()`, and `MC_SetAgentStatus_chdl()`.

13.49.21.10 EXPORTCH int MC_SetAgentStatus_chdl (void * *varg*)

Definition at line 3335 of file libmc.c.

References `MC_SetAgentStatus()`.

Referenced by `agent_ChScriptInitVar()`.

13.49.21.11 int MC_SetDefaultAgentStatus (MCAgency_t *agency*, enum MC_AgentStatus_e *status*)

Sets default incoming agent status.

Parameters:

agency a MobileC agency handle

status the status to set all incoming agents

Returns:

0 on success, `error_type_t` on failure

Note:

using this function will override any status the incoming agent attempts to set for itself.

Definition at line 1839 of file libmc.c.

References `mc_platform_s::default_agentstatus`, and `agency_s::mc_platform`.

Referenced by `MC_SetDefaultAgentStatus_chdl()`.

13.49.21.12 EXPORTCH int MC_SetDefaultAgentStatus_chdl (void * *varg*)

Definition at line 3353 of file libmc.c.

References `CHECK_NULL`, `agency_s::mc_platform`, and `MC_SetDefaultAgentStatus()`.

Referenced by `agent_ChScriptInitVar()`.

13.49.21.13 EXPORTMC int MC_SetThreadOff (MCAgencyOptions_t * *options*, enum MC_ThreadIndex_e *index*)

Sets a MobileC thread to "off" status.

Parameters:

options MobileC [options](#) previously initialized with [MC_InitializeAgencyOptions\(\)](#)

index the thread to set

Returns:

0 on success, error_code_t on failure

Note:

This function must be called before [MC_Initialize\(\)](#). Once an agency is started with MC_Initialize, the MC_SetThread functions will have no effect.

Definition at line 1866 of file libmc.c.

References SET_THREAD_OFF, and MCAgencyOptions_s::threads.

13.49.21.14 EXPORTMC int MC_SetThreadOn (MCAgencyOptions_t * options, enum MC_ThreadIndex_e index)

Sets a MobileC thread to "on" status.

Parameters:

[options](#) MobileC [options](#) previously initialized with [MC_InitializeAgencyOptions\(\)](#)

[index](#) the thread to set

Returns:

0 on success, error_code_t on failure

Note:

This function must be called before [MC_Initialize\(\)](#). Once an agency is started with MC_Initialize, the MC_SetThread functions will have no effect.

Definition at line 1849 of file libmc.c.

References SET_THREAD_ON, and MCAgencyOptions_s::threads.

13.49.21.15 EXPORTMC int MC_SetThreadsAllOff (MCAgencyOptions_t * options)

Set all MobileC threads to 'off' status.

Parameters:

[options](#) a MobileC [options](#) structure initialized with with the [MC_InitializeAgencyOptions\(\)](#) function.

Returns:

0 on success, error code on failure.

Definition at line 1873 of file libmc.c.

References MC_THREAD_ALL, SET_THREAD_OFF, and MCAgencyOptions_s::threads.

13.49.21.16 EXPORTMC int MC_SetThreadsAllOn (MCAgencyOptions_t * options)

Set all Mobile-C threads on.

Parameters:

options MobileC *options* structure, initialized with `MC_InitializeAgencyOptions()`

Returns:

0 on success, error code on failure.

Definition at line 1856 of file libmc.c.

References `MC_THREAD_ALL`, `SET_THREAD_ON`, and `MCAgencyOptions_s::threads`.

13.49.21.17 EXPORTMC int MC_Steer (MCAgency_t *attr*, int(*) (void **data*) *funcptr*, void * *arg*)

Set up a steerable algorithm.

Parameters:

attr a MobileC agency handle

funcptr a function pointer to the algorithm

arg an argument for the algorithm function

Returns:

0 on success, `error_code_t` on failure

Note:

The algorithm function must contain a call to `MC_SteerControl` in order for the algorithm to be steerable.

13.49.22 Example

Definition at line 1883 of file libmc.c.

References `agency_s::mc_platform`, `MC_RESTART`, `MC_RUN`, `mc_platform_s::MC_steer_command`, `mc_platform_s::MC_steer_lock`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

13.49.22.1 EXPORTMC enum MC_SteerCommand_e MC_SteerControl (void)

The MobileC user-algorithm steering function.

Returns:

The current steering command

Note:

This function belongs inside a user's steerable algorithm.

See also:

[MC_Steer\(\)](#)

Definition at line 1902 of file libmc.c.

References COND_WAIT, mc_platform_s::MC_steer_command, mc_platform_s::MC_steer_cond, mc_platform_s::MC_steer_lock, MC_SUSPEND, MUTEX_LOCK, and MUTEX_UNLOCK.

13.49.22.2 EXPORTMC int MC_SyncDelete (MCAgency_t *attr*, int *id*)

Deletes a previously initialized synchronization variable.

Parameters:

attr a MobileC agency handle

id the sync variable id to delete

Returns:

0 on success, or error_code_t on failure

Definition at line 1918 of file libmc.c.

References syncList_s::giant_lock, syncListNode_s::lock, MC_ERR_NOT_FOUND, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, mc_platform_s::syncList, syncListFind(), syncListNodeDestroy(), and syncListRemove().

Referenced by MC_SyncDelete_chdl().

13.49.22.3 EXPORTCH int MC_SyncDelete_chdl (void * *varg*)

Definition at line 3376 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SyncDelete().

Referenced by agent_ChScriptInitVar().

13.49.22.4 EXPORTMC int MC_SyncInit (MCAgency_t *attr*, int *id*)

Initializes a new MobileC synchronization variable.

Parameters:

attr a MobileC agency handle

id the requested sync variable id

Returns:

new sync variable's id. May be different than the requested id.

Note:

Each synchronization variable may be used as a mutex, condition variable, or semaphore. However, it should only be used as one type of synchronization variable per instance, or undefined behaviour may result.

See also:

[MC_MutexLock\(\)](#), [MC_MutexUnlock\(\)](#), [MC_CondWait\(\)](#), [MC_CondSignal\(\)](#), [MC_CondBroadcast](#), [MC_SemaphorePost\(\)](#), [MC_SemaphoreWait\(\)](#)

Definition at line 1948 of file libmc.c.

References syncList_s::giant_lock, syncListNode_s::id, agency_s::mc_platform, MUTEX_LOCK, MUTEX_UNLOCK, node, mc_platform_s::syncList, syncListAddNode(), syncListFind(), and syncListNodeNew().

Referenced by MC_SyncInit_chdl().

13.49.22.5 EXPORTCH int MC_SyncInit_chdl (void * *varg*)

Definition at line 3399 of file libmc.c.

References CHECK_NULL, agency_s::mc_platform, and MC_SyncInit().

Referenced by agent_ChScriptInitVar().

13.49.22.6 EXPORTMC int MC_TerminateAgent (MCAgent_t *agent*)

Halt a running agent.

Returns:

0 on success, error_code_t on failure

Definition at line 1973 of file libmc.c.

References agent_s::agent_interp.

Referenced by ams_ManageAgentList(), MC_DeleteAgent(), and MC_TerminateAgent_chdl().

13.49.22.7 EXPORTCH int MC_TerminateAgent_chdl (void * *varg*)

Definition at line 3423 of file libmc.c.

References MC_ERR_NOT_FOUND, MC_FindAgentByName(), and MC_TerminateAgent().

Referenced by agent_ChScriptInitVar().

13.49.22.8 EXPORTMC int MC_TerminateAgentWG (MCAgent_t *calling_agent*, MCAgent_t *agent*)

Definition at line 1983 of file libmc.c.

References agent_s::agent_interp, MC_ERR_INVALID_ARGS, and agent_s::wg_code.

Referenced by MC_DeleteAgentWG(), and MC_TerminateAgentWG_chdl().

13.49.22.9 EXPORTCH int MC_TerminateAgentWG_chdl (void * *varg*)

Definition at line 3444 of file libmc.c.

References MC_ERR_NOT_FOUND, MC_FindAgentByName(), and MC_TerminateAgentWG().

Referenced by agent_ChScriptInitVar().

13.49.22.10 EXPORTMC int MC_WaitAgent (MCAgency_t *attr*)

Wait indefinitely.

Note:

This function is intended to block the calling thread forever. Wait for an agent arrival event

This function blocks until an agent arrival signal is triggered, at which point it unblocks.

Definition at line 2022 of file libmc.c.

References `mc_platform_s::agent_queue`, `COND_WAIT`, `mc_platform`, `agency_s::mc_platform`, `MUTEX_LOCK`, `MUTEX_UNLOCK`, and `size`.

13.49.22.11 EXPORTMC MCAgent_t MC_WaitRetrieveAgent (MCAgency_t *attr*)

Wait and retrieve an agent.

Returns:

a valid MobileC agent on success, or NULL on failure

Note:

This function blocks until the arrival of an agent. The agent is retrieved after it is initialized, but before it is executed.

Definition at line 2042 of file libmc.c.

References `mc_platform_s::agent_queue`, `ListSearch()`, `agency_s::mc_platform`, `MC_RECV_AGENT`, `MC_WaitSignal()`, `MUTEX_LOCK`, and `MUTEX_UNLOCK`.

13.49.22.12 EXPORTMC int MC_WaitSignal (MCAgency_t *attr*, int *signals*)

Wait for a MobileC signal.

Parameters:

attr a MobileC agency handle

signals a flag of signals to wait for, of type 'enum MC_Signal_e'

Returns:

0 on success, `error_code_t` on failure

Note:

the parameter 'signals' may be something like 'MC_RECV_MESSAGE | MC_RECV_AGENT', etc.

13.49.23 Example

Definition at line 2060 of file libmc.c.

References COND_WAIT, mc_platform_s::giant, mc_platform_s::giant_lock, agency_s::mc_platform, mc_platform_s::MC_signal, mc_platform_s::MC_signal_cond, mc_platform_s::MC_signal_lock, MUTEX_LOCK, and MUTEX_UNLOCK.

Referenced by MC_WaitRetrieveAgent().

13.49.24 Variable Documentation

13.49.24.1 mc_platform_p g_mc_platform

Definition at line 71 of file libmc.c.

Referenced by fipa_envelope_Compose__from().

13.50 /home/dko/Projects/mobilec/trunk/src/mc_list/list.c File Reference

```
#include "list.h"
#include <stdio.h>
#include <stdlib.h>
```

Functions

- [list_p ListInitialize](#) (void)
- void [ListTerminate](#) ([list_p](#) list)
- int [list_pGetSize](#) ([list_p](#) list)
- DATA [ListGetHead](#) ([list_p](#) list)
- DATA [ListPop](#) ([list_p](#) list)
- DATA [ListSearch](#) ([list_p](#) list, const int index)
- int [ListAdd](#) ([list_p](#) list, DATA data)
- int [ListInsert](#) ([list_p](#) list, DATA data, const int index)
- DATA [ListDelete](#) ([list_p](#) list, const int index)

13.50.1 Function Documentation

13.50.1.1 int list_pGetSize (list_p list)

Definition at line 73 of file list.c.

References [list_s::size](#).

13.50.1.2 int ListAdd (list_p list, DATA data)

Definition at line 165 of file list.c.

References [list_s::listhead](#), [listNode_s::next](#), [listNode_s::node_data](#), and [list_s::size](#).

Referenced by [agent_task_Copy\(\)](#), [barrier_queue_Add\(\)](#), [df_Add\(\)](#), [df_AddRequest\(\)](#), and [syncListAddNode\(\)](#).

13.50.1.3 DATA ListDelete (list_p list, const int index)

Definition at line 220 of file list.c.

References [DATA](#), [list_s::listhead](#), [listNode_s::next](#), [listNode_s::node_data](#), and [list_s::size](#).

Referenced by [barrier_queue_Delete\(\)](#), [syncListDelete\(\)](#), and [syncListRemove\(\)](#).

13.50.1.4 DATA ListGetHead (list_p list)

Definition at line 79 of file list.c.

References [list_s::listhead](#), and [listNode_s::node_data](#).

13.50.1.5 list_p ListInitialize (void)

Definition at line 33 of file list.c.

References list_s::listhead, and list_s::size.

Referenced by barrier_queue_New(), df_Initialize(), df_request_list_New(), and syncListInit().

13.50.1.6 int ListInsert (list_p list, DATA data, const int index)

Definition at line 207 of file list.c.

13.50.1.7 DATA ListPop (list_p list)

Definition at line 89 of file list.c.

References DATA, list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_queue_Pop(), df_Destroy(), df_request_list_Destroy(), and df_request_list_Pop().

13.50.1.8 DATA ListSearch (list_p list, const int index)

Definition at line 124 of file list.c.

References list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by agent_queue_Flush(), agent_task_Copy(), ams_ManageAgentList(), ams_Print(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_queue_Delete(), MC_RetrieveAgent(), MC_WaitRetrieveAgent(), message_queue_SendOutgoing(), syncListDelete(), and syncListRemove().

13.50.1.9 void ListTerminate (list_p list)

Definition at line 49 of file list.c.

References list_s::listhead, and list_s::size.

Referenced by barrier_queue_Destroy(), df_Destroy(), and df_request_list_Destroy().

13.51 /home/dko/Projects/mobilec/trunk/src/mc_list/list.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
```

Data Structures

- struct [listNode_s](#)
- struct [list_s](#)

Defines

- #define [DATA](#) void*
- #define [QUEUE_TEMPLATE](#)(name, node_type, search_type, search_var_name)

Typedefs

- typedef struct [listNode_s](#) [listNode_t](#)
- typedef [listNode_t](#) * [listNode_p](#)
- typedef struct [list_s](#) [list_t](#)
- typedef [list_t](#) * [list_p](#)

Functions

- [list_p](#) [ListInitialize](#) (void)
- void [ListTerminate](#) ([list_p](#) list)
- int [ListGetSize](#) ([list_p](#) list)
- int [ListAdd](#) ([list_p](#) list, DATA data)
- int [ListInsert](#) ([list_p](#) list, DATA data, const int index)
- DATA [ListGetHead](#) ([list_p](#) list)
- DATA [ListPop](#) ([list_p](#) list)
- DATA [ListSearch](#) ([list_p](#) list, const int index)
- DATA [ListDelete](#) ([list_p](#) list, const int index)

13.51.1 Define Documentation

13.51.1.1 #define DATA void*

Definition at line 30 of file list.h.

Referenced by [barrier_queue_Add\(\)](#), [ListDelete\(\)](#), [ListPop\(\)](#), and [syncListAddNode\(\)](#).

13.51.1.2 #define QUEUE_TEMPLATE(name, node_type, search_type, search_var_name)

Definition at line 68 of file list.h.

13.51.2 Typedef Documentation

13.51.2.1 typedef list_t* list_p

Definition at line 49 of file list.h.

13.51.2.2 typedef struct list_s list_t

13.51.2.3 typedef listNode_t* listNode_p

Definition at line 39 of file list.h.

13.51.2.4 typedef struct listNode_s listNode_t

13.51.3 Function Documentation

13.51.3.1 int ListAdd (list_p *list*, DATA *data*)

Definition at line 165 of file list.c.

References list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by agent_task_Copy(), barrier_queue_Add(), df_Add(), df_AddRequest(), and syncListAddNode().

13.51.3.2 DATA ListDelete (list_p *list*, const int *index*)

Definition at line 220 of file list.c.

References DATA, list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by barrier_queue_Delete(), syncListDelete(), and syncListRemove().

13.51.3.3 DATA ListGetHead (list_p *list*)

Definition at line 79 of file list.c.

References list_s::listhead, and listNode_s::node_data.

13.51.3.4 int ListGetSize (list_p *list*)

13.51.3.5 list_p ListInitialize (void)

Definition at line 33 of file list.c.

References list_s::listhead, and list_s::size.

Referenced by barrier_queue_New(), df_Initialize(), df_request_list_New(), and syncListInit().

13.51.3.6 int ListInsert (list_p *list*, DATA *data*, const int *index*)

Definition at line 207 of file list.c.

13.51.3.7 DATA ListPop (*list_p list*)

Definition at line 89 of file list.c.

References DATA, list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_queue_Pop(), df_Destroy(), df_request_list_Destroy(), and df_request_list_Pop().

13.51.3.8 DATA ListSearch (*list_p list*, *const int index*)

Definition at line 124 of file list.c.

References list_s::listhead, listNode_s::next, listNode_s::node_data, and list_s::size.

Referenced by agent_queue_Flush(), agent_task_Copy(), ams_ManageAgentList(), ams_Print(), AP_QUEUE_SEARCH_TEMPLATE(), AP_QUEUE_STD_DEFN_TEMPLATE(), barrier_queue_Delete(), MC_RetrieveAgent(), MC_WaitRetrieveAgent(), message_queue_SendOutgoing(), syncListDelete(), and syncListRemove().

13.51.3.9 void ListTerminate (*list_p list*)

Definition at line 49 of file list.c.

References list_s::listhead, and list_s::size.

Referenced by barrier_queue_Destroy(), df_Destroy(), and df_request_list_Destroy().

13.52 /home/dko/Projects/mobilec/trunk/src/mc_platform.c File Reference

```
#include "config.h"
#include <unistd.h>
#include <sys/socket.h>
#include <netdb.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include "include/acc.h"
#include "include/mc_platform.h"
#include "include/macros.h"
```

Defines

- `#define` [DEFAULT_HOSTNAME_LENGTH](#) 200

Functions

- [mc_platform_p mc_platform_Initialize](#) (MCAgency_t agency, ChOptions_t *ch_options)
- [int mc_platform_Destroy](#) (mc_platform_p platform)

13.52.1 Define Documentation

13.52.1.1 `#define` [DEFAULT_HOSTNAME_LENGTH](#) 200

Definition at line 51 of file mc_platform.c.

Referenced by [mc_platform_Initialize\(\)](#).

13.52.2 Function Documentation

13.52.2.1 `int` [mc_platform_Destroy](#) (mc_platform_p *platform*)

Definition at line 244 of file mc_platform.c.

References [mc_platform_s::acc](#), [acc_Destroy\(\)](#), [mc_platform_s::agent_queue](#), [mc_platform_s::ams](#), [ams_Destroy\(\)](#), [mc_platform_s::barrier_queue](#), [barrier_queue_Destroy\(\)](#), [mc_platform_s::cmd_prompt](#), [cmd_prompt_Destroy\(\)](#), [COND_DESTROY](#), [mc_platform_s::connection_queue](#), [mc_platform_s::df](#), [df_Destroy\(\)](#), [mc_platform_s::giant_cond](#), [mc_platform_s::giant_lock](#), [mc_platform_s::interp_options](#), [mc_platform_s::MC_signal_cond](#), [mc_platform_s::MC_signal_lock](#), [mc_platform_s::MC_steer_cond](#), [mc_platform_s::MC_steer_lock](#), [MC_SUCCESS](#), [mc_platform_s::MC_sync_cond](#), [mc_platform_s::MC_sync_lock](#), [mc_platform_s::message_queue](#), [MUTEX_DESTROY](#), [mc_platform_s::quit_lock](#), [SOCKET_ERROR](#), and [mc_platform_s::sockfd](#).

Referenced by [MC_End\(\)](#).

13.52.2.2 `mc_platform_p mc_platform_Initialize (MCAgency_t agency, ChOptions_t * ch_options)`

Definition at line 53 of file `mc_platform.c`.

References `acc_Initialize()`, `acc_Start()`, `agent_ChScriptInitVar()`, `ams_Initialize()`, `ams_Start()`, `barrier_queue_New()`, `CHECK_NULL`, `cmd_prompt_Initialize()`, `cmd_prompt_Start()`, `COND_INIT`, `COND_T`, `COND_WAIT`, `agency_s::default_agentstatus`, `DEFAULT_HOSTNAME_LENGTH`, `df_Initialize()`, `df_Start()`, `GET_THREAD_MODE`, `agency_s::initInterps`, `agency_s::last_error`, `MC_ERR_MEMORY`, `MC_NO_SIGNAL`, `mc_platform`, `MC_THREAD_ACC`, `MC_THREAD_ALL`, `MC_THREAD_AMS`, `MC_THREAD_CP`, `MC_THREAD_DF`, `MUTEX_INIT`, `MUTEX_LOCK`, `MUTEX_T`, `MUTEX_UNLOCK`, `agency_s::portno`, `agency_s::stack_size`, `syncListInit()`, and `agency_s::threads`.

Referenced by `MC_Initialize()`.

13.53 /home/dko/Projects/mobilec/trunk/src/mc_rwlock.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include "include/macros.h"
#include "include/mc_error.h"
#include "include/mc_rwlock.h"
```

Functions

- [int mc_rwlock_init \(mc_rwlock_p rwlock\)](#)
- [int mc_rwlock_destroy \(mc_rwlock_p rwlock\)](#)
- [int mc_rwlock_rdlock \(mc_rwlock_p rwlock\)](#)
- [int mc_rwlock_rdunlock \(mc_rwlock_p rwlock\)](#)
- [int mc_rwlock_wrlock \(mc_rwlock_p rwlock\)](#)
- [int mc_rwlock_wrunlock \(mc_rwlock_p rwlock\)](#)

13.53.1 Function Documentation

13.53.1.1 int mc_rwlock_destroy (mc_rwlock_p *rwlock*)

Definition at line 66 of file mc_rwlock.c.

References [mc_rwlock_s::cond](#), [COND_DESTROY](#), [mc_rwlock_s::lock](#), and [MUTEX_DESTROY](#).

13.53.1.2 int mc_rwlock_init (mc_rwlock_p *rwlock*)

Definition at line 46 of file mc_rwlock.c.

References [CHECK_NULL](#), [mc_rwlock_s::cond](#), [COND_INIT](#), [COND_T](#), [mc_rwlock_s::lock](#), [MC_ERR_MEMORY](#), [MUTEX_INIT](#), [MUTEX_T](#), [mc_rwlock_s::num_readers](#), [mc_rwlock_s::write_flag](#), and [mc_rwlock_s::write_request](#).

13.53.1.3 int mc_rwlock_rdlock (mc_rwlock_p *rwlock*)

Definition at line 79 of file mc_rwlock.c.

References [mc_rwlock_s::cond](#), [COND_WAIT](#), [mc_rwlock_s::lock](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), [mc_rwlock_s::num_readers](#), [mc_rwlock_s::write_flag](#), and [mc_rwlock_s::write_request](#).

13.53.1.4 int mc_rwlock_rdunlock (mc_rwlock_p *rwlock*)

Definition at line 94 of file mc_rwlock.c.

References [mc_rwlock_s::cond](#), [COND_SIGNAL](#), [mc_rwlock_s::lock](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), and [mc_rwlock_s::num_readers](#).

13.53.1.5 int mc_rwlock_wrlock (mc_rwlock_p *rwlock*)

Definition at line 107 of file mc_rwlock.c.

References mc_rwlock_s::cond, COND_WAIT, mc_rwlock_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, mc_rwlock_s::num_readers, mc_rwlock_s::write_flag, and mc_rwlock_s::write_request.

13.53.1.6 int mc_rwlock_wrunlock (mc_rwlock_p *rwlock*)

Definition at line 126 of file mc_rwlock.c.

References mc_rwlock_s::cond, COND_SIGNAL, mc_rwlock_s::lock, MUTEX_LOCK, MUTEX_UNLOCK, and mc_rwlock_s::write_flag.

13.54 /home/dko/Projects/mobilec/trunk/src/mc_sync/sync_list.c File Reference

```
#include <pthread.h>
#include "sync_list.h"
#include "../mc_list/list.h"
#include "../include/mc_error.h"
```

Functions

- [int syncListNodeInit](#) (struct [syncListNode_s](#) **node*)
- [struct syncListNode_s * syncListNodeNew](#) (void)
- [int syncListNodeDestroy](#) (struct [syncListNode_s](#) **node*)
- [syncListNode_t * syncListFind](#) (int *id*, struct [syncList_s](#) **list*)
- [struct syncList_s * syncListInit](#) (void)
- [int syncListAddNode](#) (struct [syncListNode_s](#) **node*, struct [syncList_s](#) **list*)
- [int syncListNew](#) (int *id*, struct [syncList_s](#) **list*)
- [int syncListDelete](#) (int *id*, struct [syncList_s](#) **list*)
- [syncListNode_t * syncListRemove](#) (int *id*, struct [syncList_s](#) **list*)

13.54.1 Function Documentation

13.54.1.1 int syncListAddNode (struct syncListNode_s * *node*, struct syncList_s * *list*)

Definition at line 86 of file `sync_list.c`.

References `DATA`, `syncListNode_s::id`, `syncList_s::list`, `ListAdd()`, `list_s::listhead`, `syncList_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_WRLock`, and `RWLOCK_WRunlock`.

Referenced by `MC_SyncInit()`, and `syncListNew()`.

13.54.1.2 int syncListDelete (int *id*, struct syncList_s * *list*)

Definition at line 115 of file `sync_list.c`.

References `syncListNode_s::id`, `syncList_s::list`, `ListDelete()`, `ListSearch()`, `syncList_s::lock`, `MC_ERR_NOT_FOUND`, `RWLOCK_WRLock`, `RWLOCK_WRunlock`, `list_s::size`, and `syncListNodeDestroy()`.

13.54.1.3 syncListNode_t* syncListFind (int *id*, struct syncList_s * *list*)

Definition at line 56 of file `sync_list.c`.

References `syncList_s::list`, `list_s::listhead`, `syncList_s::lock`, `listNode_s::next`, `listNode_s::node_data`, `RWLOCK_RDlock`, and `RWLOCK_RDunlock`.

Referenced by `MC_CondBroadcast()`, `MC_CondReset()`, `MC_CondSignal()`, `MC_CondWait()`, `MC_MutexLock()`, `MC_MutexUnlock()`, `MC_SemaphorePost()`, `MC_SemaphoreWait()`, `MC_SyncDelete()`, and `MC_SyncInit()`.

13.54.1.4 struct syncList_s* syncListInit (void) [read]

Definition at line 72 of file sync_list.c.

References ListInitialize(), syncList_s::lock, MUTEX_INIT, MUTEX_T, RWLOCK_INIT, and RWLOCK_T.

Referenced by mc_platform_Initialize().

13.54.1.5 int syncListNew (int id, struct syncList_s * list)

Definition at line 105 of file sync_list.c.

References node, syncListAddNode(), and syncListNodeInit().

13.54.1.6 int syncListNodeDestroy (struct syncListNode_s * node)

Definition at line 44 of file sync_list.c.

References syncListNode_s::cond, COND_DESTROY, syncListNode_s::lock, MUTEX_DESTROY, syncListNode_s::sem, and SEMAPHORE_DESTROY.

Referenced by MC_SyncDelete(), and syncListDelete().

13.54.1.7 int syncListNodeInit (struct syncListNode_s * node)

Definition at line 11 of file sync_list.c.

References CHECK_NULL, syncListNode_s::cond, COND_INIT, COND_T, syncListNode_s::lock, MUTEX_INIT, MUTEX_T, syncListNode_s::sem, SEMAPHORE_INIT, and SEMAPHORE_T.

Referenced by syncListNew().

13.54.1.8 struct syncListNode_s* syncListNodeNew (void) [read]

Definition at line 26 of file sync_list.c.

References CHECK_NULL, syncListNode_s::cond, COND_INIT, COND_T, syncListNode_s::lock, MUTEX_INIT, MUTEX_T, syncListNode_s::sem, SEMAPHORE_INIT, SEMAPHORE_T, and syncListNode_s::signalled.

Referenced by MC_SyncInit().

13.54.1.9 syncListNode_t* syncListRemove (int id, struct syncList_s * list)

Definition at line 132 of file sync_list.c.

References syncListNode_s::id, syncList_s::list, ListDelete(), ListSearch(), syncList_s::lock, RWLOCK_WRLock, RWLOCK_WRunLock, and list_s::size.

Referenced by MC_SyncDelete().

13.55 /home/dko/Projects/mobilec/trunk/src/mc_sync/sync_list.h File Reference

```
#include "../include/macros.h"
#include "../mc_list/list.h"
#include "../include/mc_rwlock.h"
```

Data Structures

- struct [syncListNode_s](#)
- struct [syncList_s](#)

Typedefs

- typedef struct [syncListNode_s](#) [syncListNode_t](#)
- typedef [syncListNode_t](#) * [syncListNode_p](#)
- typedef struct [syncList_s](#) [syncList_t](#)
- typedef [syncList_t](#) * [syncList_p](#)

Functions

- [int syncListNodeInit](#) (struct [syncListNode_s](#) *[node](#))
- [int syncListNodeDestroy](#) (struct [syncListNode_s](#) *[node](#))
- [syncListNode_t](#) * [syncListFind](#) (int [id](#), struct [syncList_s](#) *[list](#))
- struct [syncListNode_s](#) * [syncListNodeNew](#) (void)
- [int syncListDelete](#) (int [id](#), struct [syncList_s](#) *[list](#))
- [syncListNode_t](#) * [syncListRemove](#) (int [id](#), struct [syncList_s](#) *[list](#))
- struct [syncList_s](#) * [syncListInit](#) (void)
- [int syncListAddNode](#) (struct [syncListNode_s](#) *[node](#), struct [syncList_s](#) *[list](#))
- [int syncListNew](#) (int [id](#), struct [syncList_s](#) *[list](#))
- [syncListNode_t](#) * [syncListGet](#) (int [id](#), struct [syncList_s](#) *[list](#))

13.55.1 Typedef Documentation

13.55.1.1 typedef [syncList_t](#)* [syncList_p](#)

Definition at line 33 of file [sync_list.h](#).

13.55.1.2 typedef struct [syncList_s](#) [syncList_t](#)

13.55.1.3 typedef [syncListNode_t](#)* [syncListNode_p](#)

Definition at line 23 of file [sync_list.h](#).

13.55.1.4 typedef struct syncListNode_s syncListNode_t

13.55.2 Function Documentation

13.55.2.1 int syncListAddNode (struct syncListNode_s * *node*, struct syncList_s * *list*)

Definition at line 86 of file sync_list.c.

References DATA, syncListNode_s::id, syncList_s::list, ListAdd(), list_s::listhead, syncList_s::lock, listNode_s::next, listNode_s::node_data, RWLOCK_WRLock, and RWLOCK_WRunLock.

Referenced by MC_SyncInit(), and syncListNew().

13.55.2.2 int syncListDelete (int *id*, struct syncList_s * *list*)

Definition at line 115 of file sync_list.c.

References syncListNode_s::id, syncList_s::list, ListDelete(), ListSearch(), syncList_s::lock, MC_ERR_NOT_FOUND, RWLOCK_WRLock, RWLOCK_WRunLock, list_s::size, and syncListNodeDestroy().

13.55.2.3 syncListNode_t* syncListFind (int *id*, struct syncList_s * *list*)

Definition at line 56 of file sync_list.c.

References syncList_s::list, list_s::listhead, syncList_s::lock, listNode_s::next, listNode_s::node_data, RWLOCK_RDLOCK, and RWLOCK_RDUnLock.

Referenced by MC_CondBroadcast(), MC_CondReset(), MC_CondSignal(), MC_CondWait(), MC_MutexLock(), MC_MutexUnlock(), MC_SemaphorePost(), MC_SemaphoreWait(), MC_SyncDelete(), and MC_SyncInit().

13.55.2.4 syncListNode_t* syncListGet (int *id*, struct syncList_s * *list*)

13.55.2.5 struct syncList_s* syncListInit (void) [read]

Definition at line 72 of file sync_list.c.

References ListInitialize(), syncList_s::lock, MUTEX_INIT, MUTEX_T, RWLOCK_INIT, and RWLOCK_T.

Referenced by mc_platform_Initialize().

13.55.2.6 int syncListNew (int *id*, struct syncList_s * *list*)

Definition at line 105 of file sync_list.c.

References node, syncListAddNode(), and syncListNodeInit().

13.55.2.7 int syncListNodeDestroy (struct syncListNode_s * *node*)

Definition at line 44 of file sync_list.c.

References syncListNode_s::cond, COND_DESTROY, syncListNode_s::lock, MUTEX_DESTROY, syncListNode_s::sem, and SEMAPHORE_DESTROY.

Referenced by MC_SyncDelete(), and syncListDelete().

13.55.2.8 int syncListNodeInit (struct syncListNode_s * node)

Definition at line 11 of file sync_list.c.

References CHECK_NULL, syncListNode_s::cond, COND_INIT, COND_T, syncListNode_s::lock, MUTEX_INIT, MUTEX_T, syncListNode_s::sem, SEMAPHORE_INIT, and SEMAPHORE_T.

Referenced by syncListNew().

13.55.2.9 struct syncListNode_s* syncListNodeNew (void) [read]

Definition at line 26 of file sync_list.c.

References CHECK_NULL, syncListNode_s::cond, COND_INIT, COND_T, syncListNode_s::lock, MUTEX_INIT, MUTEX_T, syncListNode_s::sem, SEMAPHORE_INIT, SEMAPHORE_T, and syncListNode_s::signalled.

Referenced by MC_SyncInit().

13.55.2.10 syncListNode_t* syncListRemove (int id, struct syncList_s * list)

Definition at line 132 of file sync_list.c.

References syncListNode_s::id, syncList_s::list, ListDelete(), ListSearch(), syncList_s::lock, RWLOCK_WRLOCK, RWLOCK_WRUNLOCK, and list_s::size.

Referenced by MC_SyncDelete().

13.56 /home/dko/Projects/mobilec/trunk/src/message.c File Reference

```
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <netdb.h>
#include <errno.h>
#include "config.h"
#include <mxml.h>
#include "include/libmc.h"
#include "include/agent.h"
#include "include/mc_platform.h"
#include "include/message.h"
#include "include/mtp_http.h"
#include "include/xml_compose.h"
#include "include/xml_helper.h"
#include "include/xml_parser.h"
#include "include/macros.h"
#include "security/asm_node.h"
```

Defines

- `#define SOCKET_INPUT_SIZE 4096`
- `#define MSG_THREADS 40`
- `#define MSG_THREAD_EXIT()`

Functions

- `message_p message_New (void)`
- `message_p message_Copy (message_p src)`
- `int message_InitializeFromAgent (mc_platform_p mc_platform, message_p message, agent_p agent)`
- `int message_InitializeFromConnection (mc_platform_p mc_platform, message_p message, connection_p connection)`
- `int http_to_hostport (const char *http_str, char **host, int *port, char **target)`
- `int message_InitializeFromString (mc_platform_p mc_platform, message_p message, const char *string, const char *destination_host, int destination_port, const char *target)`
- `int message_Destroy (message_p message)`

- `int auth_rece_send_msg` (`int sockfd`, `char *hostname`, `char *message`, `char *privkey`, `char *known_host_filename`)
- `int message_Send` (`mc_platform_t *mc_platform`, `message_p message`, `char *privatekey`)
- `void * message_send_Thread` (`void *arg`)

13.56.1 Define Documentation

13.56.1.1 #define MSG_THREAD_EXIT()

Value:

```
free(arg); \
message_Destroy(message); \
MUTEX_LOCK(&mc_platform->acc->msg_thread_lock); \
mc_platform->acc->num_msg_threads--; \
COND_SIGNAL(&mc_platform->acc->msg_thread_cond); \
MUTEX_UNLOCK(&mc_platform->acc->msg_thread_lock); \
THREAD_EXIT()
```

Definition at line 590 of file message.c.

Referenced by `message_send_Thread()`.

13.56.1.2 #define MSG_THREADS 40

Definition at line 561 of file message.c.

Referenced by `message_Send()`.

13.56.1.3 #define SOCKET_INPUT_SIZE 4096

Definition at line 61 of file message.c.

13.56.2 Function Documentation

13.56.2.1 int auth_rece_send_msg (int sockfd, char * hostname, char * message, char * privkey, char * known_host_filename)

Definition at line 437 of file message.c.

References `aes_en_de()`, `initiate_migration_process()`, `read_known_host_file()`, and `send_AES_en_MA()`.

Referenced by `message_send_Thread()`.

13.56.2.2 int http_to_hostport (const char * http_str, char ** host, int * port, char ** target)

Definition at line 287 of file message.c.

References `MC_ERR_PARSE`.

Referenced by `MC_AclSend()`.

13.56.2.3 `message_p message_Copy (message_p src)`

Definition at line 88 of file `message.c`.

13.56.2.4 `int message_Destroy (message_p message)`

Definition at line 398 of file `message.c`.

References `message_s::addr`, `message_s::agent_xml_flag`, `message_s::from_address`, `MC_SUCCESS`, `message_s::message_body`, `mxmlDelete()`, `message_s::target`, `message_s::to_address`, `message_s::update_name`, and `message_s::xml_root`.

Referenced by `acc_connection_Thread()`, `acc_MessageHandlerThread()`, `ams_ManageAgentList()`, `MC_LoadAgentFromFile()`, `MC_SendAgentMigrationMessage()`, `MC_SendAgentMigrationMessageFile()`, `message_InitializeFromConnection()`, and `message_InitializeFromString()`.

13.56.2.5 `int message_InitializeFromAgent (mc_platform_p mc_platform, message_p message, agent_p agent)`

Definition at line 96 of file `message.c`.

References `message_s::addr`, `agent_xml_compose()`, `message_s::agent_xml_flag`, `buf`, `CHECK_NULL`, `agent_s::datastate`, `mc_platform_s::err`, `message_s::from_address`, `agent_s::home`, `mc_platform_s::hostname`, `MC_ERR_MEMORY`, `MC_SUCCESS`, `message_s::message_body`, `message_s::message_id`, `message_s::message_type`, `MOBILE_AGENT`, `MXML_NO_CALLBACK`, `mxmlSaveAllocString()`, `agent_s::name`, `agent_datastate_s::number_of_tasks`, `mc_platform_s::port`, `message_s::sending_agent_name`, `agent_task_s::server_name`, `strtok_r`, `message_s::target`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, `message_s::to_address`, `message_s::update_name`, `WARN`, and `message_s::xml_root`.

13.56.2.6 `int message_InitializeFromConnection (mc_platform_p mc_platform, message_p message, connection_p connection)`

Definition at line 196 of file `message.c`.

References `connection_s::addr`, `message_s::addr`, `CHECK_NULL`, `connection_s::clientfd`, `connection_s::connect_id`, `message_s::connect_id`, `message_s::from_address`, `MC_ERR_CONNECT`, `MC_ERR_PARSE`, `MC_SUCCESS`, `message_s::message_body`, `message_Destroy()`, `message_s::message_id`, `message_xml_parse()`, `MXML_NO_CALLBACK`, `mxmlLoadString()`, `SOCKET_ERROR`, `SOCKET_INPUT_SIZE`, `message_s::target`, `message_s::to_address`, and `message_s::xml_root`.

13.56.2.7 `int message_InitializeFromString (mc_platform_p mc_platform, message_p message, const char * string, const char * destination_host, int destination_port, const char * target)`

Definition at line 322 of file `message.c`.

References `message_s::addr`, `CHECK_NULL`, `message_s::connect_id`, `mc_platform_s::err`, `message_s::from_address`, `mc_platform_s::hostname`, `MC_ERR_MEMORY`, `MC_ERR_PARSE`, `MC_SUCCESS`, `message_s::message_body`, `message_Destroy()`, `message_s::message_id`, `message_s::message_type`, `message_xml_parse()`, `MOBILE_AGENT`, `MXML_NO_CALLBACK`, `mxmlLoadString()`, `mc_platform_s::port`, `message_s::target`, `message_s::to_address`, `message_s::update_name`, and `message_s::xml_root`.

13.56.2.8 message_p message_New (void)

Definition at line 64 of file message.c.

References message_s::addr, message_s::agent_xml_flag, CHECK_NULL, message_s::connect_id, message_s::from_address, message_s::http_type, message_s::isHTTP, message_s::message_body, message_s::message_id, message_s::message_type, message_s::target, message_s::to_address, message_s::update_name, message_s::update_num, message_s::xml_payload, and message_s::xml_root.

Referenced by acc_connection_Thread(), ams_ManageAgentList(), MC_LoadAgentFromFile(), MC_SendAgentMigrationMessage(), MC_SendAgentMigrationMessageFile(), and mtp_http_CreateMessage().

13.56.2.9 int message_Send (mc_platform_t * mc_platform, message_p message, char * privatekey)

Definition at line 562 of file message.c.

References mc_platform_s::acc, COND_WAIT, message_send_arg_s::mc_platform, message_send_arg_s::message, message_send_Thread(), MSG_THREADS, MUTEX_LOCK, MUTEX_UNLOCK, message_send_arg_s::privatekey, THREAD_CREATE, THREAD_DETACH, and THREAD_T.

Referenced by acc_MessageHandlerThread(), MC_AclSend(), and message_queue_SendOutgoing().

13.56.2.10 void* message_send_Thread (void * arg)

Definition at line 601 of file message.c.

References mc_platform_s::agency, auth_rece_send_msg(), buf, CHECK_NULL, dynstring_Append(), dynstring_Destroy(), dynstring_New(), agency_s::known_host_filename, mc_platform, dynstring_s::message, message_s::message_body, MSG_THREAD_EXIT, mtp_http_ComposeMessage(), mtp_http_Destroy(), mtp_http_New(), mtp_http_Parse(), port, send, SOCKET_ERROR, SOCKET_INPUT_SIZE, strtok_r, and message_s::to_address.

Referenced by message_Send().

13.57 /home/dko/Projects/mobilec/trunk/src/message_queue.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include "include/data_structures.h"
#include "include/mc_platform.h"
```

Functions

- void [message_queue_SendOutgoing](#) (struct [mc_platform_s](#) *[mc_platform](#), [message_queue_p](#) [mqueue](#))

13.57.1 Function Documentation

13.57.1.1 void [message_queue_SendOutgoing](#) (struct [mc_platform_s](#) * [mc_platform](#), [message_queue_p](#) [mqueue](#))

Definition at line 57 of file [message_queue.c](#).

References [CHECK_NULL](#), [mc_platform_s::hostname](#), [ListSearch\(\)](#), [message_Send\(\)](#), [MUTEX_LOCK](#), [MUTEX_UNLOCK](#), [mc_platform_s::port](#), and [message_s::to_address](#).

13.58 /home/dko/Projects/mobilec/trunk/src/mtp_http.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <unistd.h>
#include <errno.h>
#include "config.h"
#include "include/connection.h"
#include "include/mtp_http.h"
#include "include/macros.h"
#include "include/mc_error.h"
#include "include/message.h"
#include "include/dynstring.h"
```

Defines

- #define [SAFE_FREE](#)(elem)
- #define [HTTP_PARSE_EXPR](#)(parse_name, struct_name)
- #define [SAFE_FREE](#)(object)

Functions

- [int mtp_http_Destroy](#) ([mtp_http_p](#) http)
- [mtp_http_p mtp_http_New](#) (void)
- [int rece_de_msg](#) (char *buffer, [connection_p](#) con, char *privatekey)
- [int mtp_http_InitializeFromConnection](#) ([mtp_http_p](#) http, [connection_p](#) connection, char *privatekey)
- const char * [http_GetExpression](#) (const char *string, char **expr)

Parse an html expression.

- [int http_ParseExpression](#) (const char *expression_string, char **name, char **value)

Parse an expression into its name and value.

- const char * [mtp_http_ParseHeader](#) (struct [mtp_http_s](#) *http, const char *string)
- [int mtp_http_Parse](#) (struct [mtp_http_s](#) *http, const char *string)
- const char * [http_ParseRequest](#) ([mtp_http_p](#) http, const char *string)
- const char * [http_GetToken](#) (const char *string, char **token)
- [int mtp_http_ParseResponse](#) (struct [mtp_http_s](#) *http, const char *string)
- [int mtp_http_ComposeMessage](#) ([message_p](#) message)
- struct [message_s](#) * [mtp_http_CreateMessage](#) ([mtp_http_t](#) *mtp_http, char *hostname, [int](#) port)

13.58.1 Define Documentation

13.58.1.1 #define HTTP_PARSE_EXPR(parse_name, struct_name)

Value:

```
if ( !strcmp(name, parse_name) ) { \
    http->struct_name = (char*)malloc \
    ( \
        sizeof(char) * \
        (strlen(value)+1) \
    ); \
    strcpy(http->struct_name, value); \
} else
```

Referenced by mtp_http_ParseHeader().

13.58.1.2 #define SAFE_FREE(object)

Value:

```
if(object) free(object); \
    object = NULL
```

13.58.1.3 #define SAFE_FREE(elem)

Value:

```
if(elem) \
    free(elem)
```

Referenced by mtp_http_Destroy(), and mtp_http_ParseHeader().

13.58.2 Function Documentation

13.58.2.1 const char* http_GetExpression (const char * *string*, char ** *expr*)

Parse an html expression.

Parameters:

string (input) The html block of text: Will parse the first expression pointed to by 'string'.

expr (output) The allocated expression

Returns:

A pointer to the next expression segment of the string block, or NULL.

Definition at line 347 of file mtp_http.c.

Referenced by mtp_http_Parse(), and mtp_http_ParseHeader().

13.58.2.2 const char* http_GetToken (const char * *string*, char ** *token*)

Definition at line 788 of file mtp_http.c.

References cur.

Referenced by http_ParseRequest().

13.58.2.3 int http_ParseExpression (const char * *expression_string*, char ** *name*, char ** *value*)

Parse an expression into its name and value.

Parameters:

expression_string (input) The expression

name (output) An allocated name string or NULL

value (output) An allocated value string or NULL

Returns:

error_code_t type

Note:

an http expression is something like 'Date: Mon, 23 May 2005 22:38:34 GMT
' where 'Date' is the name and the remainder of the string is the value

Definition at line 406 of file mtp_http.c.

References CHECK_NULL, MC_ERR_PARSE, and MC_SUCCESS.

Referenced by mtp_http_Parse(), and mtp_http_ParseHeader().

13.58.2.4 const char* http_ParseRequest (mtp_http_p *http*, const char * *string*)

Definition at line 699 of file mtp_http.c.

References cur, HTTP_CONNECT, HTTP_DELETE, HTTP_GET, http_GetToken(), HTTP_HEAD, HTTP_OPTIONS, mtp_http_s::http_performative, HTTP_PERFORMATIVE_UNDEF, HTTP_POST, HTTP_PUT, HTTP_RESPONSE, HTTP_TRACE, mtp_http_s::response_code, mtp_http_s::response_string, and mtp_http_s::target.

Referenced by mtp_http_ParseHeader().

13.58.2.5 int mtp_http_ComposeMessage (message_p *message*)

Definition at line 827 of file mtp_http.c.

References buf, message_s::isHTTP, MC_SUCCESS, message_s::message_body, PACKAGE_VERSION, message_s::target, and message_s::to_address.

13.58.2.6 struct message_s* mtp_http_CreateMessage (mtp_http_t * *mtp_http*, char * *hostname*, int *port*) [read]

Definition at line 873 of file mtp_http.c.

References `buf`, `mtp_http_s::content`, `mtp_http_content_s::content_type`, `mtp_http_content_s::data`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `mtp_http_s::host`, `message_s::isHTTP`, `dynstring_s::len`, `dynstring_s::message`, `message_s::message_body`, `message_New()`, `mtp_http_s::message_parts`, `PACKAGE_VERSION`, `mtp_http_s::target`, and `message_s::to_address`.

Referenced by `MC_AclSend()`.

13.58.2.7 `int mtp_http_Destroy (mtp_http_p http)`

Definition at line 54 of file `mtp_http.c`.

References `mtp_http_s::accept_ranges`, `mtp_http_s::boundary`, `mtp_http_s::connection`, `mtp_http_s::content`, `mtp_http_s::content_length`, `mtp_http_content_s::content_type`, `mtp_http_s::content_type`, `mtp_http_content_s::data`, `mtp_http_s::date`, `mtp_http_s::host`, `mtp_http_s::http_version`, `mtp_http_s::message_parts`, `mtp_http_s::response_string`, `mtp_http_s::return_code`, `SAFE_FREE`, `mtp_http_s::server`, `mtp_http_s::target`, and `mtp_http_s::user_agent`.

Referenced by `acc_connection_Thread()`, `MC_AclSend()`, `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

13.58.2.8 `int mtp_http_InitializeFromConnection (mtp_http_p http, connection_p connection, char *privatekey)`

Definition at line 178 of file `mtp_http.c`.

References `CHECK_NULL`, `connection_s::clientfd`, `mtp_http_s::content_length`, `dynstring_Append()`, `dynstring_Destroy()`, `dynstring_New()`, `ERR`, `mtp_http_s::header_length`, `HTTP_HEAD`, `mtp_http_s::http_performative`, `MC_ERR_CONNECT`, `mtp_http_Destroy()`, `mtp_http_New()`, `mtp_http_Parse()`, `mtp_http_ParseHeader()`, `PACKAGE_STRING`, `rece_de_msg()`, `send`, `SOCKET_ERROR`, and `SOCKET_INPUT_SIZE`.

13.58.2.9 `mtp_http_p mtp_http_New (void)`

Definition at line 87 of file `mtp_http.c`.

References `CHECK_NULL`, and `mtp_http_s::content`.

Referenced by `acc_connection_Thread()`, `MC_AclSend()`, `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

13.58.2.10 `int mtp_http_Parse (struct mtp_http_s * http, const char * string)`

Definition at line 549 of file `mtp_http.c`.

References `mtp_http_s::boundary`, `mtp_http_s::content`, `mtp_http_s::content_length`, `mtp_http_content_s::content_type`, `mtp_http_s::content_type`, `mtp_http_content_s::data`, `http_GetExpression()`, `HTTP_HEAD`, `http_ParseExpression()`, `mtp_http_s::http_performative`, `HTTP_POST`, `HTTP_PUT`, `HTTP_RESPONSE`, `MC_SUCCESS`, `mtp_http_s::message_parts`, and `mtp_http_ParseHeader()`.

Referenced by `message_send_Thread()`, and `mtp_http_InitializeFromConnection()`.

13.58.2.11 `const char* mtp_http_ParseHeader (struct mtp_http_s * http, const char * string)`

Definition at line 465 of file `mtp_http.c`.

References mtp_http_s::header_length, http_GetExpression(), HTTP_PARSE_EXPR, http_ParseExpression(), http_ParseRequest(), MC_SUCCESS, and SAFE_FREE.

Referenced by mtp_http_InitializeFromConnection(), and mtp_http_Parse().

13.58.2.12 int mtp_http_ParseResponse (struct mtp_http_s * *http*, const char * *string*)

Definition at line 821 of file mtp_http.c.

13.58.2.13 int rece_de_msg (char * *buffer*, connection_p *con*, char * *privatekey*)

Definition at line 98 of file mtp_http.c.

References aes_en_de(), connection_s::AES_key, connection_s::clientfd, int, connection_s::nonce, and receive_AES_en_MA().

Referenced by mtp_http_InitializeFromConnection().

13.59 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/config.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
```

Defines

- `#define MXML_VERSION "Mini-XML v2.2.2"`
- `#define HAVE_VSNPRINTF 1`
- `#define HAVE_STRDUP 1`

Functions

- `char * mxml_strdup (const char *, va_list)`

13.59.1 Define Documentation

13.59.1.1 `#define HAVE_STRDUP 1`

Definition at line 49 of file config.h.

13.59.1.2 `#define HAVE_VSNPRINTF 1`

Definition at line 42 of file config.h.

13.59.1.3 `#define MXML_VERSION "Mini-XML v2.2.2"`

Definition at line 35 of file config.h.

Referenced by `write_documentation()`.

13.59.2 Function Documentation

13.59.2.1 `char* mxml_strdup (const char *, va_list)`

Definition at line 62 of file mxml-string.c.

Referenced by `mxml_error()`, `mxmlNewTextf()`, and `mxmlSetTextf()`.

13.60 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/vcnet/config.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
```

Defines

- `#define MXML_VERSION "Mini-XML v2.0"`
- `#define HAVE_STRDUP 1`

Functions

- `char * mxml_strdup (const char *, va_list)`

13.60.1 Define Documentation

13.60.1.1 `#define HAVE_STRDUP 1`

Definition at line 51 of file config.h.

13.60.1.2 `#define MXML_VERSION "Mini-XML v2.0"`

Definition at line 37 of file config.h.

13.60.2 Function Documentation

13.60.2.1 `char* mxml_strdup (const char *, va_list)`

Definition at line 62 of file mxml-string.c.

13.61 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/vcnet2005/config.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
```

Defines

- `#define MXML_VERSION "Mini-XML v2.0"`
- `#define HAVE_STRDUP 1`

Functions

- `char * mxml_strdup (const char *, va_list)`

13.61.1 Define Documentation

13.61.1.1 `#define HAVE_STRDUP 1`

Definition at line 51 of file config.h.

13.61.1.2 `#define MXML_VERSION "Mini-XML v2.0"`

Definition at line 37 of file config.h.

13.61.2 Function Documentation

13.61.2.1 `char* mxml_strdup (const char *, va_list)`

Definition at line 62 of file mxml-string.c.

13.62 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/vcnet2008/config.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <ctype.h>
```

Defines

- `#define MXML_VERSION "Mini-XML v2.0"`
- `#define HAVE_STRDUP 1`

Functions

- `char * mxml_strdup (const char *, va_list)`

13.62.1 Define Documentation

13.62.1.1 `#define HAVE_STRDUP 1`

Definition at line 51 of file config.h.

13.62.1.2 `#define MXML_VERSION "Mini-XML v2.0"`

Definition at line 37 of file config.h.

13.62.2 Function Documentation

13.62.2.1 `char* mxml_strdup (const char *, va_list)`

Definition at line 62 of file mxml-string.c.

References `mxml_vsnprintf()`.

13.63 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/config.h File Reference

Defines

- `#define _CRT_SECURE_NO_DEPRECATED 1`
- `#define XYSSL_HAVE_ASM`
- `#define XYSSL_DEBUG_MSG`
- `#define XYSSL_SELF_TEST`
- `#define XYSSL_GENPRIME`
- `#define XYSSL_AES_C`
- `#define XYSSL_ARC4_C`
- `#define XYSSL_BASE64_C`
- `#define XYSSL_BIGNUM_C`
- `#define XYSSL_CERTS_C`
- `#define XYSSL_DEBUG_C`
- `#define XYSSL_DES_C`
- `#define XYSSL_DHM_C`
- `#define XYSSL_HAVEGE_C`
- `#define XYSSL_MD5_C`
- `#define XYSSL_NET_C`
- `#define XYSSL_PADLOCK_C`
- `#define XYSSL_RSA_C`
- `#define XYSSL_SHA1_C`
- `#define XYSSL_SHA2_C`
- `#define XYSSL_SHA4_C`
- `#define XYSSL_SSL_CLI_C`
- `#define XYSSL_SSL_SRV_C`
- `#define XYSSL_SSL_TLS_C`
- `#define XYSSL_TIMING_C`
- `#define XYSSL_X509_PARSE_C`
- `#define XYSSL_X509_WRITE_C`

13.63.1 Define Documentation

13.63.1.1 `#define _CRT_SECURE_NO_DEPRECATED 1`

Definition at line 12 of file config.h.

13.63.1.2 `#define XYSSL_AES_C`

Definition at line 74 of file config.h.

13.63.1.3 `#define XYSSL_ARC4_C`

Definition at line 84 of file config.h.

13.63.1.4 #define XYSSL_BASE64_C

Definition at line 92 of file config.h.

13.63.1.5 #define XYSSL_BIGNUM_C

Definition at line 103 of file config.h.

13.63.1.6 #define XYSSL_CERTS_C

Definition at line 111 of file config.h.

13.63.1.7 #define XYSSL_DEBUG_C

Definition at line 121 of file config.h.

13.63.1.8 #define XYSSL_DEBUG_MSG

Definition at line 47 of file config.h.

13.63.1.9 #define XYSSL_DES_C

Definition at line 131 of file config.h.

13.63.1.10 #define XYSSL_DHM_C

Definition at line 142 of file config.h.

13.63.1.11 #define XYSSL_GENPRIME

Definition at line 57 of file config.h.

13.63.1.12 #define XYSSL_HAVE_ASM

Definition at line 36 of file config.h.

13.63.1.13 #define XYSSL_HAVEGE_C

Definition at line 150 of file config.h.

13.63.1.14 #define XYSSL_MD5_C

Definition at line 177 of file config.h.

13.63.1.15 #define XYSSL_NET_C

Definition at line 185 of file config.h.

13.63.1.16 #define XYSSL_PADLOCK_C

Definition at line 193 of file config.h.

13.63.1.17 #define XYSSL_RSA_C

Definition at line 204 of file config.h.

13.63.1.18 #define XYSSL_SELF_TEST

Definition at line 52 of file config.h.

13.63.1.19 #define XYSSL_SHA1_C

Definition at line 215 of file config.h.

13.63.1.20 #define XYSSL_SHA2_C

Definition at line 223 of file config.h.

13.63.1.21 #define XYSSL_SHA4_C

Definition at line 231 of file config.h.

13.63.1.22 #define XYSSL_SSL_CLI_C

Definition at line 239 of file config.h.

13.63.1.23 #define XYSSL_SSL_SRV_C

Definition at line 247 of file config.h.

13.63.1.24 #define XYSSL_SSL_TLS_C

Definition at line 256 of file config.h.

13.63.1.25 #define XYSSL_TIMING_C

Definition at line 264 of file config.h.

13.63.1.26 `#define XYSSL_X509_PARSE_C`

Definition at line 274 of file config.h.

13.63.1.27 `#define XYSSL_X509_WRITE_C`

Definition at line 282 of file config.h.

13.64 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-attr.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- const char * [mxmlElementGetAttr](#) (mxml_node_t *node, const char *name)
- void [mxmlElementSetAttr](#) (mxml_node_t *node, const char *name, const char *value)

13.64.1 Function Documentation

13.64.1.1 const char* mxmlElementGetAttr (mxml_node_t * node, const char * name)

Definition at line 43 of file mxml-attr.c.

References [mxml_value_s::attrs](#), [mxml_value_u::element](#), [MXML_ELEMENT](#), [mxml_attr_s::name](#), [mxml_value_s::num_attrs](#), [mxml_node_s::type](#), [mxml_attr_s::value](#), and [mxml_node_s::value](#).

Referenced by [agent_xml_parse__agent_code\(\)](#), [agent_xml_parse__data\(\)](#), [agent_xml_parse__task\(\)](#), [agent_xml_parse__tasks\(\)](#), [fipa_envelope_HandleReceived\(\)](#), [index_compare\(\)](#), [index_find\(\)](#), [message_xml_parse__message\(\)](#), [mxml_parse_element\(\)](#), [mxmlFindElement\(\)](#), [mxmlIndexNew\(\)](#), [scan_file\(\)](#), [sort_node\(\)](#), [type_cb\(\)](#), and [write_documentation\(\)](#).

13.64.1.2 void mxmlElementSetAttr (mxml_node_t * node, const char * name, const char * value)

Definition at line 90 of file mxml-attr.c.

References [mxml_value_s::attrs](#), [mxml_value_u::element](#), [MXML_ELEMENT](#), [mxml_error\(\)](#), [mxml_value_s::name](#), [mxml_attr_s::name](#), [mxml_value_s::num_attrs](#), [mxml_node_s::type](#), [mxml_attr_s::value](#), and [mxml_node_s::value](#).

Referenced by [add_variable\(\)](#), [agent_xml_compose__agent_code\(\)](#), [agent_xml_compose__create_row_nodes\(\)](#), [agent_xml_compose__data\(\)](#), [agent_xml_compose__message\(\)](#), [agent_xml_compose__task\(\)](#), [agent_xml_compose__tasks\(\)](#), [fipa_envelope_Compose__params\(\)](#), [main\(\)](#), [mxml_parse_element\(\)](#), [scan_file\(\)](#), [sort_node\(\)](#), and [update_comment\(\)](#).

13.65 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-entity.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- static [int default_callback](#) (const char *name)
- [int mxmlEntityAddCallback](#) (int(*cb)(const char *name))
- const char * [mxmlEntityGetName](#) (int val)
- [int mxmlEntityGetValue](#) (const char *name)
- void [mxmlEntityRemoveCallback](#) (int(*cb)(const char *name))

Variables

- static [int num_callbacks](#) = 1
- static [int\(* callbacks](#) [100])(const char *name)

13.65.1 Function Documentation

13.65.1.1 static int default_callback (const char * *name*) [static]

Definition at line 169 of file mxml-entity.c.

References [int](#).

13.65.1.2 int mxmlEntityAddCallback (int(*) (const char *name) *cb*)

Definition at line 65 of file mxml-entity.c.

References [callbacks](#), [mxml_error\(\)](#), and [num_callbacks](#).

13.65.1.3 const char* mxmlEntityGetName (int *val*)

Definition at line 91 of file mxml-entity.c.

Referenced by [mxml_write_name\(\)](#), and [mxml_write_string\(\)](#).

13.65.1.4 int mxmlEntityGetValue (const char * *name*)

Definition at line 121 of file mxml-entity.c.

References [callbacks](#), and [num_callbacks](#).

Referenced by [mxml_get_entity\(\)](#).

13.65.1.5 void mxmlEntityRemoveCallback (int(*) (const char *name) cb)

Definition at line 140 of file mxml-entity.c.

References `callbacks`, and `num_callbacks`.

13.65.2 Variable Documentation**13.65.2.1 int(* callbacks[100])(const char *name) [static]**

Initial value:

```
{
    default_callback
}
```

Referenced by `mxmlEntityAddCallback()`, `mxmlEntityGetValue()`, and `mxmlEntityRemoveCallback()`.

13.65.2.2 int num_callbacks = 1 [static]

Definition at line 53 of file mxml-entity.c.

Referenced by `mxmlEntityAddCallback()`, `mxmlEntityGetValue()`, and `mxmlEntityRemoveCallback()`.

13.66 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-file.c File Reference

```
#include "config.h"
#include "mxml.h"
#include <unistd.h>
```

Data Structures

- struct [mxml_fdbuf_s](#)

Defines

- #define [ENCODE_UTF8](#) 0
- #define [ENCODE_UTF16BE](#) 1
- #define [ENCODE_UTF16LE](#) 2
- #define [mxml_bad_char](#)(ch) ((ch) < ' ' && (ch) != '\n' && (ch) != '\r' && (ch) != '\t')

Typedefs

- typedef struct [mxml_fdbuf_s](#) [mxml_fdbuf_t](#)

Functions

- static [int](#) [mxml_add_char](#) ([int](#) ch, [char](#) **ptr, [char](#) **buffer, [int](#) *bufsize)
- static [int](#) [mxml_fd_getc](#) ([void](#) *p, [int](#) *encoding)
- static [int](#) [mxml_fd_putc](#) ([int](#) ch, [void](#) *p)
- static [int](#) [mxml_fd_read](#) ([mxml_fdbuf_t](#) *buf)
- static [int](#) [mxml_fd_write](#) ([mxml_fdbuf_t](#) *buf)
- static [int](#) [mxml_file_getc](#) ([void](#) *p, [int](#) *encoding)
- static [int](#) [mxml_file_putc](#) ([int](#) ch, [void](#) *p)
- static [int](#) [mxml_get_entity](#) ([mxml_node_t](#) *parent, [void](#) *p, [int](#) *encoding, [int](#)(*getc_cb)([void](#) *, [int](#) *))
- static [mxml_node_t](#) * [mxml_load_data](#) ([mxml_node_t](#) *top, [void](#) *p, [mxml_type_t](#)(*cb)([mxml_node_t](#) *), [int](#)(*getc_cb)([void](#) *, [int](#) *))
- static [int](#) [mxml_parse_element](#) ([mxml_node_t](#) *node, [void](#) *p, [int](#) *encoding, [int](#)(*getc_cb)([void](#) *, [int](#) *))
- static [int](#) [mxml_string_getc](#) ([void](#) *p, [int](#) *encoding)
- static [int](#) [mxml_string_putc](#) ([int](#) ch, [void](#) *p)
- static [int](#) [mxml_write_name](#) ([const](#) [char](#) *s, [void](#) *p, [int](#)(*putc_cb)([int](#), [void](#) *))
- static [int](#) [mxml_write_node](#) ([mxml_node_t](#) *node, [void](#) *p, [const](#) [char](#) *(*cb)([mxml_node_t](#) *, [int](#)), [int](#) col, [int](#)(*putc_cb)([int](#), [void](#) *))
- static [int](#) [mxml_write_string](#) ([const](#) [char](#) *s, [void](#) *p, [int](#)(*putc_cb)([int](#), [void](#) *))
- static [int](#) [mxml_write_ws](#) ([mxml_node_t](#) *node, [void](#) *p, [const](#) [char](#) *(*cb)([mxml_node_t](#) *, [int](#)), [int](#) ws, [int](#) col, [int](#)(*putc_cb)([int](#), [void](#) *))
- [mxml_node_t](#) * [mxmlLoadFd](#) ([mxml_node_t](#) *top, [int](#) fd, [mxml_type_t](#)(*cb)([mxml_node_t](#) *node))
- [mxml_node_t](#) * [mxmlLoadFile](#) ([mxml_node_t](#) *top, [FILE](#) *fp, [mxml_type_t](#)(*cb)([mxml_node_t](#) *node))

- `mxml_node_t * mxmlloadString (mxml_node_t *top, const char *s, mxml_type_t(*cb)(mxml_node_t *node))`
- `char * mxmlSaveAllocString (mxml_node_t *node, const char *(*cb)(mxml_node_t *node, int ws))`
- `int mxmlSaveFd (mxml_node_t *node, int fd, const char *(*cb)(mxml_node_t *node, int ws))`
- `int mxmlSaveFile (mxml_node_t *node, FILE *fp, const char *(*cb)(mxml_node_t *node, int ws))`
- `int mxmlSaveString (mxml_node_t *node, char *buffer, int bufsize, const char *(*cb)(mxml_node_t *node, int ws))`
- `void mxmlSetCustomHandlers (mxml_custom_load_cb_t load, mxml_custom_save_cb_t save)`
- `void mxmlSetErrorCallback (void(*cb)(const char *))`

Variables

- `void(* mxml_error_cb)(const char *)`
- `static mxml_custom_load_cb_t mxml_custom_load_cb = NULL`
- `static mxml_custom_save_cb_t mxml_custom_save_cb = NULL`

13.66.1 Define Documentation

13.66.1.1 #define ENCODE_UTF16BE 1

Definition at line 68 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, and `mxml_string_getc()`.

13.66.1.2 #define ENCODE_UTF16LE 2

Definition at line 69 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, and `mxml_string_getc()`.

13.66.1.3 #define ENCODE_UTF8 0

Definition at line 67 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_load_data()`, and `mxml_string_getc()`.

13.66.1.4 #define mxml_bad_char(ch) ((ch) < ' ' && (ch) != '\n' && (ch) != '\r' && (ch) != '\t')

Definition at line 76 of file mxml-file.c.

Referenced by `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_get_entity()`, and `mxml_string_getc()`.

13.66.2 Typedef Documentation

13.66.2.1 typedef struct mxml_fdbuf_s mxml_fdbuf_t

13.66.3 Function Documentation

13.66.3.1 static int mxml_add_char (int *ch*, char ***ptr*, char ***buffer*, int **bufsize*) [static]

Definition at line 483 of file mxml-file.c.

References `mxml_error()`.

Referenced by `mxml_load_data()`, and `mxml_parse_element()`.

13.66.3.2 `static int mxml_fd_getc (void *p, int *encoding) [static]`

Definition at line 563 of file `mxml-file.c`.

References `buf`, `mxml_fdbuf_s::current`, `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_fdbuf_s::end`, `mxml_bad_char`, `mxml_error()`, and `mxml_fd_read()`.

Referenced by `mxmlLoadFd()`.

13.66.3.3 `static int mxml_fd_putc (int ch, void *p) [static]`

Definition at line 850 of file `mxml-file.c`.

References `buf`, `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, and `mxml_fd_write()`.

Referenced by `mxmlSaveFd()`.

13.66.3.4 `static int mxml_fd_read (mxml_fdbuf_t *buf) [static]`

Definition at line 919 of file `mxml-file.c`.

References `mxml_fdbuf_s::buffer`, `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, and `mxml_fdbuf_s::fd`.

Referenced by `mxml_fd_getc()`.

13.66.3.5 `static int mxml_fd_write (mxml_fdbuf_t *buf) [static]`

Definition at line 958 of file `mxml-file.c`.

References `mxml_fdbuf_s::buffer`, `mxml_fdbuf_s::current`, and `mxml_fdbuf_s::fd`.

Referenced by `mxml_fd_putc()`, and `mxmlSaveFd()`.

13.66.3.6 `static int mxml_file_getc (void *p, int *encoding) [static]`

Definition at line 1001 of file `mxml-file.c`.

References `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_bad_char`, and `mxml_error()`.

Referenced by `mxmlLoadFile()`.

13.66.3.7 `static int mxml_file_putc (int ch, void *p) [static]`

Definition at line 1200 of file `mxml-file.c`.

Referenced by `mxmlSaveFile()`.

13.66.3.8 `static int mxml_get_entity (mxml_node_t * parent, void * p, int * encoding, int(*)(void *, int *) getc_cb) [static]`

Definition at line 1255 of file mxml-file.c.

References `mxml_value_u::element`, `mxml_bad_char`, `mxml_error()`, `mxmlEntityGetValue()`, `mxml_value_s::name`, and `mxml_node_s::value`.

Referenced by `mxml_load_data()`, and `mxml_parse_element()`.

13.66.3.9 `static mxml_node_t * mxml_load_data (mxml_node_t * top, void * p, mxml_type_t(*)(mxml_node_t *) cb, int(*)(void *, int *) getc_cb) [static]`

Definition at line 1316 of file mxml-file.c.

References `mxml_value_u::element`, `ENCODE_UTF8`, `mxml_add_char()`, `MXML_CUSTOM`, `mxml_custom_load_cb`, `mxml_error()`, `mxml_get_entity()`, `MXML_INTEGER`, `MXML_OPAQUE`, `mxml_parse_element()`, `MXML_REAL`, `MXML_TEXT`, `mxmlDelete()`, `mxmlNewCustom()`, `mxmlNewElement()`, `mxmlNewInteger()`, `mxmlNewOpaque()`, `mxmlNewReal()`, `mxmlNewText()`, `mxml_value_s::name`, `mxml_node_s::parent`, and `mxml_node_s::value`.

Referenced by `mxmlLoadFd()`, `mxmlLoadFile()`, and `mxmlLoadString()`.

13.66.3.10 `static int mxml_parse_element (mxml_node_t * node, void * p, int * encoding, int(*)(void *, int *) getc_cb) [static]`

Definition at line 1840 of file mxml-file.c.

References `mxml_value_u::element`, `mxml_add_char()`, `mxml_error()`, `mxml_get_entity()`, `mxmlElementGetAttr()`, `mxmlElementSetAttr()`, `mxml_value_s::name`, and `mxml_node_s::value`.

Referenced by `mxml_load_data()`.

13.66.3.11 `static int mxml_string_getc (void * p, int * encoding) [static]`

Definition at line 2102 of file mxml-file.c.

References `ENCODE_UTF16BE`, `ENCODE_UTF16LE`, `ENCODE_UTF8`, `mxml_bad_char`, and `mxml_error()`.

Referenced by `mxmlLoadString()`.

13.66.3.12 `static int mxml_string_putc (int ch, void * p) [static]`

Definition at line 2338 of file mxml-file.c.

Referenced by `mxmlSaveString()`.

13.66.3.13 `static int mxml_write_name (const char * s, void * p, int(*)(int, void *) putc_cb) [static]`

Definition at line 2412 of file mxml-file.c.

References `mxmlEntityGetName()`.

Referenced by `mxml_write_node()`.

13.66.3.14 `static int mxml_write_node (mxml_node_t * node, void * p, const char
 ()(mxml_node_t *, int) cb, int col, int(*)(int, void *) putc_cb) [static]`

Definition at line 2487 of file mxml-file.c.

References `mxml_value_s::attrs`, `mxml_node_s::child`, `mxml_value_u::element`, `mxml_value_u::integer`, `MXML_CUSTOM`, `mxml_custom_save_cb`, `MXML_ELEMENT`, `MXML_INTEGER`, `MXML_OPAQUE`, `MXML_REAL`, `MXML_TEXT`, `MXML_WRAP`, `mxml_write_name()`, `mxml_write_string()`, `mxml_write_ws()`, `MXML_WS_AFTER_CLOSE`, `MXML_WS_AFTER_OPEN`, `MXML_WS_BEFORE_CLOSE`, `MXML_WS_BEFORE_OPEN`, `mxml_attr_s::name`, `mxml_value_s::name`, `mxml_node_s::next`, `mxml_value_s::num_attrs`, `mxml_value_u::opaque`, `mxml_node_s::prev`, `mxml_value_u::real`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, `mxml_attr_s::value`, `mxml_node_s::value`, and `mxml_text_s::whitespace`.

Referenced by `mxmlSaveFd()`, `mxmlSaveFile()`, and `mxmlSaveString()`.

13.66.3.15 `static int mxml_write_string (const char * s, void * p, int(*)(int, void *) putc_cb)
 [static]`

Definition at line 2768 of file mxml-file.c.

References `mxmlEntityGetName()`.

Referenced by `mxml_write_node()`.

13.66.3.16 `static int mxml_write_ws (mxml_node_t * node, void * p, const char *(*)(mxml_node_t
 , int) cb, int ws, int col, int()(int, void *) putc_cb) [static]`

Definition at line 2808 of file mxml-file.c.

References `MXML_TAB`.

Referenced by `mxml_write_node()`.

13.66.3.17 `mxml_node_t* mxmlLoadFd (mxml_node_t * top, int fd,
 mxml_type_t(*)(mxml_node_t *node) cb)`

Definition at line 159 of file mxml-file.c.

References `buf`, `mxml_fdbuf_s::buffer`, `mxml_fdbuf_s::current`, `mxml_fdbuf_s::end`, `mxml_fdbuf_s::fd`, `mxml_fd_getc()`, and `mxml_load_data()`.

Referenced by `main()`.

13.66.3.18 `mxml_node_t* mxmlLoadFile (mxml_node_t * top, FILE * fp,
 mxml_type_t(*)(mxml_node_t *node) cb)`

Definition at line 199 of file mxml-file.c.

References `mxml_file_getc()`, and `mxml_load_data()`.

Referenced by `main()`.

13.66.3.19 **mxmxml_node_t* mxmlLoadString (mxmxml_node_t * *top*, const char * *s*, mxml_type_t (*)(mxmxml_node_t **node*) *cb*)**

Definition at line 228 of file mxml-file.c.

References mxml_load_data(), and mxml_string_getc().

Referenced by acc_connection_Thread(), agent_xml_compose(), fipa_envelope_Compose(), fipa_envelope_Parse(), main(), MC_LoadAgentFromFile(), message_InitializeFromConnection(), and message_InitializeFromString().

13.66.3.20 **char* mxmlSaveAllocString (mxmxml_node_t * *node*, const char ** (mxmxml_node_t **node*, int *ws*) *cb*)**

Definition at line 258 of file mxml-file.c.

References mxmlSaveString().

Referenced by fipa_envelope_Compose(), MC_GetAgentXMLString(), and message_InitializeFromAgent().

13.66.3.21 **int mxmlSaveFd (mxmxml_node_t * *node*, int *fd*, const char ** (mxmxml_node_t **node*, int *ws*) *cb*)**

Definition at line 315 of file mxml-file.c.

References buf, mxml_fdbuf_s::buffer, mxml_fdbuf_s::current, mxml_fdbuf_s::end, mxml_fdbuf_s::fd, mxml_fd_putc(), mxml_fd_write(), and mxml_write_node().

Referenced by main().

13.66.3.22 **int mxmlSaveFile (mxmxml_node_t * *node*, FILE * *fp*, const char ** (mxmxml_node_t **node*, int *ws*) *cb*)**

Definition at line 362 of file mxml-file.c.

References mxml_file_putc(), and mxml_write_node().

Referenced by main().

13.66.3.23 **int mxmlSaveString (mxmxml_node_t * *node*, char * *buffer*, int *bufsize*, const char ** (mxmxml_node_t **node*, int *ws*) *cb*)**

Definition at line 404 of file mxml-file.c.

References mxml_string_putc(), and mxml_write_node().

Referenced by main(), and mxmlSaveAllocString().

13.66.3.24 **void mxmlSetCustomHandlers (mxml_custom_load_cb_t *load*, mxml_custom_save_cb_t *save*)**

Definition at line 456 of file mxml-file.c.

References mxml_custom_load_cb, and mxml_custom_save_cb.

13.66.3.25 void mxmlSetErrorCallback (void(*) (const char *) cb)

Definition at line 471 of file mxml-file.c.

References mxml_error_cb.

13.66.4 Variable Documentation**13.66.4.1 mxml_custom_load_cb_t mxml_custom_load_cb = NULL [static]**

Definition at line 103 of file mxml-file.c.

Referenced by mxml_load_data(), and mxmlSetCustomHandlers().

13.66.4.2 mxml_custom_save_cb_t mxml_custom_save_cb = NULL [static]

Definition at line 104 of file mxml-file.c.

Referenced by mxml_write_node(), and mxmlSetCustomHandlers().

13.66.4.3 void(* mxml_error_cb)(const char *)

Referenced by mxml_error(), and mxmlSetErrorCallback().

13.67 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-index.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- static [int index_compare](#) ([mxml_index_t](#) *ind, [mxml_node_t](#) *first, [mxml_node_t](#) *second)
- static [int index_find](#) ([mxml_index_t](#) *ind, const char *element, const char *value, [mxml_node_t](#) *node)
- static void [index_sort](#) ([mxml_index_t](#) *ind, int left, int right)
- void [mxmlIndexDelete](#) ([mxml_index_t](#) *ind)
- [mxml_node_t](#) * [mxmlIndexEnum](#) ([mxml_index_t](#) *ind)
- [mxml_node_t](#) * [mxmlIndexFind](#) ([mxml_index_t](#) *ind, const char *element, const char *value)
- [mxml_index_t](#) * [mxmlIndexNew](#) ([mxml_node_t](#) *node, const char *element, const char *attr)
- [mxml_node_t](#) * [mxmlIndexReset](#) ([mxml_index_t](#) *ind)

13.67.1 Function Documentation

13.67.1.1 static int [index_compare](#) ([mxml_index_t](#) * *ind*, [mxml_node_t](#) * *first*, [mxml_node_t](#) * *second*) [**static**]

Definition at line 494 of file [mxml-index.c](#).

References [mxml_index_s::attr](#), [mxml_value_u::element](#), [mxmlElementGetAttr\(\)](#), [mxml_value_s::name](#), and [mxml_node_s::value](#).

Referenced by [index_sort\(\)](#).

13.67.1.2 static int [index_find](#) ([mxml_index_t](#) * *ind*, const char * *element*, const char * *value*, [mxml_node_t](#) * *node*) [**static**]

Definition at line 533 of file [mxml-index.c](#).

References [mxml_index_s::attr](#), [mxml_value_u::element](#), [mxmlElementGetAttr\(\)](#), [mxml_value_s::name](#), and [mxml_node_s::value](#).

Referenced by [mxmlIndexFind\(\)](#).

13.67.1.3 static void [index_sort](#) ([mxml_index_t](#) * *ind*, int *left*, int *right*) [**static**]

Definition at line 576 of file [mxml-index.c](#).

References [index_compare\(\)](#), and [mxml_index_s::nodes](#).

Referenced by [mxmlIndexNew\(\)](#).

13.67.1.4 void [mxmlIndexDelete](#) ([mxml_index_t](#) * *ind*)

Definition at line 58 of file [mxml-index.c](#).

References `mxml_index_s::alloc_nodes`, `mxml_index_s::attr`, and `mxml_index_s::nodes`.

Referenced by `main()`, and `mxmlIndexNew()`.

13.67.1.5 `mxml_node_t* mxmlIndexEnum (mxml_index_t * ind)`

Definition at line 88 of file `mxml-index.c`.

References `mxml_index_s::cur_node`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`, and `mxmlIndexFind()`.

13.67.1.6 `mxml_node_t* mxmlIndexFind (mxml_index_t * ind, const char * element, const char * value)`

Definition at line 118 of file `mxml-index.c`.

References `mxml_index_s::attr`, `mxml_index_s::cur_node`, `index_find()`, `mxmlIndexEnum()`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`.

13.67.1.7 `mxml_index_t* mxmlIndexNew (mxml_node_t * node, const char * element, const char * attr)`

Definition at line 301 of file `mxml-index.c`.

References `mxml_index_s::alloc_nodes`, `mxml_index_s::attr`, `mxml_value_u::element`, `index_sort()`, `MXML_DESCEND`, `mxml_error()`, `mxmlElementGetAttr()`, `mxmlFindElement()`, `mxmlIndexDelete()`, `mxml_value_s::name`, `mxml_index_s::nodes`, `mxml_index_s::num_nodes`, and `mxml_node_s::value`.

Referenced by `main()`.

13.67.1.8 `mxml_node_t* mxmlIndexReset (mxml_index_t * ind)`

Definition at line 459 of file `mxml-index.c`.

References `mxml_index_s::cur_node`, `mxml_index_s::nodes`, and `mxml_index_s::num_nodes`.

Referenced by `main()`.

13.68 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-node.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- static `mxml_node_t * mxml_new (mxml_node_t *parent, mxml_type_t type)`
- void `mxmlAdd (mxml_node_t *parent, int where, mxml_node_t *child, mxml_node_t *node)`
- void `mxmlDelete (mxml_node_t *node)`
- `mxml_node_t * mxmlNewCustom (mxml_node_t *parent, void *data, void(*destroy)(void *))`
- `mxml_node_t * mxmlNewElement (mxml_node_t *parent, const char *name)`
- `mxml_node_t * mxmlNewInteger (mxml_node_t *parent, int integer)`
- `mxml_node_t * mxmlNewOpaque (mxml_node_t *parent, const char *opaque)`
- `mxml_node_t * mxmlNewReal (mxml_node_t *parent, double real)`
- `mxml_node_t * mxmlNewText (mxml_node_t *parent, int whitespace, const char *string)`
- `mxml_node_t * mxmlNewTextf (mxml_node_t *parent, int whitespace, const char *format,...)`
- void `mxmlRemove (mxml_node_t *node)`

13.68.1 Function Documentation

13.68.1.1 static `mxml_node_t * mxml_new (mxml_node_t * parent, mxml_type_t type)` [static]

Definition at line 617 of file mxml-node.c.

References MXML_ADD_AFTER, MXML_ADD_TO_PARENT, mxmlAdd(), node, and mxml_node_s::type.

Referenced by mxmlNewCustom(), mxmlNewElement(), mxmlNewInteger(), mxmlNewOpaque(), mxmlNewReal(), mxmlNewText(), and mxmlNewTextf().

13.68.1.2 void `mxmlAdd (mxml_node_t * parent, int where, mxml_node_t * child, mxml_node_t * node)`

Definition at line 62 of file mxml-node.c.

References mxml_node_s::child, mxml_node_s::last_child, MXML_ADD_AFTER, MXML_ADD_BEFORE, mxmlRemove(), mxml_node_s::next, mxml_node_s::parent, and mxml_node_s::prev.

Referenced by add_variable(), agent_xml_compose(), agent_xml_compose__agent_data(), agent_xml_compose__create_row_nodes(), agent_xml_compose__data(), agent_xml_compose__gaf_message(), agent_xml_compose__message(), agent_xml_compose__mobile_agent(), agent_xml_compose__task(), agent_xml_compose__tasks(), fipa_envelope_Compose(), fipa_envelope_Compose__envelope(), fipa_envelope_Compose__params(), mxml_new(), scan_file(), and sort_node().

13.68.1.3 void `mxmlDelete (mxml_node_t * node)`

Definition at line 196 of file mxml-node.c.

References `mxml_value_s::attrs`, `mxml_node_s::child`, `mxml_value_u::custom`, `mxml_custom_s::data`, `mxml_custom_s::destroy`, `mxml_value_u::element`, `MXML_CUSTOM`, `MXML_ELEMENT`, `MXML_INTEGER`, `MXML_OPAQUE`, `MXML_REAL`, `MXML_TEXT`, `mxmlDelete()`, `mxmlRemove()`, `mxml_attr_s::name`, `mxml_value_s::name`, `mxml_value_s::num_attrs`, `mxml_value_u::opaque`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, `mxml_attr_s::value`, and `mxml_node_s::value`.

Referenced by `add_variable()`, `agent_datastate_Destroy()`, `fipa_envelope_Compose()`, `fipa_envelope_Parse()`, `main()`, `message_Destroy()`, `mxml_load_data()`, `mxmlDelete()`, `scan_file()`, and `sort_node()`.

13.68.1.4 `mxml_node_t*` `mxmlNewCustom` (`mxml_node_t` * *parent*, `void` * *data*, `void`(*)(`void` *) *destroy*)

Definition at line 287 of file `mxml-node.c`.

References `mxml_value_u::custom`, `mxml_custom_s::data`, `mxml_custom_s::destroy`, `MXML_CUSTOM`, `mxml_new()`, `node`, and `mxml_node_s::value`.

Referenced by `mxml_load_data()`.

13.68.1.5 `mxml_node_t*` `mxmlNewElement` (`mxml_node_t` * *parent*, `const char` * *name*)

Definition at line 323 of file `mxml-node.c`.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_new()`, `mxml_value_s::name`, `node`, and `mxml_node_s::value`.

Referenced by `add_variable()`, `agent_xml_compose__agent_code()`, `agent_xml_compose__agent_data()`, `agent_xml_compose__create_row_nodes()`, `agent_xml_compose__data()`, `agent_xml_compose__gaf_message()`, `agent_xml_compose__home()`, `agent_xml_compose__message()`, `agent_xml_compose__mobile_agent()`, `agent_xml_compose__name()`, `agent_xml_compose__owner()`, `agent_xml_compose__task()`, `agent_xml_compose__tasks()`, `agent_xml_compose__wg_code()`, `fipa_envelope_Compose__acl_representation()`, `fipa_envelope_Compose__date()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__params()`, `fipa_envelope_Compose__payload_encoding()`, `fipa_envelope_Compose__to()`, `main()`, `mxml_load_data()`, `scan_file()`, and `xml_new_cdata()`.

13.68.1.6 `mxml_node_t*` `mxmlNewInteger` (`mxml_node_t` * *parent*, `int` *integer*)

Definition at line 361 of file `mxml-node.c`.

References `mxml_value_u::integer`, `MXML_INTEGER`, `mxml_new()`, `node`, and `mxml_node_s::value`.

Referenced by `main()`, and `mxml_load_data()`.

13.68.1.7 `mxml_node_t*` `mxmlNewOpaque` (`mxml_node_t` * *parent*, `const char` * *opaque*)

Definition at line 392 of file `mxml-node.c`.

References `mxml_new()`, `MXML_OPAQUE`, `node`, `mxml_value_u::opaque`, and `mxml_node_s::value`.

Referenced by `main()`, and `mxml_load_data()`.

13.68.1.8 `mxml_node_t*` `mxmlNewReal` (`mxml_node_t` * *parent*, `double` *real*)

Definition at line 430 of file `mxml-node.c`.

References `mxml_new()`, `MXML_REAL`, `node`, `mxml_value_u::real`, and `mxml_node_s::value`.

Referenced by `main()`, and `mxml_load_data()`.

13.68.1.9 `mxml_node_t*` `mxmlNewText` (`mxml_node_t` * *parent*, `int` *whitespace*, `const char` * *string*)

Definition at line 462 of file `mxml-node.c`.

References `mxml_new()`, `MXML_TEXT`, `node`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::value`, and `mxml_text_s::whitespace`.

Referenced by `agent_xml_compose__create_row_nodes()`, `agent_xml_compose__home()`, `agent_xml_compose__name()`, `agent_xml_compose__owner()`, `agent_xml_compose__wg_code()`, `fipa_envelope_Compose__acl_representation()`, `fipa_envelope_Compose__date()`, `fipa_envelope_Compose__from()`, `fipa_envelope_Compose__intended_receiver()`, `fipa_envelope_Compose__payload_encoding()`, `fipa_envelope_Compose__to()`, `main()`, `mxml_load_data()`, and `scan_file()`.

13.68.1.10 `mxml_node_t*` `mxmlNewTextf` (`mxml_node_t` * *parent*, `int` *whitespace*, `const char` * *format*, ...)

Definition at line 506 of file `mxml-node.c`.

References `mxml_new()`, `mxml_strdupf()`, `MXML_TEXT`, `node`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::value`, and `mxml_text_s::whitespace`.

13.68.1.11 `void` `mxmlRemove` (`mxml_node_t` * *node*)

Definition at line 553 of file `mxml-node.c`.

References `mxml_node_s::child`, `mxml_node_s::last_child`, `mxml_node_s::next`, `mxml_node_s::parent`, and `mxml_node_s::prev`.

Referenced by `mxmlAdd()`, and `mxmlDelete()`.

13.69 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-private.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- void [mxml_error](#) (const char *format,...)
- [mxml_type_t mxml_integer_cb](#) ([mxml_node_t](#) *node)
- [mxml_type_t mxml_opaque_cb](#) ([mxml_node_t](#) *node)
- [mxml_type_t mxml_real_cb](#) ([mxml_node_t](#) *node)

Variables

- void(* [mxml_error_cb](#))(const char *) = NULL

13.69.1 Function Documentation

13.69.1.1 void [mxml_error](#) (const char **format*, ...)

Definition at line 49 of file mxml-private.c.

References [mxml_error_cb](#), and [mxml_strdup\(\)](#).

Referenced by [mxml_add_char\(\)](#), [mxml_fd_getc\(\)](#), [mxml_file_getc\(\)](#), [mxml_get_entity\(\)](#), [mxml_load_data\(\)](#), [mxml_parse_element\(\)](#), [mxml_string_getc\(\)](#), [mxmlElementSetAttr\(\)](#), [mxmlEntityAddCallback\(\)](#), and [mxmlIndexNew\(\)](#).

13.69.1.2 [mxml_type_t mxml_integer_cb](#) ([mxml_node_t](#) **node*)

Definition at line 95 of file mxml-private.c.

References [MXML_INTEGER](#).

13.69.1.3 [mxml_type_t mxml_opaque_cb](#) ([mxml_node_t](#) **node*)

Definition at line 108 of file mxml-private.c.

References [MXML_OPAQUE](#).

13.69.1.4 [mxml_type_t mxml_real_cb](#) ([mxml_node_t](#) **node*)

Definition at line 121 of file mxml-private.c.

References [MXML_REAL](#).

13.69.2 Variable Documentation

13.69.2.1 `void(* mxml_error_cb)(const char *) = NULL`

Referenced by `mxml_error()`, and `mxmlSetErrorCallback()`.

13.70 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-search.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- [mxml_node_t * mxmlFindElement](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [const char * name](#), [const char * attr](#), [const char * value](#), [int descend](#))
- [mxml_node_t * mxmlWalkNext](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [int descend](#))
- [mxml_node_t * mxmlWalkPrev](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [int descend](#))

13.70.1 Function Documentation

13.70.1.1 [mxml_node_t * mxmlFindElement](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [const char * name](#), [const char * attr](#), [const char * value](#), [int descend](#))

Definition at line 51 of file `mxml-search.c`.

References `mxml_value_u::element`, `MXML_DESCEND`, `MXML_ELEMENT`, `mxmlElementGetAttr()`, `mxmlWalkNext()`, `mxml_value_s::name`, `mxml_node_s::next`, `mxml_node_s::type`, and `mxml_node_s::value`.

Referenced by `agent_xml_parse__fill_row_data()`, `agent_xml_parse__task()`, `agent_xml_parse__tasks()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleEnvelope()`, `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleParams()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_HandleReceived()`, `fipa_envelope_HandleTo()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `fipa_envelope_ParseResolvers()`, `main()`, `MC_LoadAgentFromFile()`, `message_xml_parse()`, `mxmlIndexNew()`, `scan_file()`, `sort_node()`, `write_documentation()`, `write_element()`, `xml_find_sibling()`, and `xml_get_child()`.

13.70.1.2 [mxml_node_t * mxmlWalkNext](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [int descend](#))

Definition at line 133 of file `mxml-search.c`.

References `mxml_node_s::child`, `mxml_node_s::next`, and `mxml_node_s::parent`.

Referenced by `mxmlFindElement()`, and `write_element()`.

13.70.1.3 [mxml_node_t * mxmlWalkPrev](#) ([mxml_node_t * node](#), [mxml_node_t * top](#), [int descend](#))

Definition at line 169 of file `mxml-search.c`.

References `mxml_node_s::last_child`, `mxml_node_s::parent`, and `mxml_node_s::prev`.

13.71 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-set.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Functions

- [int mxmlSetCustom](#) ([mxml_node_t](#) **node*, void **data*, void(**destroy*)(void *))
- [int mxmlSetElement](#) ([mxml_node_t](#) **node*, const char **name*)
- [int mxmlSetInteger](#) ([mxml_node_t](#) **node*, [int](#) *integer*)
- [int mxmlSetOpaque](#) ([mxml_node_t](#) **node*, const char **opaque*)
- [int mxmlSetReal](#) ([mxml_node_t](#) **node*, double *real*)
- [int mxmlSetText](#) ([mxml_node_t](#) **node*, [int](#) *whitespace*, const char **string*)
- [int mxmlSetTextf](#) ([mxml_node_t](#) **node*, [int](#) *whitespace*, const char **format*,...)

13.71.1 Function Documentation

13.71.1.1 [int mxmlSetCustom](#) ([mxml_node_t](#) * *node*, void * *data*, void(*) (void *) *destroy*)

Definition at line 46 of file mxml-set.c.

References [mxml_value_u::custom](#), [mxml_custom_s::data](#), [mxml_custom_s::destroy](#), [MXML_CUSTOM](#), [mxml_node_s::type](#), and [mxml_node_s::value](#).

13.71.1.2 [int mxmlSetElement](#) ([mxml_node_t](#) * *node*, const char * *name*)

Definition at line 79 of file mxml-set.c.

References [mxml_value_u::element](#), [MXML_ELEMENT](#), [mxml_value_s::name](#), [mxml_node_s::type](#), and [mxml_node_s::value](#).

13.71.1.3 [int mxmlSetInteger](#) ([mxml_node_t](#) * *node*, [int](#) *integer*)

Definition at line 109 of file mxml-set.c.

References [mxml_value_u::integer](#), [MXML_INTEGER](#), [mxml_node_s::type](#), and [mxml_node_s::value](#).

13.71.1.4 [int mxmlSetOpaque](#) ([mxml_node_t](#) * *node*, const char * *opaque*)

Definition at line 136 of file mxml-set.c.

References [MXML_OPAQUE](#), [mxml_value_u::opaque](#), [mxml_node_s::type](#), and [mxml_node_s::value](#).

13.71.1.5 [int mxmlSetReal](#) ([mxml_node_t](#) * *node*, double *real*)

Definition at line 166 of file mxml-set.c.

References [MXML_REAL](#), [mxml_value_u::real](#), [mxml_node_s::type](#), and [mxml_node_s::value](#).

13.71.1.6 int mxmlSetText (mxml_node_t * *node*, int *whitespace*, const char * *string*)

Definition at line 193 of file mxml-set.c.

References MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

13.71.1.7 int mxmlSetTextf (mxml_node_t * *node*, int *whitespace*, const char * *format*, ...)

Definition at line 225 of file mxml-set.c.

References mxml_strdupf(), MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

13.72 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml-string.c File Reference

```
#include "config.h"
```

Functions

- char * [mxml_strdup](#) (const char *format, va_list ap)
- int [mxml_vsnprintf](#) (char *buffer, size_t bufsz, const char *format, va_list ap)

13.72.1 Function Documentation

13.72.1.1 char* [mxml_strdup](#) (const char **format*, va_list *ap*)

Definition at line 62 of file mxml-string.c.

References [mxml_vsnprintf](#)().

Referenced by [mxml_error](#)(), [mxmlNewTextf](#)(), and [mxmlSetTextf](#)().

13.72.1.2 int [mxml_vsnprintf](#) (char **buffer*, size_t *bufsize*, const char **format*, va_list *ap*)

Definition at line 107 of file mxml-string.c.

References [size](#).

Referenced by [mxml_strdup](#)().

13.73 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxml.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <errno.h>
```

Data Structures

- struct [mxml_attr_s](#)
- struct [mxml_value_s](#)
- struct [mxml_text_s](#)
- struct [mxml_custom_s](#)
- union [mxml_value_u](#)
- struct [mxml_node_s](#)
- struct [mxml_index_s](#)

Defines

- #define [MXML_WRAP](#) 72
- #define [MXML_TAB](#) 8
- #define [MXML_NO_CALLBACK](#) 0
- #define [MXML_INTEGER_CALLBACK](#) [mxml_integer_cb](#)
- #define [MXML_OPAQUE_CALLBACK](#) [mxml_opaque_cb](#)
- #define [MXML_REAL_CALLBACK](#) [mxml_real_cb](#)
- #define [MXML_TEXT_CALLBACK](#) 0
- #define [MXML_NO_PARENT](#) 0
- #define [MXML_DESCEND](#) 1
- #define [MXML_NO_DESCEND](#) 0
- #define [MXML_DESCEND_FIRST](#) -1
- #define [MXML_WS_BEFORE_OPEN](#) 0
- #define [MXML_WS_AFTER_OPEN](#) 1
- #define [MXML_WS_BEFORE_CLOSE](#) 2
- #define [MXML_WS_AFTER_CLOSE](#) 3
- #define [MXML_ADD_BEFORE](#) 0
- #define [MXML_ADD_AFTER](#) 1
- #define [MXML_ADD_TO_PARENT](#) NULL

Typedefs

- typedef enum [mxml_type_e](#) [mxml_type_t](#)
- typedef struct [mxml_attr_s](#) [mxml_attr_t](#)
- typedef struct [mxml_value_s](#) [mxml_element_t](#)
- typedef struct [mxml_text_s](#) [mxml_text_t](#)
- typedef struct [mxml_custom_s](#) [mxml_custom_t](#)

- typedef union `mxml_value_u` `mxml_value_t`
- typedef struct `mxml_node_s` `mxml_node_t`
- typedef struct `mxml_index_s` `mxml_index_t`
- typedef `int`(* `mxml_custom_load_cb_t`)(`mxml_node_t` *, const char *)
- typedef char *(* `mxml_custom_save_cb_t`)(`mxml_node_t` *)

Enumerations

- enum `mxml_type_e` {
`MXML_ELEMENT`, `MXML_INTEGER`, `MXML_OPAQUE`, `MXML_REAL`,
`MXML_TEXT`, `MXML_CUSTOM` }

Functions

- void `mxmlAdd` (`mxml_node_t` *parent, int where, `mxml_node_t` *child, `mxml_node_t` *node)
- void `mxmlDelete` (`mxml_node_t` *node)
- const char * `mxmlElementGetAttr` (`mxml_node_t` *node, const char *name)
- void `mxmlElementSetAttr` (`mxml_node_t` *node, const char *name, const char *value)
- int `mxmlEntityAddCallback` (int(*cb)(const char *name))
- const char * `mxmlEntityGetName` (int val)
- int `mxmlEntityGetValue` (const char *name)
- void `mxmlEntityRemoveCallback` (int(*cb)(const char *name))
- `mxml_node_t` * `mxmlFindElement` (`mxml_node_t` *node, `mxml_node_t` *top, const char *name, const char *attr, const char *value, int descend)
- void `mxmlIndexDelete` (`mxml_index_t` *ind)
- `mxml_node_t` * `mxmlIndexEnum` (`mxml_index_t` *ind)
- `mxml_node_t` * `mxmlIndexFind` (`mxml_index_t` *ind, const char *element, const char *value)
- `mxml_index_t` * `mxmlIndexNew` (`mxml_node_t` *node, const char *element, const char *attr)
- `mxml_node_t` * `mxmlIndexReset` (`mxml_index_t` *ind)
- `mxml_node_t` * `mxmlLoadFd` (`mxml_node_t` *top, int fd, `mxml_type_t`(*cb)(`mxml_node_t` *))
- `mxml_node_t` * `mxmlLoadFile` (`mxml_node_t` *top, FILE *fp, `mxml_type_t`(*cb)(`mxml_node_t` *))
- `mxml_node_t` * `mxmlLoadString` (`mxml_node_t` *top, const char *s, `mxml_type_t`(*cb)(`mxml_node_t` *))
- `mxml_node_t` * `mxmlNewCustom` (`mxml_node_t` *parent, void *data, void(*destroy)(void *))
- `mxml_node_t` * `mxmlNewElement` (`mxml_node_t` *parent, const char *name)
- `mxml_node_t` * `mxmlNewInteger` (`mxml_node_t` *parent, int integer)
- `mxml_node_t` * `mxmlNewOpaque` (`mxml_node_t` *parent, const char *opaque)
- `mxml_node_t` * `mxmlNewReal` (`mxml_node_t` *parent, double real)
- `mxml_node_t` * `mxmlNewText` (`mxml_node_t` *parent, int whitespace, const char *string)
- `mxml_node_t` * `mxmlNewTextf` (`mxml_node_t` *parent, int whitespace, const char *format,...)
- void `mxmlRemove` (`mxml_node_t` *node)
- char * `mxmlSaveAllocString` (`mxml_node_t` *node, const char *(*cb)(`mxml_node_t` *, int))
- int `mxmlSaveFd` (`mxml_node_t` *node, int fd, const char *(*cb)(`mxml_node_t` *, int))
- int `mxmlSaveFile` (`mxml_node_t` *node, FILE *fp, const char *(*cb)(`mxml_node_t` *, int))
- int `mxmlSaveString` (`mxml_node_t` *node, char *buffer, int bufsize, const char *(*cb)(`mxml_node_t` *, int))
- int `mxmlSetCustom` (`mxml_node_t` *node, void *data, void(*destroy)(void *))
- void `mxmlSetCustomHandlers` (`mxml_custom_load_cb_t` load, `mxml_custom_save_cb_t` save)
- int `mxmlSetElement` (`mxml_node_t` *node, const char *name)

- void `mxmlSetErrorCallback` (void(*cb)(const char *))
- int `mxmlSetInteger` (mxml_node_t *node, int integer)
- int `mxmlSetOpaque` (mxml_node_t *node, const char *opaque)
- int `mxmlSetReal` (mxml_node_t *node, double real)
- int `mxmlSetText` (mxml_node_t *node, int whitespace, const char *string)
- int `mxmlSetTextf` (mxml_node_t *node, int whitespace, const char *format,...)
- mxml_node_t * `mxmlWalkNext` (mxml_node_t *node, mxml_node_t *top, int descend)
- mxml_node_t * `mxmlWalkPrev` (mxml_node_t *node, mxml_node_t *top, int descend)
- void `mxml_error` (const char *format,...)
- mxml_type_t `mxml_integer_cb` (mxml_node_t *node)
- mxml_type_t `mxml_opaque_cb` (mxml_node_t *node)
- mxml_type_t `mxml_real_cb` (mxml_node_t *node)

13.73.1 Define Documentation

13.73.1.1 #define MXML_ADD_AFTER 1

Definition at line 68 of file mxml.h.

Referenced by `add_variable()`, `agent_xml_compose()`, `agent_xml_compose__agent_data()`, `agent_xml_compose__create_row_nodes()`, `agent_xml_compose__data()`, `agent_xml_compose__gaf_message()`, `agent_xml_compose__message()`, `agent_xml_compose__mobile_agent()`, `agent_xml_compose__task()`, `agent_xml_compose__tasks()`, `fipa_envelope_Compose()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__params()`, `mxml_new()`, `mxmlAdd()`, `scan_file()`, and `sort_node()`.

13.73.1.2 #define MXML_ADD_BEFORE 0

Definition at line 67 of file mxml.h.

Referenced by `mxmlAdd()`, `scan_file()`, and `sort_node()`.

13.73.1.3 #define MXML_ADD_TO_PARENT NULL

Definition at line 69 of file mxml.h.

Referenced by `add_variable()`, `agent_xml_compose()`, `agent_xml_compose__create_row_nodes()`, `fipa_envelope_Compose()`, `fipa_envelope_Compose__envelope()`, `fipa_envelope_Compose__params()`, `mxml_new()`, `scan_file()`, and `sort_node()`.

13.73.1.4 #define MXML_DESCEND 1

Definition at line 58 of file mxml.h.

Referenced by `agent_xml_parse__tasks()`, `main()`, `MC_LoadAgentFromFile()`, `message_xml_parse()`, `mxmlFindElement()`, `mxmlIndexNew()`, and `write_element()`.

13.73.1.5 #define MXML_DESCEND_FIRST -1

Definition at line 60 of file mxml.h.

Referenced by `agent_xml_parse__fill_row_data()`, `agent_xml_parse__task()`, `agent_xml_parse__tasks()`, `fipa_envelope_HandleAclRepresentation()`, `fipa_envelope_HandleComments()`, `fipa_envelope_HandleDate()`, `fipa_envelope_HandleEnvelope()`, `fipa_envelope_HandleFrom()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleParams()`, `fipa_envelope_HandlePayloadEncoding()`, `fipa_envelope_HandlePayloadLength()`, `fipa_envelope_HandleReceived()`, `fipa_envelope_HandleTo()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseAgentIdentifier()`, `fipa_envelope_ParseResolvers()`, `scan_file()`, `sort_node()`, and `write_documentation()`.

13.73.1.6 **#define MXML_INTEGER_CALLBACK mxml_integer_cb**

Definition at line 48 of file `mxml.h`.

Referenced by `main()`.

13.73.1.7 **#define MXML_NO_CALLBACK 0**

Definition at line 47 of file `mxml.h`.

Referenced by `acc_connection_Thread()`, `agent_xml_compose()`, `fipa_envelope_Compose()`, `fipa_envelope_Parse()`, `main()`, `message_InitializeFromAgent()`, `message_InitializeFromConnection()`, and `message_InitializeFromString()`.

13.73.1.8 **#define MXML_NO_DESCEND 0**

Definition at line 59 of file `mxml.h`.

Referenced by `agent_xml_parse__task()`, `agent_xml_parse__tasks()`, `fipa_envelope_HandleIntendedReceiver()`, `fipa_envelope_HandleTo()`, `fipa_envelope_ParseAddresses()`, `fipa_envelope_ParseResolvers()`, `main()`, `message_xml_parse()`, `write_documentation()`, `write_element()`, `xml_find_sibling()`, and `xml_get_deep_child()`.

13.73.1.9 **#define MXML_NO_PARENT 0**

Definition at line 56 of file `mxml.h`.

Referenced by `agent_xml_compose__agent_code()`, `agent_xml_compose__create_row_nodes()`, `main()`, and `scan_file()`.

13.73.1.10 **#define MXML_OPAQUE_CALLBACK mxml_opaque_cb**

Definition at line 50 of file `mxml.h`.

Referenced by `main()`.

13.73.1.11 **#define MXML_REAL_CALLBACK mxml_real_cb**

Definition at line 52 of file `mxml.h`.

Referenced by `main()`.

13.73.1.12 #define MXML_TAB 8

Definition at line 45 of file mxml.h.

Referenced by mxml_write_ws().

13.73.1.13 #define MXML_TEXT_CALLBACK 0

Definition at line 54 of file mxml.h.

13.73.1.14 #define MXML_WRAP 72

Definition at line 44 of file mxml.h.

Referenced by mxml_write_node().

13.73.1.15 #define MXML_WS_AFTER_CLOSE 3

Definition at line 65 of file mxml.h.

Referenced by mxml_write_node(), whitespace_cb(), and ws_cb().

13.73.1.16 #define MXML_WS_AFTER_OPEN 1

Definition at line 63 of file mxml.h.

Referenced by mxml_write_node(), whitespace_cb(), and ws_cb().

13.73.1.17 #define MXML_WS_BEFORE_CLOSE 2

Definition at line 64 of file mxml.h.

Referenced by mxml_write_node(), whitespace_cb(), and ws_cb().

13.73.1.18 #define MXML_WS_BEFORE_OPEN 0

Definition at line 62 of file mxml.h.

Referenced by mxml_write_node(), whitespace_cb(), and ws_cb().

13.73.2 Typedef Documentation**13.73.2.1 typedef struct mxml_attr_s mxml_attr_t****13.73.2.2 typedef int(* mxml_custom_load_cb_t)(mxml_node_t *, const char *)**

Definition at line 142 of file mxml.h.

13.73.2.3 typedef char*(* mxml_custom_save_cb_t)(mxml_node_t *)

Definition at line 145 of file mxml.h.

13.73.2.4 `typedef struct mxml_custom_s mxml_custom_t`

13.73.2.5 `typedef struct mxml_value_s mxml_element_t`

13.73.2.6 `typedef struct mxml_index_s mxml_index_t`

13.73.2.7 `typedef struct mxml_node_s mxml_node_t`

13.73.2.8 `typedef struct mxml_text_s mxml_text_t`

13.73.2.9 `typedef enum mxml_type_e mxml_type_t`

13.73.2.10 `typedef union mxml_value_u mxml_value_t`

13.73.3 Enumeration Type Documentation

13.73.3.1 `enum mxml_type_e`

Enumerator:

MXML_ELEMENT

MXML_INTEGER

MXML_OPAQUE

MXML_REAL

MXML_TEXT

MXML_CUSTOM

Definition at line 76 of file mxml.h.

13.73.4 Function Documentation

13.73.4.1 `void mxml_error (const char *format, ...)`

Definition at line 49 of file mxml-private.c.

References `mxml_error_cb`, and `mxml_strdup()`.

Referenced by `mxml_add_char()`, `mxml_fd_getc()`, `mxml_file_getc()`, `mxml_get_entity()`, `mxml_load_data()`, `mxml_parse_element()`, `mxml_string_getc()`, `mxmlElementSetAttr()`, `mxmlEntityAddCallback()`, and `mxmlIndexNew()`.

13.73.4.2 `mxml_type_t mxml_integer_cb (mxml_node_t *node)`

Definition at line 95 of file mxml-private.c.

References `MXML_INTEGER`.

13.73.4.3 `mxml_type_t mxml_opaque_cb (mxml_node_t *node)`

Definition at line 108 of file mxml-private.c.

References `MXML_OPAQUE`.

13.73.4.4 mxml_type_t mxml_real_cb (mxml_node_t * node)

Definition at line 121 of file mxml-private.c.

References MXML_REAL.

13.73.4.5 void mxmlAdd (mxml_node_t * parent, int where, mxml_node_t * child, mxml_node_t * node)

Definition at line 62 of file mxml-node.c.

References mxml_node_s::child, mxml_node_s::last_child, MXML_ADD_AFTER, MXML_ADD_BEFORE, mxmlRemove(), mxml_node_s::next, mxml_node_s::parent, and mxml_node_s::prev.

Referenced by add_variable(), agent_xml_compose(), agent_xml_compose__agent_data(), agent_xml_compose__create_row_nodes(), agent_xml_compose__data(), agent_xml_compose__gaf_message(), agent_xml_compose__message(), agent_xml_compose__mobile_agent(), agent_xml_compose__task(), agent_xml_compose__tasks(), fipa_envelope_Compose(), fipa_envelope_Compose__envelope(), fipa_envelope_Compose__params(), mxml_new(), scan_file(), and sort_node().

13.73.4.6 void mxmlDelete (mxml_node_t * node)

Definition at line 196 of file mxml-node.c.

References mxml_value_s::attrs, mxml_node_s::child, mxml_value_u::custom, mxml_custom_s::data, mxml_custom_s::destroy, mxml_value_u::element, MXML_CUSTOM, MXML_ELEMENT, MXML_INTEGER, MXML_OPAQUE, MXML_REAL, MXML_TEXT, mxmlDelete(), mxmlRemove(), mxml_attr_s::name, mxml_value_s::name, mxml_value_s::num_attrs, mxml_value_u::opaque, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_attr_s::value, and mxml_node_s::value.

Referenced by add_variable(), agent_datastate_Destroy(), fipa_envelope_Compose(), fipa_envelope_Parse(), main(), message_Destroy(), mxml_load_data(), mxmlDelete(), scan_file(), and sort_node().

13.73.4.7 const char* mxmlElementGetAttr (mxml_node_t * node, const char * name)

Definition at line 43 of file mxml-attr.c.

References mxml_value_s::attrs, mxml_value_u::element, MXML_ELEMENT, mxml_attr_s::name, mxml_value_s::num_attrs, mxml_node_s::type, mxml_attr_s::value, and mxml_node_s::value.

Referenced by agent_xml_parse__agent_code(), agent_xml_parse__data(), agent_xml_parse__task(), agent_xml_parse__tasks(), fipa_envelope_HandleReceived(), index_compare(), index_find(), message_xml_parse__message(), mxml_parse_element(), mxmlFindElement(), mxmlIndexNew(), scan_file(), sort_node(), type_cb(), and write_documentation().

13.73.4.8 void mxmlElementSetAttr (mxml_node_t * node, const char * name, const char * value)

Definition at line 90 of file mxml-attr.c.

References mxml_value_s::attrs, mxml_value_u::element, MXML_ELEMENT, mxml_error(), mxml_value_s::name, mxml_attr_s::name, mxml_value_s::num_attrs, mxml_node_s::type, mxml_attr_s::value, and mxml_node_s::value.

Referenced by add_variable(), agent_xml_compose__agent_code(), agent_xml_compose__create_row_nodes(), agent_xml_compose__data(), agent_xml_compose__message(), agent_xml_compose__task(),

agent_xml_compose__tasks(), fipa_envelope_Compose__params(), main(), mxml_parse_element(), scan_file(), sort_node(), and update_comment().

13.73.4.9 int mxmlEntityAddCallback (int*)(const char *name) cb)

Definition at line 65 of file mxml-entity.c.

References callbacks, mxml_error(), and num_callbacks.

13.73.4.10 const char* mxmlEntityGetName (int val)

Definition at line 91 of file mxml-entity.c.

Referenced by mxml_write_name(), and mxml_write_string().

13.73.4.11 int mxmlEntityGetValue (const char * name)

Definition at line 121 of file mxml-entity.c.

References callbacks, and num_callbacks.

Referenced by mxml_get_entity().

13.73.4.12 void mxmlEntityRemoveCallback (int*)(const char *name) cb)

Definition at line 140 of file mxml-entity.c.

References callbacks, and num_callbacks.

13.73.4.13 mxml_node_t* mxmlFindElement (mxml_node_t * node, mxml_node_t * top, const char * name, const char * attr, const char * value, int descend)

Definition at line 51 of file mxml-search.c.

References mxml_value_u::element, MXML_DESCEND, MXML_ELEMENT, mxmlElementGetAttr(), mxmlWalkNext(), mxml_value_s::name, mxml_node_s::next, mxml_node_s::type, and mxml_node_s::value.

Referenced by agent_xml_parse__fill_row_data(), agent_xml_parse__task(), agent_xml_parse__tasks(), fipa_envelope_HandleAclRepresentation(), fipa_envelope_HandleComments(), fipa_envelope_HandleDate(), fipa_envelope_HandleEnvelope(), fipa_envelope_HandleFrom(), fipa_envelope_HandleIntendedReceiver(), fipa_envelope_HandleParams(), fipa_envelope_HandlePayloadEncoding(), fipa_envelope_HandlePayloadLength(), fipa_envelope_HandleReceived(), fipa_envelope_HandleTo(), fipa_envelope_ParseAddresses(), fipa_envelope_ParseAgentIdentifier(), fipa_envelope_ParseResolvers(), main(), MC_LoadAgentFromFile(), message_xml_parse(), mxmlIndexNew(), scan_file(), sort_node(), write_documentation(), write_element(), xml_find_sibling(), and xml_get_child().

13.73.4.14 void mxmlIndexDelete (mxml_index_t * ind)

Definition at line 58 of file mxml-index.c.

References mxml_index_s::alloc_nodes, mxml_index_s::attr, and mxml_index_s::nodes.

Referenced by main(), and mxmlIndexNew().

13.73.4.15 mxml_node_t* mxmlIndexEnum (mxml_index_t * *ind*)

Definition at line 88 of file mxml-index.c.

References mxml_index_s::cur_node, mxml_index_s::nodes, and mxml_index_s::num_nodes.

Referenced by main(), and mxmlIndexFind().

13.73.4.16 mxml_node_t* mxmlIndexFind (mxml_index_t * *ind*, const char * *element*, const char * *value*)

Definition at line 118 of file mxml-index.c.

References mxml_index_s::attr, mxml_index_s::cur_node, index_find(), mxmlIndexEnum(), mxml_index_s::nodes, and mxml_index_s::num_nodes.

Referenced by main().

13.73.4.17 mxml_index_t* mxmlIndexNew (mxml_node_t * *node*, const char * *element*, const char * *attr*)

Definition at line 301 of file mxml-index.c.

References mxml_index_s::alloc_nodes, mxml_index_s::attr, mxml_value_u::element, index_sort(), MXML_DESCEND, mxml_error(), mxmlElementGetAttr(), mxmlFindElement(), mxmlIndexDelete(), mxml_value_s::name, mxml_index_s::nodes, mxml_index_s::num_nodes, and mxml_node_s::value.

Referenced by main().

13.73.4.18 mxml_node_t* mxmlIndexReset (mxml_index_t * *ind*)

Definition at line 459 of file mxml-index.c.

References mxml_index_s::cur_node, mxml_index_s::nodes, and mxml_index_s::num_nodes.

Referenced by main().

13.73.4.19 mxml_node_t* mxmlLoadFd (mxml_node_t * *top*, int *fd*, mxml_type_t(*) (mxml_node_t *) *cb*)**13.73.4.20 mxml_node_t* mxmlLoadFile (mxml_node_t * *top*, FILE * *fp*, mxml_type_t(*) (mxml_node_t *) *cb*)****13.73.4.21 mxml_node_t* mxmlLoadString (mxml_node_t * *top*, const char * *s*, mxml_type_t(*) (mxml_node_t *) *cb*)****13.73.4.22 mxml_node_t* mxmlNewCustom (mxml_node_t * *parent*, void * *data*, void(*) (void *) *destroy*)**

Definition at line 287 of file mxml-node.c.

References mxml_value_u::custom, mxml_custom_s::data, mxml_custom_s::destroy, MXML_CUSTOM, mxml_new(), node, and mxml_node_s::value.

Referenced by mxml_load_data().

13.73.4.23 mxmxml_node_t* mxmxmlNewElement (mxml_node_t * *parent*, const char * *name*)

Definition at line 323 of file mxml-node.c.

References mxml_value_u::element, MXML_ELEMENT, mxml_new(), mxml_value_s::name, node, and mxml_node_s::value.

Referenced by add_variable(), agent_xml_compose__agent_code(), agent_xml_compose__agent_data(), agent_xml_compose__create_row_nodes(), agent_xml_compose__data(), agent_xml_compose__gaf_message(), agent_xml_compose__home(), agent_xml_compose__message(), agent_xml_compose__mobile_agent(), agent_xml_compose__name(), agent_xml_compose__owner(), agent_xml_compose__task(), agent_xml_compose__tasks(), agent_xml_compose__wg_code(), fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__envelope(), fipa_envelope_Compose__from(), fipa_envelope_Compose__intended_receiver(), fipa_envelope_Compose__params(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), main(), mxml_load_data(), scan_file(), and xml_new_cdata().

13.73.4.24 mxmxml_node_t* mxmxmlNewInteger (mxml_node_t * *parent*, int *integer*)

Definition at line 361 of file mxml-node.c.

References mxml_value_u::integer, MXML_INTEGER, mxml_new(), node, and mxml_node_s::value.

Referenced by main(), and mxml_load_data().

13.73.4.25 mxmxml_node_t* mxmxmlNewOpaque (mxml_node_t * *parent*, const char * *opaque*)

Definition at line 392 of file mxml-node.c.

References mxml_new(), MXML_OPAQUE, node, mxml_value_u::opaque, and mxml_node_s::value.

Referenced by main(), and mxml_load_data().

13.73.4.26 mxmxml_node_t* mxmxmlNewReal (mxml_node_t * *parent*, double *real*)

Definition at line 430 of file mxml-node.c.

References mxml_new(), MXML_REAL, node, mxml_value_u::real, and mxml_node_s::value.

Referenced by main(), and mxml_load_data().

13.73.4.27 mxmxml_node_t* mxmxmlNewText (mxml_node_t * *parent*, int *whitespace*, const char * *string*)

Definition at line 462 of file mxml-node.c.

References mxml_new(), MXML_TEXT, node, mxml_text_s::string, mxml_value_u::text, mxml_node_s::value, and mxml_text_s::whitespace.

Referenced by agent_xml_compose__create_row_nodes(), agent_xml_compose__home(), agent_xml_compose__name(), agent_xml_compose__owner(), agent_xml_compose__wg_code(), fipa_envelope_Compose__acl_representation(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), fipa_envelope_Compose__intended_receiver(), fipa_envelope_Compose__payload_encoding(), fipa_envelope_Compose__to(), main(), mxml_load_data(), and scan_file().

13.73.4.28 `mxml_node_t* mxmlNewTextf (mxml_node_t * parent, int whitespace, const char * format, ...)`

Definition at line 506 of file mxml-node.c.

References `mxml_new()`, `mxml_strdupf()`, `MXML_TEXT`, `node`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::value`, and `mxml_text_s::whitespace`.

13.73.4.29 `void mxmlRemove (mxml_node_t * node)`

Definition at line 553 of file mxml-node.c.

References `mxml_node_s::child`, `mxml_node_s::last_child`, `mxml_node_s::next`, `mxml_node_s::parent`, and `mxml_node_s::prev`.

Referenced by `mxmlAdd()`, and `mxmlDelete()`.

13.73.4.30 `char* mxmlSaveAllocString (mxml_node_t * node, const char *(*)(mxml_node_t *, int) cb)`**13.73.4.31** `int mxmlSaveFd (mxml_node_t * node, int fd, const char *(*)(mxml_node_t *, int) cb)`**13.73.4.32** `int mxmlSaveFile (mxml_node_t * node, FILE * fp, const char *(*)(mxml_node_t *, int) cb)`**13.73.4.33** `int mxmlSaveString (mxml_node_t * node, char * buffer, int bufsize, const char *(*)(mxml_node_t *, int) cb)`**13.73.4.34** `int mxmlSetCustom (mxml_node_t * node, void * data, void (*)(void *) destroy)`

Definition at line 46 of file mxml-set.c.

References `mxml_value_u::custom`, `mxml_custom_s::data`, `mxml_custom_s::destroy`, `MXML_CUSTOM`, `mxml_node_s::type`, and `mxml_node_s::value`.

13.73.4.35 `void mxmlSetCustomHandlers (mxml_custom_load_cb_t load, mxml_custom_save_cb_t save)`

Definition at line 456 of file mxml-file.c.

References `mxml_custom_load_cb`, and `mxml_custom_save_cb`.

13.73.4.36 `int mxmlSetElement (mxml_node_t * node, const char * name)`

Definition at line 79 of file mxml-set.c.

References `mxml_value_u::element`, `MXML_ELEMENT`, `mxml_value_s::name`, `mxml_node_s::type`, and `mxml_node_s::value`.

13.73.4.37 `void mxmlSetErrorCallback (void (*)(const char *) cb)`

Definition at line 471 of file mxml-file.c.

References `mxml_error_cb`.

13.73.4.38 int mxmlSetInteger (mxml_node_t * *node*, int *integer*)

Definition at line 109 of file mxml-set.c.

References mxml_value_u::integer, MXML_INTEGER, mxml_node_s::type, and mxml_node_s::value.

13.73.4.39 int mxmlSetOpaque (mxml_node_t * *node*, const char * *opaque*)

Definition at line 136 of file mxml-set.c.

References MXML_OPAQUE, mxml_value_u::opaque, mxml_node_s::type, and mxml_node_s::value.

13.73.4.40 int mxmlSetReal (mxml_node_t * *node*, double *real*)

Definition at line 166 of file mxml-set.c.

References MXML_REAL, mxml_value_u::real, mxml_node_s::type, and mxml_node_s::value.

13.73.4.41 int mxmlSetText (mxml_node_t * *node*, int *whitespace*, const char * *string*)

Definition at line 193 of file mxml-set.c.

References MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

13.73.4.42 int mxmlSetTextf (mxml_node_t * *node*, int *whitespace*, const char * *format*, ...)

Definition at line 225 of file mxml-set.c.

References mxml_strdupf(), MXML_TEXT, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and mxml_text_s::whitespace.

13.73.4.43 mxml_node_t* mxmlWalkNext (mxml_node_t * *node*, mxml_node_t * *top*, int *descend*)

Definition at line 133 of file mxml-search.c.

References mxml_node_s::child, mxml_node_s::next, and mxml_node_s::parent.

Referenced by mxmlFindElement(), and write_element().

13.73.4.44 mxml_node_t* mxmlWalkPrev (mxml_node_t * *node*, mxml_node_t * *top*, int *descend*)

Definition at line 169 of file mxml-search.c.

References mxml_node_s::last_child, mxml_node_s::parent, and mxml_node_s::prev.

13.74 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/mxmldoc.c File Reference

```
#include "config.h"
#include "mxml.h"
```

Defines

- `#define STATE_NONE` 0
- `#define STATE_PREPROCESSOR` 1
- `#define STATE_C_COMMENT` 2
- `#define STATE_CXX_COMMENT` 3
- `#define STATE_STRING` 4
- `#define STATE_CHARACTER` 5
- `#define STATE_IDENTIFIER` 6

Functions

- static `mxml_node_t` * `add_variable` (`mxml_node_t` *parent, const char *name, `mxml_node_t` *type)
- static void `safe_strcpy` (char *dst, const char *src)
- static int `scan_file` (const char *filename, FILE *fp, `mxml_node_t` *doc)
- static void `sort_node` (`mxml_node_t` *tree, `mxml_node_t` *func)
- static void `update_comment` (`mxml_node_t` *parent, `mxml_node_t` *comment)
- static void `write_documentation` (`mxml_node_t` *doc)
- static void `write_element` (`mxml_node_t` *doc, `mxml_node_t` *element)
- static void `write_string` (const char *s)
- static const char * `ws_cb` (`mxml_node_t` *node, int where)
- int `main` (int argc, char *argv[])

13.74.1 Define Documentation

13.74.1.1 `#define STATE_C_COMMENT` 2

Definition at line 125 of file `mxmldoc.c`.

Referenced by `scan_file()`.

13.74.1.2 `#define STATE_CHARACTER` 5

Definition at line 128 of file `mxmldoc.c`.

Referenced by `scan_file()`.

13.74.1.3 `#define STATE_CXX_COMMENT` 3

Definition at line 126 of file `mxmldoc.c`.

Referenced by `scan_file()`.

13.74.1.4 **#define STATE_IDENTIFIER 6**

Definition at line 129 of file mxmldoc.c.

Referenced by scan_file().

13.74.1.5 **#define STATE_NONE 0**

Definition at line 123 of file mxmldoc.c.

Referenced by scan_file().

13.74.1.6 **#define STATE_PREPROCESSOR 1**

Definition at line 124 of file mxmldoc.c.

Referenced by scan_file().

13.74.1.7 **#define STATE_STRING 4**

Definition at line 127 of file mxmldoc.c.

Referenced by scan_file().

13.74.2 **Function Documentation**

13.74.2.1 **static mxml_node_t * add_variable (mxml_node_t * *parent*, const char * *name*, mxml_node_t * *type*) [static]**

Definition at line 309 of file mxmldoc.c.

References mxml_node_s::child, mxml_node_s::last_child, MXML_ADD_AFTER, MXML_ADD_TO_PARENT, mxmlAdd(), mxmlDelete(), mxmlElementSetAttr(), mxmlNewElement(), mxml_node_s::next, node, mxml_text_s::string, mxml_value_u::text, mxml_node_s::value, and mxml_text_s::whitespace.

Referenced by scan_file().

13.74.2.2 **int main (int *argc*, char * *argv*[])**

Definition at line 155 of file mxmldoc.c.

References MXML_DESCEND, MXML_NO_CALLBACK, mxmlDelete(), mxmlElementSetAttr(), mxmlFindElement(), mxmlLoadFile(), mxmlNewElement(), mxmlSaveFile(), scan_file(), write_documentation(), and ws_cb().

13.74.2.3 **static void safe_strcpy (char * *dst*, const char * *src*) [static]**

Definition at line 423 of file mxmldoc.c.

Referenced by update_comment().

13.74.2.4 static int scan_file (const char *filename, FILE *fp, mxml_node_t *doc) [static]

Definition at line 438 of file mxmldoc.c.

References add_variable(), mxml_node_s::child, mxml_value_u::element, mxml_node_s::last_child, MXML_ADD_AFTER, MXML_ADD_BEFORE, MXML_ADD_TO_PARENT, MXML_DESCEND_FIRST, MXML_NO_PARENT, mxmlAdd(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxmlNewElement(), mxmlNewText(), mxml_value_s::name, mxml_node_s::next, node, sort_node(), STATE_C_COMMENT, STATE_CHARACTER, STATE_CXX_COMMENT, STATE_IDENTIFIER, STATE_NONE, STATE_PREPROCESSOR, STATE_STRING, mxml_text_s::string, mxml_value_u::text, update_comment(), mxml_node_s::value, and mxml_text_s::whitespace.

Referenced by main().

13.74.2.5 static void sort_node (mxml_node_t *tree, mxml_node_t *func) [static]

Definition at line 1613 of file mxmldoc.c.

References mxml_node_s::child, mxml_value_u::element, MXML_ADD_AFTER, MXML_ADD_BEFORE, MXML_ADD_TO_PARENT, MXML_DESCEND_FIRST, mxmlAdd(), mxmlDelete(), mxmlElementGetAttr(), mxmlElementSetAttr(), mxmlFindElement(), mxml_value_s::name, mxml_node_s::next, mxml_node_s::parent, and mxml_node_s::value.

Referenced by scan_file().

13.74.2.6 static void update_comment (mxml_node_t *parent, mxml_node_t *comment) [static]

Definition at line 1701 of file mxmldoc.c.

References mxml_value_u::element, mxmlElementSetAttr(), mxml_value_s::name, safe_strcpy(), mxml_text_s::string, mxml_value_u::text, and mxml_node_s::value.

Referenced by scan_file().

13.74.2.7 static void write_documentation (mxml_node_t *doc) [static]

Definition at line 1801 of file mxmldoc.c.

References mxml_node_s::child, MXML_DESCEND_FIRST, MXML_NO_DESCEND, MXML_VERSION, mxmlElementGetAttr(), mxmlFindElement(), and write_element().

Referenced by main().

13.74.2.8 static void write_element (mxml_node_t *doc, mxml_node_t *element) [static]

Definition at line 2611 of file mxmldoc.c.

References mxml_node_s::child, MXML_DESCEND, MXML_NO_DESCEND, MXML_TEXT, mxmlFindElement(), mxmlWalkNext(), node, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, mxml_text_s::whitespace, and write_string().

Referenced by write_documentation().

13.74.2.9 static void write_string (const char * *s*) [static]

Definition at line 2656 of file mxmldoc.c.

Referenced by write_element().

13.74.2.10 static const char * ws_cb (mxml_node_t * *node*, int *where*) [static]

Definition at line 2714 of file mxmldoc.c.

References mxml_value_u::element, MXML_WS_AFTER_CLOSE, MXML_WS_AFTER_OPEN, MXML_WS_BEFORE_CLOSE, MXML_WS_BEFORE_OPEN, mxml_value_s::name, mxml_node_s::parent, and mxml_node_s::value.

Referenced by main().

13.75 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/class.cxx File Reference

Data Structures

- class `foo_c`

Variables

- `foo_c f`
- `foo_c foo`
- `bar = b`

13.75.1 Variable Documentation

13.75.1.1 `bar = b`

Definition at line 77 of file class.cxx.

13.75.1.2 `foo_c f`

Referenced by `foo_float_function()`, `main()`, `MC_Initialize()`, `md5_check()`, `md5_file()`, `read_encrypted_file()`, `sha1_check()`, `sha1_file()`, `sha2_check()`, `sha2_file()`, `sha4_file()`, `x509parse_certfile()`, and `x509parse_keyfile()`.

13.75.1.3 `foo_c foo`

13.76 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/enum.cxx File Reference

Typedefs

- typedef enum [foo_enum_e](#) [foo_enum_t](#)

Enumerations

- enum [foo_enum_e](#) { [FOO_ONE](#), [FOO_TWO](#), [FOO_RED](#), [FOO_BLUE](#) }

13.76.1 Typedef Documentation

13.76.1.1 typedef enum [foo_enum_e](#) [foo_enum_t](#)

13.76.2 Enumeration Type Documentation

13.76.2.1 enum [foo_enum_e](#)

Enumerator:

FOO_ONE
FOO_TWO
FOO_RED
FOO_BLUE

Definition at line 1 of file enum.cxx.

13.77 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/function.cxx File Reference

Functions

- void [foo_void_function](#) ([int](#) one, float *two, const char *three)
- float [foo_float_function](#) ([int](#) one, const char *two)
- [int](#) [foo_default_string](#) ([int](#) one, const char *two="2")
- [int](#) [foo_default_int](#) ([int](#) one, [int](#) two=2)

13.77.1 Function Documentation

13.77.1.1 [int](#) [foo_default_int](#) ([int](#) *one*, [int](#) *two* = 2)

Definition at line 65 of file function.cxx.

13.77.1.2 [int](#) [foo_default_string](#) ([int](#) *one*, const char * *two* = "2")

Definition at line 45 of file function.cxx.

13.77.1.3 [float](#) [foo_float_function](#) ([int](#) *one*, const char * *two*)

Definition at line 26 of file function.cxx.

References [f](#).

13.77.1.4 [void](#) [foo_void_function](#) ([int](#) *one*, float * *two*, const char * *three*)

Definition at line 6 of file function.cxx.

13.78 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/test/struct.cxx File Reference

Data Structures

- struct [foo_s](#)

Typedefs

- typedef struct [foo_s](#) [foo_t](#)

13.78.1 Typedef Documentation

13.78.1.1 typedef struct foo_s foo_t

13.79 /home/dko/Projects/mobilec/trunk/src/mxml-2.2.2/testmxml.c File Reference

```
#include "config.h"
#include "mxml.h"
#include <unistd.h>
#include <fcntl.h>
```

Defines

- `#define O_BINARY 0`

Functions

- `mxml_type_t type_cb (mxml_node_t *node)`
- `const char * whitespace_cb (mxml_node_t *node, int where)`
- `int main (int argc, char *argv[])`

13.79.1 Define Documentation

13.79.1.1 `#define O_BINARY 0`

Definition at line 42 of file testmxml.c.

Referenced by `main()`.

13.79.2 Function Documentation

13.79.2.1 `int main (int argc, char *argv[])`

Definition at line 59 of file testmxml.c.

References `mxml_node_s::child`, `mxml_value_u::element`, `f`, `mxml_value_u::integer`, `mxml_node_s::last_child`, `MXML_DESCEND`, `MXML_ELEMENT`, `MXML_INTEGER`, `MXML_INTEGER_CALLBACK`, `MXML_NO_CALLBACK`, `MXML_NO_DESCEND`, `MXML_NO_PARENT`, `MXML_OPAQUE`, `MXML_OPAQUE_CALLBACK`, `MXML_REAL`, `MXML_REAL_CALLBACK`, `MXML_TEXT`, `mxmlDelete()`, `mxmlFindElement()`, `mxmlIndexDelete()`, `mxmlIndexEnum()`, `mxmlIndexFind()`, `mxmlIndexNew()`, `mxmlIndexReset()`, `mxmlLoadFd()`, `mxmlLoadFile()`, `mxmlLoadString()`, `mxmlNewElement()`, `mxmlNewInteger()`, `mxmlNewOpaque()`, `mxmlNewReal()`, `mxmlNewText()`, `mxmlSaveFd()`, `mxmlSaveFile()`, `mxmlSaveString()`, `mxml_value_s::name`, `mxml_node_s::next`, `node`, `mxml_index_s::num_nodes`, `O_BINARY`, `mxml_value_u::opaque`, `mxml_value_u::real`, `mxml_text_s::string`, `mxml_value_u::text`, `mxml_node_s::type`, `type_cb()`, `mxml_node_s::value`, `mxml_text_s::whitespace`, and `whitespace_cb()`.

13.79.2.2 `mxml_type_t type_cb (mxml_node_t *node)`

Definition at line 542 of file testmxml.c.

References `mxml_value_u::element`, `MXML_INTEGER`, `MXML_OPAQUE`, `MXML_REAL`, `MXML_TEXT`, `mxmlElementGetAttr()`, `mxml_value_s::name`, and `mxml_node_s::value`.

Referenced by `main()`.

13.79.2.3 `const char* whitespace_cb (mxml_node_t * node, int where)`

Definition at line 571 of file `testmxml.c`.

References `mxml_node_s::child`, `mxml_value_u::element`, `MXML_WS_AFTER_CLOSE`, `MXML_WS_AFTER_OPEN`, `MXML_WS_BEFORE_CLOSE`, `MXML_WS_BEFORE_OPEN`, `mxml_value_s::name`, `mxml_node_s::parent`, and `mxml_node_s::value`.

13.80 /home/dko/Projects/mobilec/trunk/src/security/asm.c File Reference

```
#include "../include/mc_platform.h"
#include "../include/message.h"
#include "asm.h"
#include "asm_message_composer.h"
#include "config.h"
#include "mc_dh.h"
```

13.81 /home/dko/Projects/mobilec/trunk/src/security/asm.h File Reference

```
#include "config.h"
#include "../include/ap_queue_template.h"
#include "../include/data_structures.h"
#include "../mc_list/list.h"
#include "asm_node.h"
```

13.82 /home/dko/Projects/mobilec/trunk/src/security/asm_message_composer.c File Reference

```
#include "asm_message_composer.h"  
#include "config.h"  
#include "../include/mc_platform.h"
```

13.83 /home/dko/Projects/mobilec/trunk/src/security/asm_-message_composer.h File Reference

```
#include <mxml.h>
#include "asm.h"
#include "config.h"
```

13.84 /home/dko/Projects/mobilec/trunk/src/security/asm_message_parser.c File Reference

```
#include "asm_message_parser.h"  
#include "../include/xml_helper.h"  
#include "config.h"
```

13.85 /home/dko/Projects/mobilec/trunk/src/security/asm_- message_parser.h File Reference

```
#include "../include/mc_error.h"  
#include "../include/xml_parser.h"  
#include "asm_node.h"  
#include "config.h"
```

13.86 /home/dko/Projects/mobilec/trunk/src/security/asm_node.c File Reference

```
#include <netdb.h>
#include "config.h"
#include "asm.h"
#include "asm_message_parser.h"
#include "asm_node.h"
#include "xyssl-0.9/include/xyssl/havege.h"
#include "xyssl-0.9/include/xyssl/bignum.h"
```

13.87 /home/dko/Projects/mobilec/trunk/src/security/asm_node.h

File Reference

```
#include <mxml.h>
#include <netinet/in.h>
#include "../include/macros.h"
#include "../include/message.h"
#include "xyssl-0.9/include/xyssl/dhm.h"
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "xyssl-0.9/include/xyssl/aes.h"
#include "config.h"
```

13.88 /home/dko/Projects/mobilec/trunk/src/security/interface.c File Reference

```
#include "interface.h"
```

Functions

- static void [mystrncpy_binary](#) (char *dest, char *src, [int](#) start_index, [int](#) length)
- static void [separate_key_parts](#) (char *key, char *N, char *E, char *D, char *P, char *Q, char *DP, char *DQ, char *QP)
- [int](#) [rsa_encryption](#) (char *publickey, char *plaintext, char *ciphertext)
- [int](#) [rsa_decryption](#) (char *ciphertext, char *plaintext, char *privatekey)
- static [int](#) [append_nonce_to_MA](#) ([int](#) *my_nonce, char *MA_file)
- static [int](#) [remove_nonce_from_MA](#) (char *MA_file)
- static [int](#) [extract_nonce_from_MA](#) ([int](#) sockfd, [int](#) *my_nonce, char *MA_file)
- [int](#) [read_known_host_file](#) (char *pubkey, char *hname, char *filename)
- [int](#) [read_encrypted_file](#) (char *enfile, char *string, unsigned char *passphrase)
- [int](#) [initiate_migration_process](#) ([int](#) new_fd, [int](#) *my_nonce, char *pubkey, char *privkey, unsigned char *aes_key)
- [int](#) [reply_migration_process](#) ([int](#) sockfd, [int](#) *my_nonce, char *pubkey, char *privkey, unsigned char *aes_key)
- void [generate_AES_key](#) (char *key)
- [int](#) [aes_en_de](#) ([int](#) mode, char *infile, char *outfile, unsigned char *AES_key, [int](#) *nonce, [int](#) new_fd)
- [int](#) [send_AES_en_MA](#) ([int](#) sockfd, [int](#) *my_nonce, char *outfile, char *pubkey)
- [int](#) [receive_AES_en_MA](#) ([int](#) new_fd, [int](#) *nonce, char *infile, char *privkey)
- [int](#) [generate_RSA_keys_plaintext](#) (char *pubkeyfile, char *privkeyfile)
- [int](#) [generate_RSA_keys_ciphertext](#) (char *pubkeyfile, char *privkeyfile, unsigned char *passphrase)

13.88.1 Function Documentation

13.88.1.1 [int](#) [aes_en_de](#) ([int](#) mode, char *infile, char *outfile, unsigned char *AES_key, [int](#) *nonce, [int](#) new_fd)

Definition at line 1112 of file interface.c.

References [aes_crypt_ecb\(\)](#), [AES_DECRYPT](#), [AES_ENCRYPT](#), [aes_setkey_dec\(\)](#), [aes_setkey_enc\(\)](#), [append_nonce_to_MA\(\)](#), [extract_nonce_from_MA\(\)](#), [int](#), [MODE_DECRYPT](#), [MODE_ENCRYPT](#), [sha2_finish\(\)](#), [sha2_hmac_finish\(\)](#), [sha2_hmac_starts\(\)](#), [sha2_hmac_update\(\)](#), [sha2_starts\(\)](#), and [sha2_update\(\)](#).

Referenced by [auth_rece_send_msg\(\)](#), [generate_RSA_keys_ciphertext\(\)](#), [read_encrypted_file\(\)](#), and [rece_de_msg\(\)](#).

13.88.1.2 [static int](#) [append_nonce_to_MA](#) ([int](#) *my_nonce, char *MA_file) [[static](#)]

Definition at line 176 of file interface.c.

Referenced by [aes_en_de\(\)](#).

**13.88.1.3 static int extract_nonce_from_MA (int *sockfd*, int * *my_nonce*, char * *MA_file*)
[static]**

Definition at line 253 of file interface.c.

References `remove_nonce_from_MA()`, and `send`.

Referenced by `aes_en_de()`.

13.88.1.4 void generate_AES_key (char * *key*)

Definition at line 1089 of file interface.c.

References `havege_init()`, and `havege_rand()`.

Referenced by `initiate_migration_process()`, and `reply_migration_process()`.

13.88.1.5 int generate_RSA_keys_ciphertext (char * *pubkeyfile*, char * *privkeyfile*, unsigned char * *passphrase*)

Definition at line 1743 of file interface.c.

References `aes_en_de()`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `mpi_write_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_gen_key()`, `rsa_init()`, and `RSA_PKCS_V15`.

Referenced by `main()`.

13.88.1.6 int generate_RSA_keys_plaintext (char * *pubkeyfile*, char * *privkeyfile*)

Definition at line 1684 of file interface.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `mpi_write_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_gen_key()`, `rsa_init()`, and `RSA_PKCS_V15`.

Referenced by `main()`.

13.88.1.7 int initiate_migration_process (int *new_fd*, int * *my_nonce*, char * *pubkey*, char * *privkey*, unsigned char * *aes_key*)

Definition at line 422 of file interface.c.

References `generate_AES_key()`, `havege_init()`, `havege_rand()`, `md5()`, `rsa_decryption()`, `rsa_encryption()`, and `send`.

Referenced by `auth_rece_send_msg()`.

**13.88.1.8 static void mystrncpy_binary (char * *dest*, char * *src*, int *start_index*, int *length*)
[static]**

Definition at line 13 of file interface.c.

13.88.1.9 int read_encrypted_file (char * *enfile*, char * *string*, unsigned char * *passphase*)

Definition at line 388 of file interface.c.

References aes_en_de(), and f.

Referenced by MC_Initialize().

13.88.1.10 int read_known_host_file (char * *pubkey*, char * *hname*, char * *filename*)

Definition at line 319 of file interface.c.

Referenced by auth_conn_rece_key(), and auth_rece_send_msg().

13.88.1.11 int receive_AES_en_MA (int *new_fd*, int * *nonce*, char * *infile*, char * *privkey*)

Definition at line 1546 of file interface.c.

References rsa_decryption(), and size.

Referenced by rece_de_msg().

13.88.1.12 static int remove_nonce_from_MA (char * *MA_file*) [static]

Definition at line 209 of file interface.c.

References int.

Referenced by extract_nonce_from_MA().

13.88.1.13 int reply_migration_process (int *sockfd*, int * *my_nonce*, char * *pubkey*, char * *privkey*, unsigned char * *aes_key*)

Definition at line 767 of file interface.c.

References generate_AES_key(), md5(), rsa_decryption(), rsa_encryption(), and send.

Referenced by auth_conn_rece_key().

13.88.1.14 int rsa_decryption (char * *ciphertext*, char * *plaintext*, char * *privatekey*)

Definition at line 125 of file interface.c.

References rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, rsa_context::len, mpi_msb(), mpi_read_mystring(), rsa_context::N, rsa_context::P, P, rsa_context::Q, rsa_context::QP, rsa_check_privkey(), RSA_DE, rsa_init(), rsa_pkcs1_decrypt(), RSA_PKCS_V15, and separate_key_parts().

Referenced by initiate_migration_process(), receive_AES_en_MA(), and reply_migration_process().

13.88.1.15 int rsa_encryption (char * *publickey*, char * *plaintext*, char * *ciphertext*)

Definition at line 88 of file interface.c.

References rsa_context::E, rsa_context::len, mpi_msb(), mpi_read_mystring(), rsa_context::N, rsa_check_pubkey(), RSA_EN, rsa_init(), rsa_pkcs1_encrypt(), RSA_PKCS_V15, and separate_key_parts().

Referenced by `initiate_migration_process()`, `reply_migration_process()`, and `send_AES_en_MA()`.

13.88.1.16 `int send_AES_en_MA (int sockfd, int * my_nonce, char * outfile, char * pubkey)`

Definition at line 1410 of file `interface.c`.

References `int`, `rsa_encryption()`, `send`, and `size`.

Referenced by `auth_rece_send_msg()`.

13.88.1.17 `static void separate_key_parts (char * key, char * N, char * E, char * D, char * P,
char * Q, char * DP, char * DQ, char * QP) [static]`

Definition at line 25 of file `interface.c`.

Referenced by `rsa_decryption()`, and `rsa_encryption()`.

13.89 /home/dko/Projects/mobilec/trunk/src/security/interface.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/stat.h>
#include "xyssl-0.9/include/xyssl/havege.h"
#include "xyssl-0.9/include/xyssl/bignum.h"
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "xyssl-0.9/include/xyssl/sha2.h"
#include "xyssl-0.9/include/xyssl/sha1.h"
#include "xyssl-0.9/include/xyssl/aes.h"
#include "xyssl-0.9/include/xyssl/md5.h"
```

Defines

- #define [PATH](#) xyssl-0.9/include/xyssl
- #define [MODE_ENCRYPT](#) 0
- #define [MODE_DECRYPT](#) 1
- #define [MAXDATASIZE](#) 4096
- #define [KEY_SIZE](#) 1024
- #define [EXPONENT](#) 65537
- #define [RSA_EN](#) 0
- #define [RSA_DE](#) 1

Functions

- static void [mystrncpy](#) (char *dest, char *src, [int](#) start_index, [int](#) length)
- [int](#) [read_known_host_file](#) (char *pubkey, char *hname, char *filename)
- static void [separate_key_parts](#) (char *pubkey, char *N, char *E, char *D, char *P, char *Q, char *DP, char *DQ, char *QP)
- [int](#) [read_encrypted_file](#) (char *enfile, char *string, unsigned char *passphrase)
- [int](#) [rsa_encryption](#) (char *publickey, char *plaintext, char *ciphertext)
- [int](#) [rsa_decryption](#) (char *ciphertext, char *plaintext, char *privatekeyfile)
- [int](#) [initiate_migration_process](#) ([int](#) sockfd, [int](#) *nonce, char *publickey, char *privatekey, unsigned char *aes_key)
- [int](#) [reply_migration_process](#) ([int](#) new_fd, [int](#) *nonce, char *publickey, char *privatekey, unsigned char *aes_key)
- static void [generate_AES_key](#) (char *key)
- [int](#) [generate_encrypt_send_AES_key](#) ([int](#) sockfd, [int](#) *nonce, unsigned char *key, char *publickey, char *privkey)
- [int](#) [receive_decrypt_AES_key](#) ([int](#) new_fd, [int](#) *nonce, unsigned char *key, char *privkey, char *publickey)
- [int](#) [aes_en_de](#) ([int](#) mode, char *infile, char *outfile, unsigned char *AES_key, [int](#) *nonce, [int](#) sockfd)

- static [int append_nonce_to_MA](#) ([int](#) *my_nonce, char *MA_file)
- static [int extract_nonce_from_MA](#) ([int](#) sockfd, [int](#) *my_nonce, char *MA_file)
- [int send_AES_en_MA](#) ([int](#) sockfd, [int](#) *nonce, char *outfile, char *peer_pubkey)
- [int receive_AES_en_MA](#) ([int](#) new_fd, [int](#) *nonce, char *infile, char *privatekey)
- [int receiving_verifying_MA_RSA](#) ([int](#) sockfd, char *privkeyfile)
- [int receiving_decrypting_MA_RSA](#) ([int](#) new_fd, char *privkeyfile)
- [int generate_RSA_keys_plaintext](#) (char *pubkeyfile, char *privkeyfile)
- [int generate_RSA_keys_ciphertext](#) (char *pubkeyfile, char *privkeyfile, unsigned char *passphrase)

13.89.1 Define Documentation

13.89.1.1 #define EXPONENT 65537

Definition at line 25 of file interface.h.

Referenced by [generate_RSA_keys_ciphertext\(\)](#), [generate_RSA_keys_plaintext\(\)](#), and [main\(\)](#).

13.89.1.2 #define KEY_SIZE 1024

Definition at line 24 of file interface.h.

Referenced by [generate_RSA_keys_ciphertext\(\)](#), [generate_RSA_keys_plaintext\(\)](#), and [main\(\)](#).

13.89.1.3 #define MAXDATASIZE 4096

Definition at line 22 of file interface.h.

13.89.1.4 #define MODE_DECRYPT 1

Definition at line 21 of file interface.h.

Referenced by [aes_en_de\(\)](#), and [main\(\)](#).

13.89.1.5 #define MODE_ENCRYPT 0

Definition at line 20 of file interface.h.

Referenced by [aes_en_de\(\)](#), and [main\(\)](#).

13.89.1.6 #define PATH xyssl-0.9/include/xyssl

Definition at line 9 of file interface.h.

13.89.1.7 #define RSA_DE 1

Definition at line 28 of file interface.h.

Referenced by [rsa_decryption\(\)](#).

13.89.1.8 #define RSA_EN 0

Definition at line 27 of file interface.h.

Referenced by rsa_encryption().

13.89.2 Function Documentation**13.89.2.1 int aes_en_de (int *mode*, char * *infile*, char * *outfile*, unsigned char * *AES_key*, int * *nonce*, int *sockfd*)**

Definition at line 1112 of file interface.c.

References aes_crypt_ecb(), AES_DECRYPT, AES_ENCRYPT, aes_setkey_dec(), aes_setkey_enc(), append_nonce_to_MA(), extract_nonce_from_MA(), int, MODE_DECRYPT, MODE_ENCRYPT, sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), sha2_hmac_update(), sha2_starts(), and sha2_update().

Referenced by auth_rece_send_msg(), generate_RSA_keys_ciphertext(), read_encrypted_file(), and rece_de_msg().

13.89.2.2 static int append_nonce_to_MA (int * *my_nonce*, char * *MA_file*) [static]**13.89.2.3 static int extract_nonce_from_MA (int *sockfd*, int * *my_nonce*, char * *MA_file*) [static]****13.89.2.4 static void generate_AES_key (char * *key*) [static]**

Definition at line 1089 of file interface.c.

References havege_init(), and havege_rand().

Referenced by initiate_migration_process(), and reply_migration_process().

13.89.2.5 int generate_encrypt_send_AES_key (int *sockfd*, int * *nonce*, unsigned char * *key*, char * *publickey*, char * *privkey*)**13.89.2.6 int generate_RSA_keys_ciphertext (char * *pubkeyfile*, char * *privkeyfile*, unsigned char * *passphrase*)**

Definition at line 1743 of file interface.c.

References aes_en_de(), rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, EXPONENT, havege_init(), havege_rand(), KEY_SIZE, mpi_write_file(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, rsa_gen_key(), rsa_init(), and RSA_PKCS_V15.

Referenced by main().

13.89.2.7 int generate_RSA_keys_plaintext (char * *pubkeyfile*, char * *privkeyfile*)

Definition at line 1684 of file interface.c.

References rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, EXPONENT, havege_init(), havege_rand(), KEY_SIZE, mpi_write_file(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, rsa_gen_key(), rsa_init(), and RSA_PKCS_V15.

Referenced by `main()`.

13.89.2.8 `int initiate_migration_process (int sockfd, int * nonce, char * publickey, char * privatekey, unsigned char * aes_key)`

Definition at line 422 of file `interface.c`.

References `generate_AES_key()`, `havege_init()`, `havege_rand()`, `md5()`, `rsa_decryption()`, `rsa_encryption()`, and `send`.

Referenced by `auth_rece_send_msg()`.

13.89.2.9 `static void mystrncpy (char * dest, char * src, int start_index, int length) [static]`

13.89.2.10 `int read_encrypted_file (char * enfile, char * string, unsigned char * passphrase)`

Definition at line 388 of file `interface.c`.

References `aes_en_de()`, and `f`.

Referenced by `MC_Initialize()`.

13.89.2.11 `int read_known_host_file (char * pubkey, char * hname, char * filename)`

Definition at line 319 of file `interface.c`.

Referenced by `auth_conn_rece_key()`, and `auth_rece_send_msg()`.

13.89.2.12 `int receive_AES_en_MA (int new_fd, int * nonce, char * infile, char * privatekey)`

Definition at line 1546 of file `interface.c`.

References `rsa_decryption()`, and `size`.

Referenced by `rece_de_msg()`.

13.89.2.13 `int receive_decrypt_AES_key (int new_fd, int * nonce, unsigned char * key, char * privkey, char * publickey)`

13.89.2.14 `int receiving_decrypting_MA_RSA (int new_fd, char * privkeyfile)`

13.89.2.15 `int receiving_verifying_MA_RSA (int sockfd, char * privkeyfile)`

13.89.2.16 `int reply_migration_process (int new_fd, int * nonce, char * publickey, char * privatekey, unsigned char * aes_key)`

Definition at line 767 of file `interface.c`.

References `generate_AES_key()`, `md5()`, `rsa_decryption()`, `rsa_encryption()`, and `send`.

Referenced by `auth_conn_rece_key()`.

13.89.2.17 int rsa_decryption (char * *ciphertext*, char * *plaintext*, char * *privatekeyfile*)

Definition at line 125 of file interface.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::len`, `mpi_msb()`, `mpi_read_mystring()`, `rsa_context::N`, `rsa_context::P`, `P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `RSA_DE`, `rsa_init()`, `rsa_pkcs1_decrypt()`, `RSA_PKCS_V15`, and `separate_key_parts()`.

Referenced by `initiate_migration_process()`, `receive_AES_en_MA()`, and `reply_migration_process()`.

13.89.2.18 int rsa_encryption (char * *publickey*, char * *plaintext*, char * *ciphertext*)

Definition at line 88 of file interface.c.

References `rsa_context::E`, `rsa_context::len`, `mpi_msb()`, `mpi_read_mystring()`, `rsa_context::N`, `rsa_check_pubkey()`, `RSA_EN`, `rsa_init()`, `rsa_pkcs1_encrypt()`, `RSA_PKCS_V15`, and `separate_key_parts()`.

Referenced by `initiate_migration_process()`, `reply_migration_process()`, and `send_AES_en_MA()`.

13.89.2.19 int send_AES_en_MA (int *sockfd*, int * *nonce*, char * *outfile*, char * *peer_pubkey*)

Definition at line 1410 of file interface.c.

References `int`, `rsa_encryption()`, `send`, and `size`.

Referenced by `auth_rece_send_msg()`.

13.89.2.20 static void separate_key_parts (char * *pubkey*, char * *N*, char * *E*, char * *D*, char * *P*, char * *Q*, char * *DP*, char * *DQ*, char * *QP*) [static]

13.90 /home/dko/Projects/mobilec/trunk/src/security/mc_dh.c File Reference

```
#include <stdio.h>
#include "../include/mc_error.h"
#include "xyssl-0.9/include/xyssl/dhm.h"
#include "xyssl-0.9/include/xyssl/havege.h"
#include "mc_dh.h"
#include "asm_node.h"
#include "config.h"
```

13.91 /home/dko/Projects/mobilec/trunk/src/security/mc_dh.h File Reference

```
#include "xyssl-0.9/include/xyssl/rsa.h"
#include "asm_node.h"
#include "config.h"
```

13.92 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/aes.h File Reference

Data Structures

- struct [aes_context](#)
AES context structure.

Defines

- #define [AES_ENCRYPT](#) 1
- #define [AES_DECRYPT](#) 0

Functions

- void [aes_setkey_enc](#) ([aes_context](#) *ctx, unsigned char *key, [int](#) keysize)
AES key schedule (encryption).
- void [aes_setkey_dec](#) ([aes_context](#) *ctx, unsigned char *key, [int](#) keysize)
AES key schedule (decryption).
- void [aes_crypt_ecb](#) ([aes_context](#) *ctx, [int](#) mode, unsigned char input[16], unsigned char output[16])
AES-ECB block encryption/decryption.
- void [aes_crypt_cbc](#) ([aes_context](#) *ctx, [int](#) mode, [int](#) length, unsigned char iv[16], unsigned char *input, unsigned char *output)
AES-CBC buffer encryption/decryption.
- void [aes_crypt_cfb](#) ([aes_context](#) *ctx, [int](#) mode, [int](#) length, [int](#) *iv_off, unsigned char iv[16], unsigned char *input, unsigned char *output)
AES-CFB buffer encryption/decryption.
- [int](#) [aes_self_test](#) ([int](#) verbose)
Checkup routine.

13.92.1 Detailed Description

Definition in file [aes.h](#).

13.92.2 Define Documentation

13.92.2.1 #define AES_DECRYPT 0

Definition at line 8 of file [aes.h](#).

Referenced by [aes_crypt_cbc\(\)](#), [aes_crypt_cfb\(\)](#), [aes_crypt_ecb\(\)](#), [aes_en_de\(\)](#), [aes_self_test\(\)](#), [main\(\)](#), and [ssl_decrypt_buf\(\)](#).

13.92.2.2 #define AES_ENCRYPT 1

Definition at line 7 of file aes.h.

Referenced by aes_crypt_cfb(), aes_en_de(), main(), and ssl_encrypt_buf().

13.92.3 Function Documentation

13.92.3.1 void aes_crypt_cbc (aes_context * ctx, int mode, int length, unsigned char iv[16], unsigned char * input, unsigned char * output)

AES-CBC buffer encryption/decryption.

Parameters:

ctx AES context
mode AES_ENCRYPT or AES_DECRYPT
length length of the input data
iv initialization vector (updated after use)
input buffer holding the input data
output buffer holding the output data

Definition at line 732 of file aes.c.

References aes_crypt_ecb(), and AES_DECRYPT.

Referenced by aes_self_test(), main(), ssl_decrypt_buf(), and ssl_encrypt_buf().

13.92.3.2 void aes_crypt_cfb (aes_context * ctx, int mode, int length, int * iv_off, unsigned char iv[16], unsigned char * input, unsigned char * output)

AES-CFB buffer encryption/decryption.

Parameters:

ctx AES context
mode AES_ENCRYPT or AES_DECRYPT
length length of the input data
iv_off offset in IV (updated after use)
iv initialization vector (updated after use)
input buffer holding the input data
output buffer holding the output data

Definition at line 787 of file aes.c.

References aes_crypt_ecb(), AES_DECRYPT, and AES_ENCRYPT.

Referenced by aes_self_test().

13.92.3.3 void aes_crypt_ecb (aes_context * *ctx*, int *mode*, unsigned char *input*[16], unsigned char *output*[16])

AES-ECB block encryption/decryption.

Parameters:

ctx AES context
mode AES_ENCRYPT or AES_DECRYPT
input 16-byte input block
output 16-byte output block

Definition at line 639 of file aes.c.

References AES_DECRYPT, AES_FROUND, AES_RROUND, FSb, GET_ULONG_LE, aes_context::nr, PUT_ULONG_LE, aes_context::rk, and RSb.

Referenced by aes_crypt_cbc(), aes_crypt_cfb(), aes_en_de(), aes_self_test(), and main().

13.92.3.4 int aes_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 902 of file aes.c.

References aes_crypt_cbc(), aes_crypt_cfb(), aes_crypt_ecb(), AES_DECRYPT, aes_setkey_dec(), aes_setkey_enc(), aes_test_cbc_dec, aes_test_cbc_enc, aes_test_cfb_dec, aes_test_cfb_enc, aes_test_ecb_dec, aes_test_ecb_enc, buf, and prv.

Referenced by main().

13.92.3.5 void aes_setkey_dec (aes_context * *ctx*, unsigned char * *key*, int *keysize*)

AES key schedule (decryption).

Parameters:

ctx AES context to be initialized
key decryption key
keysize must be 128, 192 or 256

Definition at line 542 of file aes.c.

References aes_setkey_enc(), aes_context::buf, FSb, aes_context::nr, aes_context::rk, RT0, RT1, RT2, and RT3.

Referenced by aes_en_de(), aes_self_test(), main(), and ssl_derive_keys().

13.92.3.6 void aes_setkey_enc (aes_context * *ctx*, unsigned char * *key*, int *keysize*)

AES key schedule (encryption).

Parameters:

ctx AES context to be initialized

key encryption key

keysize must be 128, 192 or 256

Definition at line 439 of file aes.c.

References `aes_gen_tables()`, `aes_init_done`, `aes_context::buf`, `FSb`, `GET_ULONG_LE`, `aes_context::nr`, `RCON`, and `aes_context::rk`.

Referenced by `aes_en_de()`, `aes_self_test()`, `aes_setkey_dec()`, `main()`, and `ssl_derive_keys()`.

13.93 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/arc4.h File Reference

Data Structures

- struct [arc4_context](#)
ARC4 context structure.

Functions

- void [arc4_setup](#) ([arc4_context](#) *ctx, unsigned char *key, [int](#) keylen)
ARC4 key schedule.
- void [arc4_crypt](#) ([arc4_context](#) *ctx, unsigned char *buf, [int](#) buflen)
ARC4 cipher function.
- [int](#) [arc4_self_test](#) ([int](#) verbose)

13.93.1 Detailed Description

Definition in file [arc4.h](#).

13.93.2 Function Documentation

13.93.2.1 void [arc4_crypt](#) ([arc4_context](#) * ctx, unsigned char * buf, [int](#) buflen)

ARC4 cipher function.

Parameters:

ctx ARC4 context
buf buffer to be processed
buflen amount of data in buf

Definition at line 63 of file [arc4.c](#).

References [arc4_context::m](#), [arc4_context::x](#), and [arc4_context::y](#).

Referenced by [arc4_self_test\(\)](#), [main\(\)](#), [ssl_decrypt_buf\(\)](#), and [ssl_encrypt_buf\(\)](#).

13.93.2.2 [int](#) [arc4_self_test](#) ([int](#) verbose)

Definition at line 122 of file [arc4.c](#).

References [arc4_crypt\(\)](#), [arc4_setup\(\)](#), [arc4_test_ct](#), [arc4_test_key](#), [arc4_test_pt](#), and [buf](#).

Referenced by [main\(\)](#).

13.93.2.3 void arc4_setup (arc4_context * *ctx*, unsigned char * *key*, int *keylen*)

ARC4 key schedule.

Parameters:

ctx ARC4 context to be initialized

key the secret key

keylen length of the key

Definition at line 35 of file arc4.c.

References arc4_context::m, arc4_context::x, and arc4_context::y.

Referenced by arc4_self_test(), main(), and ssl_derive_keys().

13.94 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/base64.h File Reference

Defines

- `#define XYSSL_ERR_BASE64_BUFFER_TOO_SMALL -0x0010`
- `#define XYSSL_ERR_BASE64_INVALID_CHARACTER -0x0012`

Functions

- `int base64_encode` (unsigned char *dst, int *dlen, unsigned char *src, int slen)
Encode a buffer into base64 format.
- `int base64_decode` (unsigned char *dst, int *dlen, unsigned char *src, int slen)
Decode a base64-formatted buffer.
- `int base64_self_test` (int verbose)
Checkup routine.

13.94.1 Detailed Description

Definition in file [base64.h](#).

13.94.2 Define Documentation

13.94.2.1 `#define XYSSL_ERR_BASE64_BUFFER_TOO_SMALL -0x0010`

Definition at line 7 of file [base64.h](#).

Referenced by [base64_decode\(\)](#), and [base64_encode\(\)](#).

13.94.2.2 `#define XYSSL_ERR_BASE64_INVALID_CHARACTER -0x0012`

Definition at line 8 of file [base64.h](#).

Referenced by [base64_decode\(\)](#), [x509parse_crt\(\)](#), and [x509parse_key\(\)](#).

13.94.3 Function Documentation

13.94.3.1 `int base64_decode` (unsigned char *dst, int *dlen, unsigned char *src, int slen)

Decode a base64-formatted buffer.

Parameters:

- dst* destination buffer
- dlen* size of the buffer
- src* source buffer

slen amount of data to be decoded

Returns:

0 if successful, XYSSL_ERR_BASE64_BUFFER_TOO_SMALL, or XYSSL_ERR_BASE64_INVALID_DATA if the input data is not correct. *dlen is always updated to reflect the amount of data that has (or would have) been written.

Note:

Call this function with *dlen = 0 to obtain the required buffer size in *dlen

Definition at line 121 of file base64.c.

References base64_dec_map, XYSSL_ERR_BASE64_BUFFER_TOO_SMALL, and XYSSL_ERR_BASE64_INVALID_CHARACTER.

Referenced by base64_self_test(), x509parse_crt(), and x509parse_key().

13.94.3.2 int base64_encode (unsigned char *dst, int *dlen, unsigned char *src, int slen)

Encode a buffer into base64 format.

Parameters:

dst destination buffer

dlen size of the buffer

src source buffer

slen amount of data to be encoded

Returns:

0 if successful, or XYSSL_ERR_BASE64_BUFFER_TOO_SMALL. *dlen is always updated to reflect the amount of data that has (or would have) been written.

Note:

Call this function with *dlen = 0 to obtain the required buffer size in *dlen

Definition at line 58 of file base64.c.

References base64_enc_map, and XYSSL_ERR_BASE64_BUFFER_TOO_SMALL.

Referenced by base64_self_test().

13.94.3.3 int base64_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 206 of file base64.c.

References base64_decode(), base64_encode(), base64_test_dec, and base64_test_enc.

Referenced by main().

13.95 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/bignum.h File Reference

```
#include <stdio.h>
```

Data Structures

- struct [mpi](#)
MPI structure.

Defines

- #define [XYSSL_ERR_MPI_FILE_IO_ERROR](#) -0x0002
- #define [XYSSL_ERR_MPI_BAD_INPUT_DATA](#) -0x0004
- #define [XYSSL_ERR_MPI_INVALID_CHARACTER](#) -0x0006
- #define [XYSSL_ERR_MPI_BUFFER_TOO_SMALL](#) -0x0008
- #define [XYSSL_ERR_MPI_NEGATIVE_VALUE](#) -0x000A
- #define [XYSSL_ERR_MPI_DIVISION_BY_ZERO](#) -0x000C
- #define [XYSSL_ERR_MPI_NOT_ACCEPTABLE](#) -0x000E
- #define [MPI_CHK](#)(f) if((ret = f) != 0) goto cleanup

Typedefs

- typedef unsigned long [t_int](#)
- typedef unsigned long long [t_dbl](#)

Functions

- void [mpi_init](#) ([mpi](#) *X,...)
Initialize one or more [mpi](#).
- void [mpi_free](#) ([mpi](#) *X,...)
Unallocate one or more [mpi](#).
- int [mpi_grow](#) ([mpi](#) *X, int nlimbs)
Enlarge to the specified number of limbs.
- int [mpi_copy](#) ([mpi](#) *X, [mpi](#) *Y)
Copy the contents of Y into X.
- void [mpi_swap](#) ([mpi](#) *X, [mpi](#) *Y)
Swap the contents of X and Y.
- int [mpi_lset](#) ([mpi](#) *X, int z)
Set value from integer.
- int [mpi_lsb](#) ([mpi](#) *X)

Return the number of least significant bits.

- `int mpi_msb (mpi *X)`

Return the number of most significant bits.

- `int mpi_size (mpi *X)`

Return the total size in bytes.

- `int mpi_read_string (mpi *X, int radix, char *s)`

Import from an ASCII string.

- `int mpi_write_string (mpi *X, int radix, char *s, int *slen)`

Export into an ASCII string.

- `int mpi_read_file (mpi *X, int radix, FILE *fin)`

Read X from an opened file.

- `int mpi_read_mystring (mpi *X, int radix, char *s)`

- `int mpi_write_file (char *p, mpi *X, int radix, FILE *fout)`

Write X into an opened file, or stdout.

- `int mpi_read_binary (mpi *X, unsigned char *buf, int buflen)`

Import X from unsigned binary data, big endian.

- `int mpi_write_binary (mpi *X, unsigned char *buf, int buflen)`

Export X into unsigned binary data, big endian.

- `int mpi_shift_l (mpi *X, int count)`

Left-shift: $X \ll = \text{count}$.

- `int mpi_shift_r (mpi *X, int count)`

Right-shift: $X \gg = \text{count}$.

- `int mpi_cmp_abs (mpi *X, mpi *Y)`

Compare unsigned values.

- `int mpi_cmp_mpi (mpi *X, mpi *Y)`

Compare signed values.

- `int mpi_cmp_int (mpi *X, int z)`

Compare signed values.

- `int mpi_add_abs (mpi *X, mpi *A, mpi *B)`

Unsigned addition: $X = |A| + |B|$.

- `int mpi_sub_abs (mpi *X, mpi *A, mpi *B)`

Unsigned subtraction: $X = |A| - |B|$.

- `int mpi_add_mpi (mpi *X, mpi *A, mpi *B)`

Signed addition: $X = A + B$.

- `int mpi_sub_mpi (mpi *X, mpi *A, mpi *B)`
Signed subtraction: $X = A - B$.
- `int mpi_add_int (mpi *X, mpi *A, int b)`
Signed addition: $X = A + b$.
- `int mpi_sub_int (mpi *X, mpi *A, int b)`
Signed subtraction: $X = A - b$.
- `int mpi_mul_mpi (mpi *X, mpi *A, mpi *B)`
*Baseline multiplication: $X = A * B$.*
- `int mpi_mul_int (mpi *X, mpi *A, t_int b)`
*Baseline multiplication: $X = A * b$.*
- `int mpi_div_mpi (mpi *Q, mpi *R, mpi *A, mpi *B)`
*Division by `mpi`: $A = Q * B + R$.*
- `int mpi_div_int (mpi *Q, mpi *R, mpi *A, int b)`
*Division by `int`: $A = Q * b + R$.*
- `int mpi_mod_mpi (mpi *R, mpi *A, mpi *B)`
Modulo: $R = A \bmod B$.
- `int mpi_mod_int (t_int *r, mpi *A, int b)`
Modulo: $r = A \bmod b$.
- `int mpi_exp_mod (mpi *X, mpi *A, mpi *E, mpi *N, mpi *_RR)`
Sliding-window exponentiation: $X = A^E \bmod N$.
- `int mpi_gcd (mpi *G, mpi *A, mpi *B)`
Greatest common divisor: $G = \gcd(A, B)$.
- `int mpi_inv_mod (mpi *X, mpi *A, mpi *N)`
Modular inverse: $X = A^{-1} \bmod N$.
- `int mpi_is_prime (mpi *X, int(*f_rng)(void *), void *p_rng)`
Miller-Rabin primality test.
- `int mpi_gen_prime (mpi *X, int nbits, int dh_flag, int(*f_rng)(void *), void *p_rng)`
Prime number generation.
- `int mpi_self_test (int verbose)`
Checkup routine.

13.95.1 Detailed Description

Definition in file [bignum.h](#).

13.95.2 Define Documentation

13.95.2.1 **#define MPI_CHK(f) if((ret = f) != 0) goto cleanup**

Definition at line 17 of file bignum.h.

Referenced by dhmm_calc_secret(), dhmm_make_params(), dhmm_make_public(), mpi_add_abs(), mpi_add_mpi(), mpi_copy(), mpi_div_mpi(), mpi_exp_mod(), mpi_gcd(), mpi_gen_prime(), mpi_inv_mod(), mpi_is_prime(), mpi_lset(), mpi_mod_mpi(), mpi_mul_mpi(), mpi_read_binary(), mpi_read_string(), mpi_self_test(), mpi_shift_l(), mpi_sub_abs(), mpi_sub_mpi(), mpi_write_file(), mpi_write_hlp(), mpi_write_string(), rsa_check_privkey(), rsa_gen_key(), rsa_private(), and rsa_public().

13.95.2.2 **#define XYSSL_ERR_MPI_BAD_INPUT_DATA -0x0004**

Definition at line 10 of file bignum.h.

Referenced by mpi_exp_mod(), mpi_gen_prime(), mpi_inv_mod(), mpi_read_string(), mpi_write_hlp(), and mpi_write_string().

13.95.2.3 **#define XYSSL_ERR_MPI_BUFFER_TOO_SMALL -0x0008**

Definition at line 12 of file bignum.h.

Referenced by mpi_write_binary(), and mpi_write_string().

13.95.2.4 **#define XYSSL_ERR_MPI_DIVISION_BY_ZERO -0x000C**

Definition at line 14 of file bignum.h.

Referenced by mpi_div_mpi(), and mpi_mod_int().

13.95.2.5 **#define XYSSL_ERR_MPI_FILE_IO_ERROR -0x0002**

Definition at line 9 of file bignum.h.

Referenced by mpi_read_file(), and mpi_write_file().

13.95.2.6 **#define XYSSL_ERR_MPI_INVALID_CHARACTER -0x0006**

Definition at line 11 of file bignum.h.

Referenced by mpi_get_digit().

13.95.2.7 **#define XYSSL_ERR_MPI_NEGATIVE_VALUE -0x000A**

Definition at line 13 of file bignum.h.

Referenced by mpi_sub_abs().

13.95.2.8 **#define XYSSL_ERR_MPI_NOT_ACCEPTABLE -0x000E**

Definition at line 15 of file bignum.h.

Referenced by `mpi_gen_prime()`, `mpi_inv_mod()`, and `mpi_is_prime()`.

13.95.3 Typedef Documentation

13.95.3.1 typedef unsigned long long t_dbl

Definition at line 39 of file `bignum.h`.

13.95.3.2 typedef unsigned long t_int

Definition at line 30 of file `bignum.h`.

13.95.4 Function Documentation

13.95.4.1 int mpi_add_abs (mpi * X, mpi * A, mpi * B)

Unsigned addition: $X = |A| + |B|$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 694 of file `bignum.c`.

References `MPI_CHK`, `mpi_copy()`, `mpi_grow()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, and `mpi_sub_mpi()`.

13.95.4.2 int mpi_add_int (mpi * X, mpi * A, int b)

Signed addition: $X = A + b$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 860 of file `bignum.c`.

References `mpi_add_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_gen_prime()`, and `mpi_read_string()`.

13.95.4.3 int mpi_add_mpi (mpi * X, mpi * A, mpi * B)

Signed addition: $X = A + B$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 798 of file `bignum.c`.

References `mpi_add_abs()`, `MPI_CHK`, `mpi_cmp_abs()`, `mpi_sub_abs()`, and `mpi::s`.

Referenced by `mpi_add_int()`, `mpi_div_mpi()`, `mpi_inv_mod()`, `mpi_mod_mpi()`, and `rsa_private()`.

13.95.4.4 int mpi_cmp_abs (mpi * X, mpi * Y)

Compare unsigned values.

Returns:

1 if $|X|$ is greater than $|Y|$, -1 if $|X|$ is lesser than $|Y|$ or 0 if $|X|$ is equal to $|Y|$

Definition at line 615 of file bignum.c.

References mpi::n, and mpi::p.

Referenced by mpi_add_mpi(), mpi_div_mpi(), mpi_montmul(), mpi_sub_abs(), and mpi_sub_mpi().

13.95.4.5 int mpi_cmp_int (mpi * X, int z)

Compare signed values.

Returns:

1 if X is greater than z, -1 if X is lesser than z or 0 if X is equal to z

Definition at line 678 of file bignum.c.

References mpi_cmp_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_div_mpi(), mpi_exp_mod(), mpi_gcd(), mpi_inv_mod(), mpi_is_prime(), mpi_mod_mpi(), mpi_write_hlp(), rsa_check_privkey(), and rsa_gen_key().

13.95.4.6 int mpi_cmp_mpi (mpi * X, mpi * Y)

Compare signed values.

Returns:

1 if X is greater than Y, -1 if X is lesser than Y or 0 if X is equal to Y

Definition at line 645 of file bignum.c.

References mpi::n, mpi::p, and mpi::s.

Referenced by dhm_make_params(), dhm_make_public(), mpi_cmp_int(), mpi_div_mpi(), mpi_exp_mod(), mpi_gcd(), mpi_inv_mod(), mpi_is_prime(), mpi_mod_mpi(), mpi_self_test(), rsa_check_privkey(), rsa_gen_key(), rsa_private(), and rsa_public().

13.95.4.7 int mpi_copy (mpi * X, mpi * Y)

Copy the contents of Y into X.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 128 of file bignum.c.

References ciL, MPI_CHK, mpi_grow(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_add_abs(), mpi_div_mpi(), mpi_exp_mod(), mpi_gcd(), mpi_inv_mod(), mpi_is_prime(), mpi_mul_mpi(), mpi_sub_abs(), and mpi_write_string().

13.95.4.8 int mpi_div_int (mpi * *Q*, mpi * *R*, mpi * *A*, int *b*)

Division by int: $A = Q * b + R$.

Returns:

0 if successful, 1 if memory allocation failed, XYSSL_ERR_MPI_DIVISION_BY_ZERO if $b == 0$

Note:

Either *Q* or *R* can be NULL.

Definition at line 1173 of file bignum.c.

References mpi_div_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), and mpi_write_hlp().

13.95.4.9 int mpi_div_mpi (mpi * *Q*, mpi * *R*, mpi * *A*, mpi * *B*)

Division by mpi: $A = Q * B + R$.

Returns:

0 if successful, 1 if memory allocation failed, XYSSL_ERR_MPI_DIVISION_BY_ZERO if $B == 0$

Note:

Either *Q* or *R* can be NULL.

Definition at line 1008 of file bignum.c.

References biH, biL, mpi_add_mpi(), MPI_CHK, mpi_cmp_abs(), mpi_cmp_int(), mpi_cmp_mpi(), mpi_copy(), mpi_free(), mpi_grow(), mpi_init(), mpi_lset(), mpi_msb(), mpi_mul_int(), mpi_shift_l(), mpi_shift_r(), mpi_sub_mpi(), mpi::n, mpi::p, mpi::s, and XYSSL_ERR_MPI_DIVISION_BY_ZERO.

Referenced by mpi_div_int(), mpi_mod_mpi(), and mpi_self_test().

13.95.4.10 int mpi_exp_mod (mpi * *X*, mpi * *A*, mpi * *E*, mpi * *N*, mpi * *_RR*)

Sliding-window exponentiation: $X = A^E \bmod N$.

Returns:

0 if successful, 1 if memory allocation failed, XYSSL_ERR_MPI_BAD_INPUT_DATA if *N* is negative or even

Note:

_RR is used to avoid re-computing $R * R \bmod N$ across multiple calls, which speeds up things a bit. It can be set to NULL if the extra performance is unneeded.

Definition at line 1328 of file bignum.c.

References biL, MPI_CHK, mpi_cmp_int(), mpi_cmp_mpi(), mpi_copy(), mpi_free(), mpi_grow(), mpi_init(), mpi_lset(), mpi_mod_mpi(), mpi_montg_init(), mpi_montmul(), mpi_montred(), mpi_msb(), mpi_shift_l(), mpi::n, mpi::p, and XYSSL_ERR_MPI_BAD_INPUT_DATA.

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_make_public(), main(), mpi_is_prime(), mpi_self_test(), rsa_private(), and rsa_public().

13.95.4.11 void mpi_free (mpi * X, ...)

Unallocate one or more [mpi](#).

Definition at line 73 of file bignum.c.

References [ciL](#), [mpi::n](#), [mpi::p](#), and [mpi::s](#).

Referenced by [dhm_free\(\)](#), [main\(\)](#), [mpi_div_mpi\(\)](#), [mpi_exp_mod\(\)](#), [mpi_gcd\(\)](#), [mpi_gen_prime\(\)](#), [mpi_inv_mod\(\)](#), [mpi_is_prime\(\)](#), [mpi_mul_mpi\(\)](#), [mpi_read_string\(\)](#), [mpi_self_test\(\)](#), [mpi_sub_abs\(\)](#), [mpi_write_string\(\)](#), [rsa_check_privkey\(\)](#), [rsa_free\(\)](#), [rsa_gen_key\(\)](#), [rsa_private\(\)](#), and [rsa_public\(\)](#).

13.95.4.12 int mpi_gcd (mpi * G, mpi * A, mpi * B)

Greatest common divisor: $G = \gcd(A, B)$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 1507 of file bignum.c.

References [MPI_CHK](#), [mpi_cmp_int\(\)](#), [mpi_cmp_mpi\(\)](#), [mpi_copy\(\)](#), [mpi_free\(\)](#), [mpi_init\(\)](#), [mpi_lset\(\)](#), [mpi_mul_mpi\(\)](#), [mpi_shift_r\(\)](#), [mpi_sub_abs\(\)](#), [mpi::p](#), and [mpi::s](#).

Referenced by [mpi_inv_mod\(\)](#), [rsa_check_privkey\(\)](#), and [rsa_gen_key\(\)](#).

13.95.4.13 int mpi_gen_prime (mpi * X, int nbits, int dh_flag, int(*) (void *) f_rng, void * p_rng)

Prime number generation.

Parameters:

X destination [mpi](#)

nbits required size of X in bits

dh_flag if 1, then $(X-1)/2$ will be prime too

f_rng RNG function

p_rng RNG parameter

Returns:

0 if successful (probably prime), 1 if memory allocation failed, [XYSSL_ERR_MPI_BAD_INPUT_DATA](#) if nbits is < 3

Definition at line 1778 of file bignum.c.

References [BITS_TO_LIMBS](#), [ciL](#), [mpi_add_int\(\)](#), [MPI_CHK](#), [mpi_free\(\)](#), [mpi_grow\(\)](#), [mpi_init\(\)](#), [mpi_is_prime\(\)](#), [mpi_lset\(\)](#), [mpi_msb\(\)](#), [mpi_shift_l\(\)](#), [mpi_shift_r\(\)](#), [mpi_sub_int\(\)](#), [mpi::p](#), [XYSSL_ERR_MPI_BAD_INPUT_DATA](#), and [XYSSL_ERR_MPI_NOT_ACCEPTABLE](#).

Referenced by [main\(\)](#), and [rsa_gen_key\(\)](#).

13.95.4.14 int mpi_grow (mpi * X, int nblimbs)

Enlarge to the specified number of limbs.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 100 of file bignum.c.

References `ciL`, `mpi::n`, and `mpi::p`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_add_abs()`, `mpi_copy()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gen_prime()`, `mpi_is_prime()`, `mpi_lset()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_read_string()`, and `mpi_shift_l()`.

13.95.4.15 void mpi_init (mpi * X, ...)

Initialize one or more [mpi](#).

Definition at line 52 of file bignum.c.

References `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_self_test()`, `mpi_sub_abs()`, `mpi_write_string()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

13.95.4.16 int mpi_inv_mod (mpi * X, mpi * A, mpi * N)

Modular inverse: $X = A^{-1} \bmod N$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_BAD_INPUT_DATA` if `N` is negative or nil `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `A` has no inverse mod `N`

Definition at line 1549 of file bignum.c.

References `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_shift_r()`, `mpi_sub_mpi()`, `mpi::p`, `XYSSL_ERR_MPI_BAD_INPUT_DATA`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, `mpi_self_test()`, and `rsa_gen_key()`.

13.95.4.17 int mpi_is_prime (mpi * X, int(*) (void *) f_rng, void * p_rng)

Miller-Rabin primality test.

Returns:

0 if successful (probably prime), 1 if memory allocation failed, `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `X` is not prime

Definition at line 1667 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_lsb()`, `mpi_mod_int()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_shift_r()`, `mpi_sub_int()`, `mpi::n`, `mpi::p`, `R`, `mpi::s`, `small_prime`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, and `mpi_gen_prime()`.

13.95.4.18 `int mpi_lsb (mpi * X)`

Return the number of least significant bits.

Definition at line 185 of file bignum.c.

References `biL`, `int`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_is_prime()`.

13.95.4.19 `int mpi_lset (mpi * X, int z)`

Set value from integer.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 167 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_grow()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_inv_mod()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_read_string()`, and `rsa_gen_key()`.

13.95.4.20 `int mpi_mod_int (t_int * r, mpi * A, int b)`

Modulo: $r = A \bmod b$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if $b == 0$

Definition at line 1209 of file bignum.c.

References `biH`, `mpi::n`, `mpi::p`, and `XYSSL_ERR_MPI_DIVISION_BY_ZERO`.

Referenced by `mpi_is_prime()`, and `mpi_write_hlp()`.

13.95.4.21 `int mpi_mod_mpi (mpi * R, mpi * A, mpi * B)`

Modulo: $R = A \bmod B$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if $B == 0$

Definition at line 1189 of file bignum.c.

References `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_div_mpi()`, and `mpi_sub_mpi()`.

Referenced by `mpi_exp_mod()`, `mpi_inv_mod()`, `mpi_is_prime()`, `rsa_check_privkey()`, `rsa_gen_key()`, and `rsa_private()`.

13.95.4.22 int mpi_msb (mpi * X)

Return the number of most significant bits.

Definition at line 200 of file bignum.c.

References `biL`, `mpi::n`, and `mpi::p`.

Referenced by `d2i_RSA_PUBKEY()`, `main()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gen_prime()`, `mpi_is_prime()`, `mpi_shift_l()`, `mpi_size()`, `mpi_write_string()`, `rsa_check_pubkey()`, `rsa_decryption()`, `rsa_encryption()`, and `rsa_gen_key()`.

13.95.4.23 int mpi_mul_int (mpi * X, mpi * A, t_int b)

Baseline multiplication: $X = A * b$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 992 of file bignum.c.

References `mpi_mul_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_div_mpi()`, and `mpi_read_string()`.

13.95.4.24 int mpi_mul_mpi (mpi * X, mpi * A, mpi * B)

Baseline multiplication: $X = A * B$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 956 of file bignum.c.

References `MPI_CHK`, `mpi_copy()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_lset()`, `mpi_mul_hlp()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, `mpi_gcd()`, `mpi_is_prime()`, `mpi_mul_int()`, `mpi_self_test()`, `rsa_check_privkey()`, `rsa_gen_key()`, and `rsa_private()`.

13.95.4.25 int mpi_read_binary (mpi * X, unsigned char * buf, int buflen)

Import X from unsigned binary data, big endian.

Parameters:

X destination `mpi`

buf input buffer

buflen input buffer size

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 484 of file bignum.c.

References CHARS_TO_LIMBS, ciL, MPI_CHK, mpi_grow(), mpi_lset(), and mpi::p.

Referenced by asn1_get_mpi(), d2i_RSA_PUBKEY(), dhm_read_bignum(), dhm_read_public(), rsa_private(), and rsa_public().

13.95.4.26 int mpi_read_file (mpi * X, int radix, FILE * fin)

Read X from an opened file.

Parameters:

- X* destination [mpi](#)
- radix* input numeric base
- fin* input file handle

Returns:

- 0 if successful, or an XYSSL_ERR_MPI_XXX error code

Definition at line 391 of file bignum.c.

References mpi_get_digit(), mpi_read_string(), and XYSSL_ERR_MPI_FILE_IO_ERROR.

Referenced by main().

13.95.4.27 int mpi_read_mystring (mpi * X, int radix, char * s)

Definition at line 418 of file bignum.c.

References mpi_get_digit(), and mpi_read_string().

Referenced by rsa_decryption(), and rsa_encryption().

13.95.4.28 int mpi_read_string (mpi * X, int radix, char * s)

Import from an ASCII string.

Parameters:

- X* destination [mpi](#)
- radix* input numeric base
- s* null-terminated string buffer

Returns:

- 0 if successful, or an XYSSL_ERR_MPI_XXX error code

Definition at line 243 of file bignum.c.

References BITS_TO_LIMBS, ciL, int, mpi_add_int(), MPI_CHK, mpi_free(), mpi_get_digit(), mpi_grow(), mpi_init(), mpi_lset(), mpi_mul_int(), mpi::p, mpi::s, and XYSSL_ERR_MPI_BAD_INPUT_DATA.

Referenced by main(), mpi_read_file(), mpi_read_mystring(), mpi_self_test(), rsa_self_test(), and ssl_set_dh_param().

13.95.4.29 int mpi_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1854 of file bignum.c.

References MPI_CHK, mpi_cmp_mpi(), mpi_div_mpi(), mpi_exp_mod(), mpi_free(), mpi_init(), mpi_inv_mod(), mpi_mul_mpi(), and mpi_read_string().

Referenced by main().

13.95.4.30 int mpi_shift_l (mpi * X, int count)

Left-shift: $X \ll= \text{count}$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 526 of file bignum.c.

References biL, BITS_TO_LIMBS, MPI_CHK, mpi_grow(), mpi_msb(), mpi::n, and mpi::p.

Referenced by mpi_div_mpi(), mpi_exp_mod(), and mpi_gen_prime().

13.95.4.31 int mpi_shift_r (mpi * X, int count)

Right-shift: $X \gg= \text{count}$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 575 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by dhm_make_params(), dhm_make_public(), mpi_div_mpi(), mpi_gcd(), mpi_gen_prime(), mpi_inv_mod(), and mpi_is_prime().

13.95.4.32 int mpi_size (mpi * X)

Return the total size in bytes.

Definition at line 218 of file bignum.c.

References mpi_msb().

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_read_params(), mpi_write_binary(), x509parse_crt(), and x509parse_key().

13.95.4.33 int mpi_sub_abs (mpi * X, mpi * A, mpi * B)

Unsigned subtraction: $X = |A| - |B|$.

Returns:

0 if successful, XYSSL_ERR_MPI_NEGATIVE_VALUE if B is greater than A

Definition at line 761 of file bignum.c.

References MPI_CHK, mpi_cmp_abs(), mpi_copy(), mpi_free(), mpi_init(), mpi_sub_hlp(), mpi::n, mpi::p, and XYSSL_ERR_MPI_NEGATIVE_VALUE.

Referenced by mpi_add_mpi(), mpi_gcd(), and mpi_sub_mpi().

13.95.4.34 int mpi_sub_int (mpi * X, mpi * A, int b)

Signed subtraction: $X = A - b$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 876 of file bignum.c.

References mpi_sub_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), mpi_gen_prime(), mpi_is_prime(), rsa_check_privkey(), and rsa_gen_key().

13.95.4.35 int mpi_sub_mpi (mpi * X, mpi * A, mpi * B)

Signed subtraction: $X = A - B$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 829 of file bignum.c.

References mpi_add_abs(), MPI_CHK, mpi_cmp_abs(), mpi_sub_abs(), and mpi::s.

Referenced by mpi_div_mpi(), mpi_inv_mod(), mpi_mod_mpi(), mpi_sub_int(), and rsa_private().

13.95.4.36 void mpi_swap (mpi * X, mpi * Y)

Swap the contents of X and Y.

Definition at line 155 of file bignum.c.

Referenced by rsa_gen_key().

13.95.4.37 int mpi_write_binary (mpi * X, unsigned char * buf, int buflen)

Export X into unsigned binary data, big endian.

Parameters:

X source [mpi](#)

buf output buffer

buflen output buffer size

Returns:

0 if successful, XYSSL_ERR_MPI_BUFFER_TOO_SMALL if buf isn't large enough

Note:

Call this function with *buflen = 0 to obtain the minimum required buffer size in *buflen.

Definition at line 506 of file bignum.c.

References ciL, mpi_size(), mpi::p, and XYSSL_ERR_MPI_BUFFER_TOO_SMALL.

Referenced by dhm_calc_secret(), dhm_make_public(), rsa_private(), and rsa_public().

13.95.4.38 int mpi_write_file (char * *p*, mpi * *X*, int *radix*, FILE * *fout*)

Write X into an opened file, or stdout.

Parameters:

p prefix, can be NULL
X source [mpi](#)
radix output numeric base
fout output file handle

Returns:

0 if successful, or an XYSSL_ERR_MPI_XXX error code

Note:

Set fout == NULL to print X on the console.

Definition at line 447 of file bignum.c.

References MPI_CHK, mpi_write_string(), and XYSSL_ERR_MPI_FILE_IO_ERROR.

Referenced by generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), and main().

13.95.4.39 int mpi_write_string (mpi * *X*, int *radix*, char * *s*, int * *slen*)

Export into an ASCII string.

Parameters:

X source [mpi](#)
radix output numeric base
s string buffer
slen string buffer size

Returns:

0 if successful, or an XYSSL_ERR_MPI_XXX error code

Note:

Call this function with *slen = 0 to obtain the minimum required buffer size in *slen.

Definition at line 328 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_copy()`, `mpi_free()`, `mpi_init()`, `mpi_msb()`, `mpi_write_hlp()`, `mpi::n`, `mpi::p`, `mpi::s`, `XYSSL_ERR_MPI_BAD_INPUT_DATA`, and `XYSSL_ERR_MPI_BUFFER_TOO_SMALL`.

Referenced by `mpi_write_file()`.

13.96 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/bn_mul.h File Reference

```
#include "xyssl/config.h"
```

Defines

- #define [MULADDC_INIT](#)
- #define [MULADDC_CORE](#)
- #define [MULADDC_STOP](#) }

13.96.1 Detailed Description

Definition in file [bn_mul.h](#).

13.96.2 Define Documentation

13.96.2.1 #define MULADDC_CORE

Value:

```
s0 = ( *s << biH ) >> biH;          \
s1 = ( *s >> biH ); s++;             \
rx = s0 * b1; r0 = s0 * b0;          \
ry = s1 * b0; r1 = s1 * b1;          \
r1 += ( rx >> biH );                  \
r1 += ( ry >> biH );                  \
rx <=<= biH; ry <=<= biH;              \
r0 += rx; r1 += (r0 < rx);            \
r0 += ry; r1 += (r0 < ry);            \
r0 += c; r1 += (r0 < c);              \
r0 += *d; r1 += (r0 < *d);            \
c = r1; *(d++) = r0;
```

Definition at line 662 of file [bn_mul.h](#).

Referenced by [mpi_mul_hlp\(\)](#).

13.96.2.2 #define MULADDC_INIT

Value:

```
{                                     \
    t_int s0, s1, b0, b1;           \
    t_int r0, r1, rx, ry;           \
    b0 = ( b << biH ) >> biH;       \
    b1 = ( b >> biH );
```

Definition at line 655 of file [bn_mul.h](#).

Referenced by [mpi_mul_hlp\(\)](#).

13.96.2.3 `#define MULADDC_STOP` }

Definition at line 676 of file bn_mul.h.

Referenced by mpi_mul_hlp().

13.97 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/certs.h File Reference

Variables

- char [test_ca_cert](#) []
- char [test_ca_key](#) []
- char [test_ca_pwd](#) []
- char [test_srv_cert](#) []
- char [test_srv_key](#) []
- char [test_cli_cert](#) []
- char [test_cli_key](#) []
- char [xyssl_ca_cert](#) []

13.97.1 Detailed Description

Definition in file [certs.h](#).

13.97.2 Variable Documentation

13.97.2.1 char test_ca_cert[]

Definition at line 25 of file [certs.c](#).

Referenced by [main\(\)](#), [ssl_test\(\)](#), and [x509_self_test\(\)](#).

13.97.2.2 char test_ca_key[]

Definition at line 49 of file [certs.c](#).

Referenced by [x509_self_test\(\)](#).

13.97.2.3 char test_ca_pwd[]

Definition at line 81 of file [certs.c](#).

Referenced by [x509_self_test\(\)](#).

13.97.2.4 char test_cli_cert[]

Definition at line 134 of file [certs.c](#).

Referenced by [main\(\)](#), and [x509_self_test\(\)](#).

13.97.2.5 char test_cli_key[]

Definition at line 156 of file [certs.c](#).

Referenced by [main\(\)](#).

13.97.2.6 char test_srv_cert[]

Definition at line 83 of file certs.c.

Referenced by main(), and ssl_test().

13.97.2.7 char test_srv_key[]

Definition at line 105 of file certs.c.

Referenced by main(), and ssl_test().

13.97.2.8 char xyssl_ca_cert[]

Definition at line 185 of file certs.c.

Referenced by main().

13.98 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/debug.h File Reference

```
#include "xyssl/config.h"
#include "xyssl/ssl.h"
```

Defines

- #define [SSL_DEBUG_MSG](#)(level, args) debug_print_msg(ssl, level, __FILE__, __LINE__, debug_fmt args);
- #define [SSL_DEBUG_RET](#)(level, text, ret) debug_print_ret(ssl, level, __FILE__, __LINE__, text, ret);
- #define [SSL_DEBUG_BUF](#)(level, text, buf, len) debug_print_buf(ssl, level, __FILE__, __LINE__, text, buf, len);
- #define [SSL_DEBUG_MPI](#)(level, text, X) debug_print_mpi(ssl, level, __FILE__, __LINE__, text, X);
- #define [SSL_DEBUG_CRT](#)(level, text, crt) debug_print_crt(ssl, level, __FILE__, __LINE__, text, crt);

Functions

- char * [debug_fmt](#) (const char *format,...)
- void [debug_print_msg](#) (ssl_context *ssl, int level, char *file, int line, char *text)
- void [debug_print_ret](#) (ssl_context *ssl, int level, char *file, int line, char *text, int ret)
- void [debug_print_buf](#) (ssl_context *ssl, int level, char *file, int line, char *text, unsigned char *buf, int len)
- void [debug_print_mpi](#) (ssl_context *ssl, int level, char *file, int line, char *text, mpi *X)
- void [debug_print_crt](#) (ssl_context *ssl, int level, char *file, int line, char *text, x509_cert *crt)

13.98.1 Detailed Description

Definition in file [debug.h](#).

13.98.2 Define Documentation

13.98.2.1 #define [SSL_DEBUG_BUF](#)(level, text, buf, len) debug_print_buf(ssl, level, __FILE__, __LINE__, text, buf, len);

Definition at line 18 of file [debug.h](#).

Referenced by [ssl_calc_finished\(\)](#), [ssl_calc_verify\(\)](#), [ssl_decrypt_buf\(\)](#), [ssl_derive_keys\(\)](#), [ssl_encrypt_buf\(\)](#), [ssl_parse_client_hello\(\)](#), [ssl_parse_server_hello\(\)](#), [ssl_parse_server_key_exchange\(\)](#), [ssl_read_record\(\)](#), [ssl_write_certificate_request\(\)](#), [ssl_write_client_hello\(\)](#), [ssl_write_record\(\)](#), [ssl_write_server_hello\(\)](#), and [ssl_write_server_key_exchange\(\)](#).

13.98.2.2 `#define SSL_DEBUG_CRT(level, text, crt) debug_print_crt(ssl, level, __FILE__,
__LINE__, text, crt);`

Definition at line 24 of file debug.h.

Referenced by ssl_parse_certificate(), and ssl_write_certificate().

13.98.2.3 `#define SSL_DEBUG_MPI(level, text, X) debug_print_mpi(ssl, level, __FILE__,
__LINE__, text, X);`

Definition at line 21 of file debug.h.

Referenced by ssl_parse_client_key_exchange(), ssl_parse_server_key_exchange(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

13.98.2.4 `#define SSL_DEBUG_MSG(level, args) debug_print_msg(ssl, level, __FILE__,
__LINE__, debug_fmt args);`

Definition at line 12 of file debug.h.

Referenced by ssl_calc_finished(), ssl_calc_verify(), ssl_close_notify(), ssl_decrypt_buf(), ssl_derive_keys(), ssl_encrypt_buf(), ssl_fetch_input(), ssl_flush_output(), ssl_free(), ssl_handshake(), ssl_handshake_client(), ssl_handshake_server(), ssl_init(), ssl_parse_certificate(), ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_change_cipher_spec(), ssl_parse_client_hello(), ssl_parse_client_key_exchange(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), ssl_read(), ssl_read_record(), ssl_write(), ssl_write_certificate(), ssl_write_certificate_request(), ssl_write_certificate_verify(), ssl_write_change_cipher_spec(), ssl_write_client_hello(), ssl_write_client_key_exchange(), ssl_write_finished(), ssl_write_record(), ssl_write_server_hello(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

13.98.2.5 `#define SSL_DEBUG_RET(level, text, ret) debug_print_ret(ssl, level, __FILE__,
__LINE__, text, ret);`

Definition at line 15 of file debug.h.

Referenced by ssl_close_notify(), ssl_fetch_input(), ssl_flush_output(), ssl_parse_certificate(), ssl_parse_certificate_request(), ssl_parse_certificate_verify(), ssl_parse_change_cipher_spec(), ssl_parse_client_hello(), ssl_parse_client_key_exchange(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), ssl_read(), ssl_read_record(), ssl_set_dh_param(), ssl_write(), ssl_write_certificate(), ssl_write_certificate_verify(), ssl_write_change_cipher_spec(), ssl_write_client_hello(), ssl_write_client_key_exchange(), ssl_write_finished(), ssl_write_record(), ssl_write_server_hello_done(), and ssl_write_server_key_exchange().

13.98.3 Function Documentation

13.98.3.1 `char* debug_fmt (const char * format, ...)`

Definition at line 38 of file debug.c.

13.98.3.2 void debug_print_buf (ssl_context * ssl, int level, char * file, int line, char * text, unsigned char * buf, int len)

Definition at line 82 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

13.98.3.3 void debug_print_cert (ssl_context * ssl, int level, char * file, int line, char * text, x509_cert * crt)

Definition at line 169 of file debug.c.

References `debug_print_mpi()`, `rsa_context::E`, `_ssl_context::f_dbg`, `rsa_context::N`, `_x509_cert::next`, `_ssl_context::p_dbg`, `_x509_cert::rsa`, and `x509parse_cert_info()`.

13.98.3.4 void debug_print_mpi (ssl_context * ssl, int level, char * file, int line, char * text, mpi * X)

Definition at line 124 of file debug.c.

References `_ssl_context::f_dbg`, `mpi::n`, `mpi::p`, and `_ssl_context::p_dbg`.

Referenced by `debug_print_cert()`.

13.98.3.5 void debug_print_msg (ssl_context * ssl, int level, char * file, int line, char * text)

Definition at line 52 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

13.98.3.6 void debug_print_ret (ssl_context * ssl, int level, char * file, int line, char * text, int ret)

Definition at line 66 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

13.99 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/des.h File Reference

Data Structures

- struct [des_context](#)
DES context structure.
- struct [des3_context](#)
Triple-DES context structure.

Defines

- #define [DES_ENCRYPT](#) 1
- #define [DES_DECRYPT](#) 0

Functions

- void [des_setkey_enc](#) ([des_context](#) *ctx, unsigned char key[8])
DES key schedule (56-bit, encryption).
- void [des_setkey_dec](#) ([des_context](#) *ctx, unsigned char key[8])
DES key schedule (56-bit, decryption).
- void [des3_set2key_enc](#) ([des3_context](#) *ctx, unsigned char key[16])
Triple-DES key schedule (112-bit, encryption).
- void [des3_set2key_dec](#) ([des3_context](#) *ctx, unsigned char key[16])
Triple-DES key schedule (112-bit, decryption).
- void [des3_set3key_enc](#) ([des3_context](#) *ctx, unsigned char key[24])
Triple-DES key schedule (168-bit, encryption).
- void [des3_set3key_dec](#) ([des3_context](#) *ctx, unsigned char key[24])
Triple-DES key schedule (168-bit, decryption).
- void [des_crypt_ecb](#) ([des_context](#) *ctx, unsigned char input[8], unsigned char output[8])
DES-ECB block encryption/decryption.
- void [des_crypt_cbc](#) ([des_context](#) *ctx, int mode, int length, unsigned char iv[8], unsigned char *input, unsigned char *output)
DES-CBC buffer encryption/decryption.
- void [des3_crypt_ecb](#) ([des3_context](#) *ctx, unsigned char input[8], unsigned char output[8])
3DES-ECB block encryption/decryption
- void [des3_crypt_cbc](#) ([des3_context](#) *ctx, int mode, int length, unsigned char iv[8], unsigned char *input, unsigned char *output)

3DES-CBC buffer encryption/decryption

- [int des_self_test](#) ([int](#) verbose)

13.99.1 Detailed Description

Definition in file [des.h](#).

13.99.2 Define Documentation

13.99.2.1 #define DES_DECRYPT 0

Definition at line 8 of file [des.h](#).

Referenced by [des_self_test\(\)](#), [ssl_decrypt_buf\(\)](#), and [x509_des3_decrypt\(\)](#).

13.99.2.2 #define DES_ENCRYPT 1

Definition at line 7 of file [des.h](#).

Referenced by [des3_crypt_cbc\(\)](#), [des_crypt_cbc\(\)](#), [main\(\)](#), and [ssl_encrypt_buf\(\)](#).

13.99.3 Function Documentation

13.99.3.1 void des3_crypt_cbc (des3_context * *ctx*, int *mode*, int *length*, unsigned char *iv*[8], unsigned char * *input*, unsigned char * *output*)

3DES-CBC buffer encryption/decryption

Parameters:

ctx 3DES context
mode DES_ENCRYPT or DES_DECRYPT
length length of the input data
iv initialization vector (updated after use)
input buffer holding the input data
output buffer holding the output data

Definition at line 593 of file [des.c](#).

References [des3_crypt_ecb\(\)](#), and [DES_ENCRYPT](#).

Referenced by [des_self_test\(\)](#), [main\(\)](#), [ssl_decrypt_buf\(\)](#), [ssl_encrypt_buf\(\)](#), and [x509_des3_decrypt\(\)](#).

13.99.3.2 void des3_crypt_ecb (des3_context * *ctx*, unsigned char *input*[8], unsigned char *output*[8])

3DES-ECB block encryption/decryption

Parameters:

ctx 3DES context

input 64-bit input block

output 64-bit output block

Definition at line 552 of file des.c.

References DES_FP, DES_IP, DES_ROUND, GET_ULONG_BE, PUT_ULONG_BE, and des3_context::sk.

Referenced by des3_crypt_cbc(), and des_self_test().

13.99.3.3 void des3_set2key_dec (des3_context * ctx, unsigned char key[16])

Triple-DES key schedule (112-bit, decryption).

Parameters:

ctx 3DES context to be initialized

key 16-byte secret key

Definition at line 420 of file des.c.

References des3_set2key(), and des3_context::sk.

Referenced by des_self_test().

13.99.3.4 void des3_set2key_enc (des3_context * ctx, unsigned char key[16])

Triple-DES key schedule (112-bit, encryption).

Parameters:

ctx 3DES context to be initialized

key 16-byte secret key

Definition at line 409 of file des.c.

References des3_set2key(), and des3_context::sk.

Referenced by des_self_test().

13.99.3.5 void des3_set3key_dec (des3_context * ctx, unsigned char key[24])

Triple-DES key schedule (168-bit, decryption).

Parameters:

ctx 3DES context to be initialized

key 24-byte secret key

Definition at line 465 of file des.c.

References des3_set3key(), and des3_context::sk.

Referenced by des_self_test(), ssl_derive_keys(), and x509_des3_decrypt().

13.99.3.6 void des3_set3key_enc (des3_context * ctx, unsigned char key[24])

Triple-DES key schedule (168-bit, encryption).

Parameters:

ctx 3DES context to be initialized

key 24-byte secret key

Definition at line 454 of file des.c.

References des3_set3key(), and des3_context::sk.

Referenced by des_self_test(), main(), and ssl_derive_keys().

13.99.3.7 void des_crypt_cbc (des_context * ctx, int mode, int length, unsigned char iv[8], unsigned char * input, unsigned char * output)

DES-CBC buffer encryption/decryption.

Parameters:

ctx DES context

mode DES_ENCRYPT or DES_DECRYPT

length length of the input data

iv initialization vector (updated after use)

input buffer holding the input data

output buffer holding the output data

Definition at line 505 of file des.c.

References des_crypt_ecb(), and DES_ENCRYPT.

Referenced by des_self_test(), and main().

13.99.3.8 void des_crypt_ecb (des_context * ctx, unsigned char input[8], unsigned char output[8])

DES-ECB block encryption/decryption.

Parameters:

ctx DES context

input 64-bit input block

output 64-bit output block

Definition at line 476 of file des.c.

References DES_FP, DES_IP, DES_ROUND, GET_ULONG_BE, PUT_ULONG_BE, and des_context::sk.

Referenced by des_crypt_cbc(), and des_self_test().

13.99.3.9 int des_self_test (int *verbose*)

Definition at line 694 of file des.c.

References `buf`, `des3_crypt_cbc()`, `des3_crypt_ecb()`, `des3_set2key_dec()`, `des3_set2key_enc()`, `des3_set3key_dec()`, `des3_set3key_enc()`, `des3_test_buf`, `des3_test_cbc_dec`, `des3_test_cbc_enc`, `des3_test_ecb_dec`, `des3_test_ecb_enc`, `des3_test_iv`, `des3_test_keys`, `des_crypt_cbc()`, `des_crypt_ecb()`, `DES_DECRYPT`, `des_setkey_dec()`, `des_setkey_enc()`, and `prv`.

Referenced by `main()`.

13.99.3.10 void des_setkey_dec (des_context * *ctx*, unsigned char *key*[8])

DES key schedule (56-bit, decryption).

Parameters:

ctx DES context to be initialized

key 8-byte secret key

Definition at line 368 of file des.c.

References `des_setkey()`, `des_context::sk`, and `SWAP`.

Referenced by `des_self_test()`.

13.99.3.11 void des_setkey_enc (des_context * *ctx*, unsigned char *key*[8])

DES key schedule (56-bit, encryption).

Parameters:

ctx DES context to be initialized

key 8-byte secret key

Definition at line 360 of file des.c.

References `des_setkey()`, and `des_context::sk`.

Referenced by `des_self_test()`, and `main()`.

13.100 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/dhm.h File Reference

```
#include "bignum.h"
```

Data Structures

- struct [dhm_context](#)

Defines

- #define [XYSSL_ERR_DHM_BAD_INPUT_DATA](#) -0x0480
- #define [XYSSL_ERR_DHM_READ_PARAMS_FAILED](#) -0x0490
- #define [XYSSL_ERR_DHM_MAKE_PARAMS_FAILED](#) -0x04A0
- #define [XYSSL_ERR_DHM_READ_PUBLIC_FAILED](#) -0x04B0
- #define [XYSSL_ERR_DHM_MAKE_PUBLIC_FAILED](#) -0x04C0
- #define [XYSSL_ERR_DHM_CALC_SECRET_FAILED](#) -0x04D0

Functions

- [int dhm_read_params](#) ([dhm_context](#) *ctx, unsigned char **p, unsigned char *end)
Parse the ServerKeyExchange parameters.
- [int dhm_make_params](#) ([dhm_context](#) *ctx, [int](#) s_size, unsigned char *output, [int](#) *olen, [int](#)(*f_rng)(void *), void *p_rng)
Setup and write the ServerKeyExchange parameters.
- [int dhm_read_public](#) ([dhm_context](#) *ctx, unsigned char *input, [int](#) ilen)
Import the peer's public value G^Y .
- [int dhm_make_public](#) ([dhm_context](#) *ctx, [int](#) s_size, unsigned char *output, [int](#) olen, [int](#)(*f_rng)(void *), void *p_rng)
Create own private value X and export G^X .
- [int dhm_calc_secret](#) ([dhm_context](#) *ctx, unsigned char *output, [int](#) *olen)
Derive and export the shared secret $(G^Y)^X \bmod P$.
- void [dhm_free](#) ([dhm_context](#) *ctx)
- [int dhm_self_test](#) ([int](#) verbose)
Checkup routine.

13.100.1 Detailed Description

Definition in file [dhm.h](#).

13.100.2 Define Documentation

13.100.2.1 `#define XYSSL_ERR_DHM_BAD_INPUT_DATA -0x0480`

Definition at line 9 of file dhm.h.

Referenced by `dhm_calc_secret()`, `dhm_make_public()`, `dhm_read_bignum()`, `dhm_read_params()`, and `dhm_read_public()`.

13.100.2.2 `#define XYSSL_ERR_DHM_CALC_SECRET_FAILED -0x04D0`

Definition at line 14 of file dhm.h.

Referenced by `dhm_calc_secret()`.

13.100.2.3 `#define XYSSL_ERR_DHM_MAKE_PARAMS_FAILED -0x04A0`

Definition at line 11 of file dhm.h.

Referenced by `dhm_make_params()`.

13.100.2.4 `#define XYSSL_ERR_DHM_MAKE_PUBLIC_FAILED -0x04C0`

Definition at line 13 of file dhm.h.

Referenced by `dhm_make_public()`.

13.100.2.5 `#define XYSSL_ERR_DHM_READ_PARAMS_FAILED -0x0490`

Definition at line 10 of file dhm.h.

Referenced by `dhm_read_bignum()`.

13.100.2.6 `#define XYSSL_ERR_DHM_READ_PUBLIC_FAILED -0x04B0`

Definition at line 12 of file dhm.h.

Referenced by `dhm_read_public()`.

13.100.3 Function Documentation

13.100.3.1 `int dhm_calc_secret (dhm_context * ctx, unsigned char * output, int * olen)`

Derive and export the shared secret $(G^Y)^X \bmod P$.

Parameters:

ctx DHM context

output destination buffer

olen number of chars written

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 208 of file dhm.c.

References dhm_context::GY, dhm_context::K, MPI_CHK, mpi_exp_mod(), mpi_size(), mpi_write_binary(), dhm_context::P, dhm_context::RP, dhm_context::X, XYSSL_ERR_DHM_BAD_INPUT_DATA, and XYSSL_ERR_DHM_CALC_SECRET_FAILED.

Referenced by main(), ssl_parse_client_key_exchange(), and ssl_write_client_key_exchange().

13.100.3.2 void dhm_free (dhm_context * ctx)

Definition at line 234 of file dhm.c.

References dhm_context::G, dhm_context::GX, dhm_context::GY, dhm_context::K, mpi_free(), dhm_context::P, dhm_context::RP, and dhm_context::X.

Referenced by main(), and ssl_free().

13.100.3.3 int dhm_make_params (dhm_context * ctx, int s_size, unsigned char * output, int * olen, int(*) (void *) f_rng, void * p_rng)

Setup and write the ServerKeyExchange parameters.

Parameters:

ctx DHM context
x_size private value size in bits
output destination buffer
olen number of chars written
f_rng RNG function
p_rng RNG parameter

Note:

This function assumes that ctx->P and ctx->G have already been properly set (for example using mpi_read_string or mpi_read_binary).

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 93 of file dhm.c.

References DHM_MPI_EXPORT, dhm_context::G, dhm_context::GX, dhm_context::len, MPI_CHK, mpi_cmp_mpi(), mpi_exp_mod(), mpi_grow(), mpi_lset(), mpi_shift_r(), mpi_size(), dhm_context::P, mpi::p, dhm_context::RP, dhm_context::X, and XYSSL_ERR_DHM_MAKE_PARAMS_FAILED.

Referenced by main(), and ssl_write_server_key_exchange().

13.100.3.4 int dhm_make_public (dhm_context * ctx, int s_size, unsigned char * output, int olen, int(*) (void *) f_rng, void * p_rng)

Create own private value X and export G^X .

Parameters:

ctx DHM context
x_size private value size in bits
output destination buffer
olen must be equal to `ctx->P.len`
f_rng RNG function
p_rng RNG parameter

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 167 of file dhm.c.

References `dhm_context::G`, `dhm_context::GX`, `dhm_context::len`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_grow()`, `mpi_lset()`, `mpi_shift_r()`, `mpi_write_binary()`, `dhm_context::P`, `mpi::p`, `dhm_context::RP`, `dhm_context::X`, `XYSSL_ERR_DHM_BAD_INPUT_DATA`, and `XYSSL_ERR_DHM_MAKE_PUBLIC_FAILED`.

Referenced by `main()`, and `ssl_write_client_key_exchange()`.

13.100.3.5 `int dhm_read_params (dhm_context * ctx, unsigned char ** p, unsigned char * end)`

Parse the ServerKeyExchange parameters.

Parameters:

ctx DHM context
p &(start of input buffer)
end end of buffer

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 63 of file dhm.c.

References `dhm_read_bignum()`, `dhm_context::G`, `dhm_context::GY`, `dhm_context::len`, `mpi_size()`, `dhm_context::P`, and `XYSSL_ERR_DHM_BAD_INPUT_DATA`.

Referenced by `main()`, and `ssl_parse_server_key_exchange()`.

13.100.3.6 `int dhm_read_public (dhm_context * ctx, unsigned char * input, int ilen)`

Import the peer's public value G^Y .

Parameters:

ctx DHM context
input input buffer
ilen size of buffer

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 150 of file dhm.c.

References dhm_context::GY, dhm_context::len, mpi_read_binary(), XYSSL_ERR_DHM_BAD_INPUT_DATA, and XYSSL_ERR_DHM_READ_PUBLIC_FAILED.

Referenced by main(), and ssl_parse_client_key_exchange().

13.100.3.7 int dhm_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 246 of file dhm.c.

13.101 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/havege.h File Reference

Data Structures

- struct [havege_state](#)
HAVEGE state structure.

Defines

- #define [COLLECT_SIZE](#) 1024

Functions

- void [havege_init](#) ([havege_state](#) *hs)
HAVEGE initialization.
- int [havege_rand](#) (void *p_rng)
HAVEGE rand function.

13.101.1 Detailed Description

Definition in file [havege.h](#).

13.101.2 Define Documentation

13.101.2.1 #define COLLECT_SIZE 1024

Definition at line 7 of file [havege.h](#).

Referenced by [havege_fill\(\)](#), and [havege_rand\(\)](#).

13.101.3 Function Documentation

13.101.3.1 void havege_init (havege_state * hs)

HAVEGE initialization.

Parameters:

hs HAVEGE state to be initialized

Definition at line 188 of file [havege.c](#).

References [havege_fill\(\)](#).

Referenced by [generate_AES_key\(\)](#), [generate_RSA_keys_ciphertext\(\)](#), [generate_RSA_keys_plaintext\(\)](#), [initiate_migration_process\(\)](#), [main\(\)](#), and [ssl_test\(\)](#).

13.101.3.2 int havege_rand (void * *p_rng*)

HAVEGE rand function.

Parameters:

rng_st points to an HAVEGE state

Returns:

A random int

Definition at line 198 of file havege.c.

References COLLECT_SIZE, havege_fill(), havege_state::offset, and havege_state::pool.

Referenced by generate_AES_key(), generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), initiate_migration_process(), main(), and ssl_test().

13.102 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md2.h File Reference

Data Structures

- struct [md2_context](#)
MD2 context structure.

Functions

- void [md2_starts](#) ([md2_context](#) *ctx)
MD2 context setup.
- void [md2_update](#) ([md2_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD2 process buffer.
- void [md2_finish](#) ([md2_context](#) *ctx, unsigned char output[16])
MD2 final digest.
- void [md2](#) (unsigned char *input, [int](#) ilen, unsigned char output[16])
Output = MD2(input buffer).
- [int](#) [md2_file](#) (char *path, unsigned char output[16])
Output = MD2(file contents).
- void [md2_hmac_starts](#) ([md2_context](#) *ctx, unsigned char *key, [int](#) keylen)
MD2 HMAC context setup.
- void [md2_hmac_update](#) ([md2_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD2 HMAC process buffer.
- void [md2_hmac_finish](#) ([md2_context](#) *ctx, unsigned char output[16])
MD2 HMAC final digest.
- void [md2_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char output[16])
Output = HMAC-MD2(hmac key, input buffer).
- [int](#) [md2_self_test](#) ([int](#) verbose)
Checkup routine.

13.102.1 Detailed Description

Definition in file [md2.h](#).

13.102.2 Function Documentation

13.102.2.1 void md2 (unsigned char * *input*, int *ilen*, unsigned char *output*[16])

Output = MD2(input buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output MD2 checksum result

Referenced by x509_hash().

13.102.2.2 int md2_file (char * *path*, unsigned char *output*[16])

Output = MD2(file contents).

Parameters:

path input file name
output MD2 checksum result

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

13.102.2.3 void md2_finish (md2_context * *ctx*, unsigned char *output*[16])

MD2 final digest.

Parameters:

ctx MD2 context
output MD2 checksum result

13.102.2.4 void md2_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char *output*[16])

Output = HMAC-MD2(hmac key, input buffer).

Parameters:

key HMAC secret key
keylen length of the HMAC key
input buffer holding the data
ilen length of the input data
output HMAC-MD2 result

13.102.2.5 void md2_hmac_finish (md2_context * *ctx*, unsigned char *output*[16])

MD2 HMAC final digest.

Parameters:

ctx HMAC context

output MD2 HMAC checksum result

13.102.2.6 void md2_hmac_starts (md2_context * *ctx*, unsigned char * *key*, int *keylen*)

MD2 HMAC context setup.

Parameters:

ctx HMAC context to be initialized

key HMAC secret key

keylen length of the HMAC key

13.102.2.7 void md2_hmac_update (md2_context * *ctx*, unsigned char * *input*, int *ilen*)

MD2 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

13.102.2.8 int md2_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Referenced by main().

13.102.2.9 void md2_starts (md2_context * *ctx*)

MD2 context setup.

Parameters:

ctx context to be initialized

13.102.2.10 void md2_update (md2_context * *ctx*, unsigned char * *input*, int *ilen*)

MD2 process buffer.

Parameters:

ctx MD2 context

input buffer holding the data

ilen length of the input data

13.103 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md4.h File Reference

Data Structures

- struct [md4_context](#)
MD4 context structure.

Functions

- void [md4_starts](#) ([md4_context](#) *ctx)
MD4 context setup.
- void [md4_update](#) ([md4_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD4 process buffer.
- void [md4_finish](#) ([md4_context](#) *ctx, unsigned char output[16])
MD4 final digest.
- void [md4](#) (unsigned char *input, [int](#) ilen, unsigned char output[16])
Output = MD4(input buffer).
- [int](#) [md4_file](#) (char *path, unsigned char output[16])
Output = MD4(file contents).
- void [md4_hmac_starts](#) ([md4_context](#) *ctx, unsigned char *key, [int](#) keylen)
MD4 HMAC context setup.
- void [md4_hmac_update](#) ([md4_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD4 HMAC process buffer.
- void [md4_hmac_finish](#) ([md4_context](#) *ctx, unsigned char output[16])
MD4 HMAC final digest.
- void [md4_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char output[16])
Output = HMAC-MD4(hmac key, input buffer).
- [int](#) [md4_self_test](#) ([int](#) verbose)
Checkup routine.

13.103.1 Detailed Description

Definition in file [md4.h](#).

13.103.2 Function Documentation

13.103.2.1 void md4 (unsigned char * *input*, int *ilen*, unsigned char *output*[16])

Output = MD4(input buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output MD4 checksum result

Referenced by main(), and x509_hash().

13.103.2.2 int md4_file (char * *path*, unsigned char *output*[16])

Output = MD4(file contents).

Parameters:

path input file name
output MD4 checksum result

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

13.103.2.3 void md4_finish (md4_context * *ctx*, unsigned char *output*[16])

MD4 final digest.

Parameters:

ctx MD4 context
output MD4 checksum result

13.103.2.4 void md4_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char *output*[16])

Output = HMAC-MD4(hmac key, input buffer).

Parameters:

key HMAC secret key
keylen length of the HMAC key
input buffer holding the data
ilen length of the input data
output HMAC-MD4 result

13.103.2.5 void md4_hmac_finish (md4_context * *ctx*, unsigned char *output*[16])

MD4 HMAC final digest.

Parameters:

ctx HMAC context

output MD4 HMAC checksum result

13.103.2.6 void md4_hmac_starts (md4_context * *ctx*, unsigned char * *key*, int *keylen*)

MD4 HMAC context setup.

Parameters:

ctx HMAC context to be initialized

key HMAC secret key

keylen length of the HMAC key

13.103.2.7 void md4_hmac_update (md4_context * *ctx*, unsigned char * *input*, int *ilen*)

MD4 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

13.103.2.8 int md4_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Referenced by main().

13.103.2.9 void md4_starts (md4_context * *ctx*)

MD4 context setup.

Parameters:

ctx context to be initialized

13.103.2.10 void md4_update (md4_context * *ctx*, unsigned char * *input*, int *ilen*)

MD4 process buffer.

Parameters:

ctx MD4 context

input buffer holding the data

ilen length of the input data

13.104 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/md5.h File Reference

Data Structures

- struct [md5_context](#)
MD5 context structure.

Functions

- void [md5_starts](#) ([md5_context](#) *ctx)
MD5 context setup.
- void [md5_update](#) ([md5_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD5 process buffer.
- void [md5_finish](#) ([md5_context](#) *ctx, unsigned char output[16])
MD5 final digest.
- void [md5](#) (unsigned char *input, [int](#) ilen, unsigned char output[16])
Output = MD5(input buffer).
- [int](#) [md5_file](#) (char *path, unsigned char output[16])
Output = MD5(file contents).
- void [md5_hmac_starts](#) ([md5_context](#) *ctx, unsigned char *key, [int](#) keylen)
MD5 HMAC context setup.
- void [md5_hmac_update](#) ([md5_context](#) *ctx, unsigned char *input, [int](#) ilen)
MD5 HMAC process buffer.
- void [md5_hmac_finish](#) ([md5_context](#) *ctx, unsigned char output[16])
MD5 HMAC final digest.
- void [md5_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char output[16])
Output = HMAC-MD5(hmac key, input buffer).
- [int](#) [md5_self_test](#) ([int](#) verbose)
Checkup routine.

13.104.1 Detailed Description

Definition in file [md5.h](#).

13.104.2 Function Documentation

13.104.2.1 void md5 (unsigned char * *input*, int *ilen*, unsigned char *output*[16])

Output = MD5(*input* buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output MD5 checksum result

Definition at line 278 of file md5.c.

References md5_finish(), md5_starts(), and md5_update().

Referenced by initiate_migration_process(), main(), md5_hmac_starts(), md5_self_test(), reply_migration_process(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_finished(), ssl_parse_server_key_exchange(), ssl_write_finished(), ssl_write_server_key_exchange(), and x509_hash().

13.104.2.2 int md5_file (char * *path*, unsigned char *output*[16])

Output = MD5(*file* contents).

Parameters:

path input file name
output MD5 checksum result

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 292 of file md5.c.

References buf, f, md5_finish(), md5_starts(), and md5_update().

Referenced by md5_wrapper().

13.104.2.3 void md5_finish (md5_context * *ctx*, unsigned char *output*[16])

MD5 final digest.

Parameters:

ctx MD5 context
output MD5 checksum result

Definition at line 250 of file md5.c.

References md5_padding, md5_update(), PUT_ULONG_LE, md5_context::state, and md5_context::total.

Referenced by md5(), md5_file(), md5_hmac_finish(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

13.104.2.4 `void md5_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char output[16])`

Output = HMAC-MD5(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key

input buffer holding the data

ilen length of the input data

output HMAC-MD5 result

Definition at line 378 of file md5.c.

References md5_hmac_finish(), md5_hmac_starts(), and md5_hmac_update().

Referenced by ssl_decrypt_buf(), ssl_encrypt_buf(), and tls1_prf().

13.104.2.5 `void md5_hmac_finish (md5_context * ctx, unsigned char output[16])`

MD5 HMAC final digest.

Parameters:

ctx HMAC context

output MD5 HMAC checksum result

Definition at line 362 of file md5.c.

References md5_finish(), md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac(), and md5_self_test().

13.104.2.6 `void md5_hmac_starts (md5_context * ctx, unsigned char * key, int keylen)`

MD5 HMAC context setup.

Parameters:

ctx HMAC context to be initialized

key HMAC secret key

keylen length of the HMAC key

Definition at line 324 of file md5.c.

References md5_context::ipad, md5(), md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac(), and md5_self_test().

13.104.2.7 `void md5_hmac_update (md5_context * ctx, unsigned char * input, int ilen)`

MD5 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

Definition at line 354 of file md5.c.

References md5_update().

Referenced by md5_hmac(), and md5_self_test().

13.104.2.8 int md5_self_test (int verbose)

Checksum routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 495 of file md5.c.

References buf, md5(), md5_hmac_finish(), md5_hmac_starts(), md5_hmac_test_buf, md5_hmac_test_buflen, md5_hmac_test_key, md5_hmac_test_keylen, md5_hmac_test_sum, md5_hmac_update(), md5_test_buf, md5_test_buflen, and md5_test_sum.

Referenced by main().

13.104.2.9 void md5_starts (md5_context * ctx)

MD5 context setup.

Parameters:

ctx context to be initialized

Definition at line 61 of file md5.c.

References md5_context::state, and md5_context::total.

Referenced by md5(), md5_file(), md5_hmac_finish(), md5_hmac_starts(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_init(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

13.104.2.10 void md5_update (md5_context * ctx, unsigned char * input, int ilen)

MD5 process buffer.

Parameters:

ctx MD5 context

input buffer holding the data

ilen length of the input data

Definition at line 198 of file md5.c.

References md5_context::buffer, md5_process(), and md5_context::total.

Referenced by md5(), md5_file(), md5_finish(), md5_hmac_finish(), md5_hmac_starts(), md5_hmac_update(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_read_record(), ssl_write_record(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

13.105 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/net.h File Reference

Defines

- `#define XYSSL_ERR_NET_UNKNOWN_HOST -0x0F00`
- `#define XYSSL_ERR_NET_SOCKET_FAILED -0x0F10`
- `#define XYSSL_ERR_NET_CONNECT_FAILED -0x0F20`
- `#define XYSSL_ERR_NET_BIND_FAILED -0x0F30`
- `#define XYSSL_ERR_NET_LISTEN_FAILED -0x0F40`
- `#define XYSSL_ERR_NET_ACCEPT_FAILED -0x0F50`
- `#define XYSSL_ERR_NET_RECV_FAILED -0x0F60`
- `#define XYSSL_ERR_NET_SEND_FAILED -0x0F70`
- `#define XYSSL_ERR_NET_CONN_RESET -0x0F80`
- `#define XYSSL_ERR_NET_TRY_AGAIN -0x0F90`

Functions

- `int net_connect (int *fd, char *host, int port)`
Initiate a TCP connection with host:port.
- `int net_bind (int *fd, char *bind_ip, int port)`
Create a listening socket on bind_ip:port. If bind_ip == NULL, all interfaces are binded.
- `int net_accept (int bind_fd, int *client_fd, void *client_ip)`
Accept a connection from a remote client.
- `int net_set_block (int fd)`
Set the socket blocking.
- `int net_set_nonblock (int fd)`
Set the socket non-blocking.
- `void net_usleep (unsigned long usec)`
Portable usleep helper.
- `int net_recv (void *ctx, unsigned char *buf, int len)`
Read at most 'len' characters. len is updated to reflect the actual number of characters read.
- `int net_send (void *ctx, unsigned char *buf, int len)`
Write at most 'len' characters. len is updated to reflect the number of characters _not_ written.
- `void net_close (int fd)`
Gracefully shutdown the connection.

13.105.1 Detailed Description

Definition in file [net.h](#).

13.105.2 Define Documentation

13.105.2.1 #define XYSSL_ERR_NET_ACCEPT_FAILED -0x0F50

Definition at line 12 of file net.h.

Referenced by net_accept().

13.105.2.2 #define XYSSL_ERR_NET_BIND_FAILED -0x0F30

Definition at line 10 of file net.h.

Referenced by net_bind().

13.105.2.3 #define XYSSL_ERR_NET_CONN_RESET -0x0F80

Definition at line 15 of file net.h.

Referenced by main(), net_recv(), net_send(), and ssl_test().

13.105.2.4 #define XYSSL_ERR_NET_CONNECT_FAILED -0x0F20

Definition at line 9 of file net.h.

Referenced by net_connect().

13.105.2.5 #define XYSSL_ERR_NET_LISTEN_FAILED -0x0F40

Definition at line 11 of file net.h.

Referenced by net_bind().

13.105.2.6 #define XYSSL_ERR_NET_RECV_FAILED -0x0F60

Definition at line 13 of file net.h.

Referenced by net_recv().

13.105.2.7 #define XYSSL_ERR_NET_SEND_FAILED -0x0F70

Definition at line 14 of file net.h.

Referenced by net_send().

13.105.2.8 #define XYSSL_ERR_NET_SOCKET_FAILED -0x0F10

Definition at line 8 of file net.h.

Referenced by net_bind(), and net_connect().

13.105.2.9 `#define XYSSL_ERR_NET_TRY_AGAIN -0x0F90`

Definition at line 16 of file net.h.

Referenced by `main()`, `net_accept()`, `net_recv()`, `net_send()`, and `ssl_test()`.

13.105.2.10 `#define XYSSL_ERR_NET_UNKNOWN_HOST -0x0F00`

Definition at line 7 of file net.h.

Referenced by `net_connect()`.

13.105.3 Function Documentation

13.105.3.1 `int net_accept (int bind_fd, int * client_fd, void * client_ip)`

Accept a connection from a remote client.

Returns:

0 if successful, `XYSSL_ERR_NET_ACCEPT_FAILED`, or `XYSSL_ERR_NET_WOULD_BLOCK` if `bind_fd` was set to non-blocking and `accept()` is blocking.

Definition at line 212 of file net.c.

References `int`, `net_is_blocking()`, `XYSSL_ERR_NET_ACCEPT_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

13.105.3.2 `int net_bind (int * fd, char * bind_ip, int port)`

Create a listening socket on `bind_ip:port`. If `bind_ip == NULL`, all interfaces are binded.

Returns:

0 if successful, or one of: `XYSSL_ERR_NET_SOCKET_FAILED`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`

Definition at line 126 of file net.c.

References `net_htons()`, `SOCKET_ERROR`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`, and `XYSSL_ERR_NET_SOCKET_FAILED`.

Referenced by `main()`, and `ssl_test()`.

13.105.3.3 `void net_close (int fd)`

Gracefully shutdown the connection.

Definition at line 338 of file net.c.

Referenced by `main()`, and `ssl_test()`.

13.105.3.4 `int net_connect (int *fd, char *host, int port)`

Initiate a TCP connection with host:port.

Returns:

0 if successful, or one of: XYSSL_ERR_NET_SOCKET_FAILED, XYSSL_ERR_NET_UNKNOWN_HOST, XYSSL_ERR_NET_CONNECT_FAILED

Definition at line 81 of file net.c.

References `net_htons()`, `SOCKET_ERROR`, `XYSSL_ERR_NET_CONNECT_FAILED`, `XYSSL_ERR_NET_SOCKET_FAILED`, and `XYSSL_ERR_NET_UNKNOWN_HOST`.

Referenced by `main()`, and `ssl_test()`.

13.105.3.5 `int net_recv (void *ctx, unsigned char *buf, int len)`

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

Returns:

This function returns the number of bytes received, or a negative error code; XYSSL_ERR_NET_TRY_AGAIN indicates `read()` is blocking.

Definition at line 277 of file net.c.

References `net_is_blocking()`, `XYSSL_ERR_NET_CONN_RESET`, `XYSSL_ERR_NET_RECV_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

13.105.3.6 `int net_send (void *ctx, unsigned char *buf, int len)`

Write at most 'len' characters. len is updated to reflect the number of characters `_not_` written.

Returns:

This function returns the number of bytes sent, or a negative error code; XYSSL_ERR_NET_TRY_AGAIN indicates `write()` is blocking.

Definition at line 309 of file net.c.

References `net_is_blocking()`, `XYSSL_ERR_NET_CONN_RESET`, `XYSSL_ERR_NET_SEND_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

13.105.3.7 `int net_set_block (int fd)`

Set the socket blocking.

Returns:

0 if successful, or a non-zero error code

Definition at line 243 of file net.c.

13.105.3.8 int net_set_nonblock (int *fd*)

Set the socket non-blocking.

Returns:

0 if successful, or a non-zero error code

Definition at line 253 of file net.c.

Referenced by ssl_test().

13.105.3.9 void net_usleep (unsigned long *usec*)

Portable usleep helper.

Note:

Real amount of time slept will not be less than select()'s timeout granularity (typically, 10ms).

Definition at line 266 of file net.c.

13.106 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/openssl.h File Reference

```
#include "xyssl/aes.h"
#include "xyssl/md5.h"
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

Defines

- #define [AES_SIZE](#) 16
- #define [AES_BLOCK_SIZE](#) 16
- #define [AES_KEY](#) aes_context
- #define [MD5_CTX](#) md5_context
- #define [SHA_CTX](#) sha1_context
- #define [SHA1_Init](#)(CTX) sha1_starts((CTX))
- #define [SHA1_Update](#)(CTX, BUF, LEN) sha1_update((CTX), (unsigned char *) (BUF), (LEN))
- #define [SHA1_Final](#)(OUT, CTX) sha1_finish((CTX), (OUT))
- #define [MD5_Init](#)(CTX) md5_starts((CTX))
- #define [MD5_Update](#)(CTX, BUF, LEN) md5_update((CTX), (unsigned char *) (BUF), (LEN))
- #define [MD5_Final](#)(OUT, CTX) md5_finish((CTX), (OUT))
- #define [AES_set_encrypt_key](#)(KEY, KEYSIZE, CTX) aes_setkey_enc((CTX), (KEY), (KEYSIZE))
- #define [AES_set_decrypt_key](#)(KEY, KEYSIZE, CTX) aes_setkey_dec((CTX), (KEY), (KEYSIZE))
- #define [AES_cbc_encrypt](#)(INPUT, OUTPUT, LEN, CTX, IV, MODE) aes_crypt_cbc((CTX), (MODE), (LEN), (IV), (INPUT), (OUTPUT))
- #define [RSA](#) rsa_context
- #define [RSA_PKCS1_PADDING](#) 1
- #define [RSA_size](#)(CTX) (CTX)->len
- #define [RSA_free](#)(CTX) rsa_free(CTX)
- #define [ERR_get_error](#)() "ERR_get_error() not supported"
- #define [RSA_blinding_off](#)(IGNORE)
- #define [d2i_RSAPrivateKey](#)(a, b, c) new rsa_context

Functions

- [int __RSA_Passthrough](#) (void *output, void *input, [int size](#))
- [rsa_context * d2i_RSA_PUBKEY](#) (void *ignore, unsigned char **bufptr, [int len](#))
- [int RSA_public_decrypt](#) ([int size](#), unsigned char *input, unsigned char *output, RSA *key, [int ignore](#))
- [int RSA_private_decrypt](#) ([int size](#), unsigned char *input, unsigned char *output, RSA *key, [int ignore](#))
- [int RSA_public_encrypt](#) ([int size](#), unsigned char *input, unsigned char *output, RSA *key, [int ignore](#))
- [int RSA_private_encrypt](#) ([int size](#), unsigned char *input, unsigned char *output, RSA *key, [int ignore](#))

13.106.1 Detailed Description

Definition in file [openssl.h](#).

13.106.2 Define Documentation

13.106.2.1 `#define AES_BLOCK_SIZE 16`

Definition at line 16 of file `openssl.h`.

13.106.2.2 `#define AES_cbc_encrypt(INPUT, OUTPUT, LEN, CTX, IV, MODE) aes_crypt_cbc((CTX), (MODE), (LEN), (IV), (INPUT), (OUTPUT))`

Definition at line 39 of file `openssl.h`.

13.106.2.3 `#define AES_KEY aes_context`

Definition at line 17 of file `openssl.h`.

13.106.2.4 `#define AES_set_decrypt_key(KEY, KEYSIZE, CTX) aes_setkey_dec((CTX), (KEY), (KEYSIZE))`

Definition at line 37 of file `openssl.h`.

13.106.2.5 `#define AES_set_encrypt_key(KEY, KEYSIZE, CTX) aes_setkey_enc((CTX), (KEY), (KEYSIZE))`

Definition at line 35 of file `openssl.h`.

13.106.2.6 `#define AES_SIZE 16`

Definition at line 15 of file `openssl.h`.

13.106.2.7 `#define d2i_RSAPrivateKey(a, b, c) new rsa_context`

Definition at line 102 of file `openssl.h`.

13.106.2.8 `#define ERR_get_error() "ERR_get_error() not supported"`

Definition at line 99 of file `openssl.h`.

13.106.2.9 `#define MD5_CTX md5_context`

Definition at line 18 of file `openssl.h`.

13.106.2.10 #define MD5_Final(OUT, CTX) md5_finish((CTX), (OUT))

Definition at line 32 of file openssl.h.

13.106.2.11 #define MD5_Init(CTX) md5_starts((CTX))

Definition at line 28 of file openssl.h.

13.106.2.12 #define MD5_Update(CTX, BUF, LEN) md5_update((CTX), (unsigned char *) (BUF), (LEN))

Definition at line 30 of file openssl.h.

13.106.2.13 #define RSA_rsa_context

Definition at line 95 of file openssl.h.

13.106.2.14 #define RSA_blinding_off(IGNORE)

Definition at line 100 of file openssl.h.

13.106.2.15 #define RSA_free(CTX) rsa_free(CTX)

Definition at line 98 of file openssl.h.

13.106.2.16 #define RSA_PKCS1_PADDING 1

Definition at line 96 of file openssl.h.

13.106.2.17 #define RSA_size(CTX) (CTX)->len

Definition at line 97 of file openssl.h.

Referenced by RSA_private_encrypt(), and RSA_public_encrypt().

13.106.2.18 #define SHA1_Final(OUT, CTX) sha1_finish((CTX), (OUT))

Definition at line 25 of file openssl.h.

13.106.2.19 #define SHA1_Init(CTX) sha1_starts((CTX))

Definition at line 21 of file openssl.h.

13.106.2.20 #define SHA1_Update(CTX, BUF, LEN) sha1_update((CTX), (unsigned char *) (BUF), (LEN))

Definition at line 23 of file openssl.h.

13.106.2.21 #define SHA_CTX sha1_context

Definition at line 19 of file openssl.h.

13.106.3 Function Documentation**13.106.3.1 int __RSA_Passthrough (void * *output*, void * *input*, int *size*) [inline]**

Definition at line 45 of file openssl.h.

13.106.3.2 rsa_context* d2i_RSA_PUBKEY (void * *ignore*, unsigned char ** *bufptr*, int *len*) [inline]

Definition at line 51 of file openssl.h.

References `rsa_context::E`, `rsa_context::len`, `mpi_msb()`, `mpi_read_binary()`, and `rsa_context::N`.

13.106.3.3 int RSA_private_decrypt (int *size*, unsigned char * *input*, unsigned char * *output*, RSA * *key*, int *ignore*) [inline]

Definition at line 105 of file openssl.h.

References `rsa_pkcs1_decrypt()`, and `RSA_PRIVATE`.

13.106.3.4 int RSA_private_encrypt (int *size*, unsigned char * *input*, unsigned char * *output*, RSA * *key*, int *ignore*) [inline]

Definition at line 107 of file openssl.h.

References `rsa_pkcs1_encrypt()`, `RSA_PRIVATE`, and `RSA_size`.

13.106.3.5 int RSA_public_decrypt (int *size*, unsigned char * *input*, unsigned char * *output*, RSA * *key*, int *ignore*) [inline]

Definition at line 104 of file openssl.h.

References `rsa_pkcs1_decrypt()`, and `RSA_PUBLIC`.

13.106.3.6 int RSA_public_encrypt (int *size*, unsigned char * *input*, unsigned char * *output*, RSA * *key*, int *ignore*) [inline]

Definition at line 106 of file openssl.h.

References `rsa_pkcs1_encrypt()`, `RSA_PUBLIC`, and `RSA_size`.

13.107 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/padlock.h File Reference

```
#include "xyssl/aes.h"
```

13.107.1 Detailed Description

Definition in file [padlock.h](#).

13.108 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/rsa.h File Reference

```
#include "bignum.h"
```

Data Structures

- struct [rsa_context](#)
RSA context structure.

Defines

- #define [XYSSL_ERR_RSA_BAD_INPUT_DATA](#) -0x0400
- #define [XYSSL_ERR_RSA_INVALID_PADDING](#) -0x0410
- #define [XYSSL_ERR_RSA_KEY_GEN_FAILED](#) -0x0420
- #define [XYSSL_ERR_RSA_KEY_CHECK_FAILED](#) -0x0430
- #define [XYSSL_ERR_RSA_PUBLIC_FAILED](#) -0x0440
- #define [XYSSL_ERR_RSA_PRIVATE_FAILED](#) -0x0450
- #define [XYSSL_ERR_RSA_VERIFY_FAILED](#) -0x0460
- #define [RSA_RAW](#) 0
- #define [RSA_MD2](#) 2
- #define [RSA_MD4](#) 3
- #define [RSA_MD5](#) 4
- #define [RSA_SHA1](#) 5
- #define [RSA_SHA256](#) 6
- #define [RSA_PUBLIC](#) 0
- #define [RSA_PRIVATE](#) 1
- #define [RSA_PKCS_V15](#) 0
- #define [RSA_PKCS_V21](#) 1
- #define [RSA_SIGN](#) 1
- #define [RSA_CRYPT](#) 2
- #define [ASN1_HASH_MDX](#)
- #define [ASN1_HASH_SHA1](#)

Functions

- void [rsa_init](#) ([rsa_context](#) *ctx, int padding, int hash_id, int(*f_rng)(void *), void *p_rng)
Initialize an RSA context.
- int [rsa_gen_key](#) ([rsa_context](#) *ctx, int nbits, int exponent)
Generate an RSA keypair.
- int [rsa_check_pubkey](#) ([rsa_context](#) *ctx)
Check a public RSA key.
- int [rsa_check_privkey](#) ([rsa_context](#) *ctx)
Check a private RSA key.

- `int rsa_public` (`rsa_context` *ctx, unsigned char *input, unsigned char *output)
Do an RSA public key operation.
- `int rsa_private` (`rsa_context` *ctx, unsigned char *input, unsigned char *output)
Do an RSA private key operation.
- `int rsa_pkcs1_encrypt` (`rsa_context` *ctx, `int` mode, `int` ilen, unsigned char *input, unsigned char *output)
Add the message padding, then do an RSA operation.
- `int rsa_pkcs1_decrypt` (`rsa_context` *ctx, `int` mode, `int` *olen, unsigned char *input, unsigned char *output)
Do an RSA operation, then remove the message padding.
- `int rsa_pkcs1_sign` (`rsa_context` *ctx, `int` mode, `int` hash_id, `int` hashlen, unsigned char *hash, unsigned char *sig)
Do a private RSA to sign a message digest.
- `int rsa_pkcs1_verify` (`rsa_context` *ctx, `int` mode, `int` hash_id, `int` hashlen, unsigned char *hash, unsigned char *sig)
Do a public RSA and check the message digest.
- `void rsa_free` (`rsa_context` *ctx)
Free the components of an RSA key.
- `int rsa_self_test` (`int` verbose)
Checkup routine.

13.108.1 Detailed Description

Definition in file [rsa.h](#).

13.108.2 Define Documentation

13.108.2.1 #define ASN1_HASH_MDX

Value:

```
"\x30\x20\x30\x0C\x06\x08\x2A\x86\x48" \
"\x86\xF7\x0D\x02\x00\x05\x00\x04\x10"
```

Definition at line 45 of file [rsa.h](#).

Referenced by [rsa_pkcs1_sign\(\)](#), and [rsa_pkcs1_verify\(\)](#).

13.108.2.2 #define ASN1_HASH_SHA1**Value:**

```
"\x30\x21\x30\x09\x06\x05\x2B\x0E\x03"  \
"\x02\x1A\x05\x00\x04\x14"
```

Definition at line 49 of file rsa.h.

Referenced by rsa_pkcs1_sign(), and rsa_pkcs1_verify().

13.108.2.3 #define RSA_CRYPT 2

Definition at line 34 of file rsa.h.

Referenced by rsa_pkcs1_decrypt(), and rsa_pkcs1_encrypt().

13.108.2.4 #define RSA_MD2 2

Definition at line 21 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_hash(), and x509parse_cert_info().

13.108.2.5 #define RSA_MD4 3

Definition at line 22 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_hash(), and x509parse_cert_info().

13.108.2.6 #define RSA_MD5 4

Definition at line 23 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), x509_hash(), and x509parse_cert_info().

13.108.2.7 #define RSA_PKCS_V15 0

Definition at line 30 of file rsa.h.

Referenced by generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), main(), rsa_decryption(), rsa_encryption(), rsa_pkcs1_decrypt(), rsa_pkcs1_encrypt(), rsa_pkcs1_sign(), and rsa_pkcs1_verify().

13.108.2.8 #define RSA_PKCS_V21 1

Definition at line 31 of file rsa.h.

13.108.2.9 #define RSA_PRIVATE 1

Definition at line 28 of file rsa.h.

Referenced by main(), RSA_private_decrypt(), RSA_private_encrypt(), rsa_self_test(), ssl_parse_client_key_exchange(), ssl_write_certificate_verify(), and ssl_write_server_key_exchange().

13.108.2.10 #define RSA_PUBLIC 0

Definition at line 27 of file rsa.h.

Referenced by main(), rsa_pkcs1_decrypt(), rsa_pkcs1_encrypt(), rsa_pkcs1_sign(), rsa_pkcs1_verify(), RSA_public_decrypt(), RSA_public_encrypt(), rsa_self_test(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), ssl_write_client_key_exchange(), and x509parse_verify().

13.108.2.11 #define RSA_RAW 0

Definition at line 20 of file rsa.h.

Referenced by rsa_pkcs1_sign(), rsa_pkcs1_verify(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), ssl_write_certificate_verify(), and ssl_write_server_key_exchange().

13.108.2.12 #define RSA_SHA1 5

Definition at line 24 of file rsa.h.

Referenced by main(), rsa_pkcs1_sign(), rsa_pkcs1_verify(), rsa_self_test(), x509_hash(), and x509parse_cert_info().

13.108.2.13 #define RSA_SHA256 6

Definition at line 25 of file rsa.h.

13.108.2.14 #define RSA_SIGN 1

Definition at line 33 of file rsa.h.

Referenced by rsa_pkcs1_sign(), and rsa_pkcs1_verify().

13.108.2.15 #define XYSSL_ERR_RSA_BAD_INPUT_DATA -0x0400

Definition at line 9 of file rsa.h.

Referenced by rsa_gen_key(), rsa_pkcs1_decrypt(), rsa_pkcs1_encrypt(), rsa_pkcs1_sign(), rsa_pkcs1_verify(), rsa_private(), and rsa_public().

13.108.2.16 #define XYSSL_ERR_RSA_INVALID_PADDING -0x0410

Definition at line 10 of file rsa.h.

Referenced by rsa_pkcs1_decrypt(), rsa_pkcs1_encrypt(), rsa_pkcs1_sign(), and rsa_pkcs1_verify().

13.108.2.17 #define XYSSL_ERR_RSA_KEY_CHECK_FAILED -0x0430

Definition at line 12 of file rsa.h.

Referenced by rsa_check_privkey(), and rsa_check_pubkey().

13.108.2.18 #define XYSSL_ERR_RSA_KEY_GEN_FAILED -0x0420

Definition at line 11 of file rsa.h.

Referenced by rsa_gen_key().

13.108.2.19 #define XYSSL_ERR_RSA_PRIVATE_FAILED -0x0450

Definition at line 14 of file rsa.h.

Referenced by rsa_private().

13.108.2.20 #define XYSSL_ERR_RSA_PUBLIC_FAILED -0x0440

Definition at line 13 of file rsa.h.

Referenced by rsa_public().

13.108.2.21 #define XYSSL_ERR_RSA_VERIFY_FAILED -0x0460

Definition at line 15 of file rsa.h.

Referenced by rsa_pkcs1_verify().

13.108.3 Function Documentation**13.108.3.1 int rsa_check_privkey (rsa_context * ctx)**

Check a private RSA key.

Parameters:

ctx RSA context to be checked

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Definition at line 152 of file rsa.c.

References rsa_context::D, rsa_context::E, MPI_CHK, mpi_cmp_int(), mpi_cmp_mpi(), mpi_free(), mpi_gcd(), mpi_init(), mpi_mod_mpi(), mpi_mul_mpi(), mpi_sub_int(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_check_pubkey(), and XYSSL_ERR_RSA_KEY_CHECK_FAILED.

Referenced by rsa_decryption(), rsa_self_test(), and x509parse_key().

13.108.3.2 int rsa_check_pubkey (rsa_context * ctx)

Check a public RSA key.

Parameters:

ctx RSA context to be checked

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Definition at line 132 of file rsa.c.

References `rsa_context::E`, `mpi_msb()`, `rsa_context::N`, `mpi::p`, and `XYSSL_ERR_RSA_KEY_CHECK_FAILED`.

Referenced by `rsa_check_privkey()`, `rsa_encryption()`, `rsa_self_test()`, and `x509parse_crt()`.

13.108.3.3 void rsa_free (rsa_context * ctx)

Free the components of an RSA key.

Definition at line 558 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `mpi_free()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_context::RN`, `rsa_context::RP`, and `rsa_context::RQ`.

Referenced by `main()`, `rsa_gen_key()`, `rsa_self_test()`, `ssl_test()`, `x509_free()`, `x509_self_test()`, and `x509parse_key()`.

13.108.3.4 int rsa_gen_key (rsa_context * ctx, int nbits, int exponent)

Generate an RSA keypair.

Parameters:

ctx RSA context that will hold the key

nbits size of the public key in bits

exponent public exponent (e.g., 65537)

Note:

[rsa_init\(\)](#) must be called beforehand to setup the RSA context (especially `f_rng` and `p_rng`).

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Definition at line 60 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::f_rng`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `mpi_swap()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::p_rng`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_KEY_GEN_FAILED`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, and `main()`.

13.108.3.5 void rsa_init (rsa_context * ctx, int padding, int hash_id, int(*) (void *) f_rng, void * p_rng)

Initialize an RSA context.

Parameters:

ctx RSA context to be initialized
padding RSA_PKCS_V15 or RSA_PKCS_V21
hash_id RSA_PKCS_V21 hash identifier
f_rng RNG function
p_rng RNG parameter

Note:

The *hash_id* parameter is actually ignored when using RSA_PKCS_V15 padding.
 Currently (xyssl-0.8), RSA_PKCS_V21 padding is not supported.

Definition at line 40 of file `rsa.c`.

References `rsa_context::f_rng`, `rsa_context::hash_id`, `rsa_context::p_rng`, and `rsa_context::padding`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, and `rsa_encryption()`.

13.108.3.6 `int rsa_pkcs1_decrypt(rsa_context * ctx, int mode, int * olen, unsigned char * input, unsigned char * output)`

Do an RSA operation, then remove the message padding.

Parameters:

ctx RSA context
mode RSA_PUBLIC or RSA_PRIVATE
input buffer holding the encrypted data
output buffer that will hold the plaintext
olen will contain the plaintext length

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The output buffer must be as large as the size of `ctx->N` (eg. 128 bytes if RSA-1024 is used).

Definition at line 326 of file `rsa.c`.

References `buf`, `int`, `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_decryption()`, `RSA_private_decrypt()`, `RSA_public_decrypt()`, `rsa_self_test()`, and `ssl_parse_client_key_exchange()`.

13.108.3.7 `int rsa_pkcs1_encrypt(rsa_context * ctx, int mode, int ilen, unsigned char * input, unsigned char * output)`

Add the message padding, then do an RSA operation.

Parameters:

ctx RSA context
mode RSA_PUBLIC or RSA_PRIVATE
ilen contains the the plaintext length
input buffer holding the data to be encrypted
output buffer that will hold the ciphertext

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The output buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 280 of file rsa.c.

References `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_encryption()`, `RSA_private_encrypt()`, `RSA_public_encrypt()`, `rsa_self_test()`, and `ssl_write_client_key_exchange()`.

13.108.3.8 `int rsa_pkcs1_sign(rsa_context * ctx, int mode, int hash_id, int hashlen, unsigned char * hash, unsigned char * sig)`

Do a private RSA to sign a message digest.

Parameters:

ctx RSA context
mode RSA_PUBLIC or RSA_PRIVATE
hash_id RSA_RAW, RSA_MD{2,4,5} or RSA_SHA{1,256}
hashlen message digest length (for RSA_RAW only)
hash buffer holding the message digest
sig buffer that will hold the ciphertext

Returns:

0 if the signing operation was successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 379 of file rsa.c.

References `ASN1_HASH_MDX`, `ASN1_HASH_SHA1`, `rsa_context::len`, `rsa_context::padding`, `RSA_MD2`, `RSA_MD4`, `RSA_MD5`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `RSA_RAW`, `RSA_SHA1`, `RSA_SIGN`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `main()`, `rsa_self_test()`, `ssl_write_certificate_verify()`, and `ssl_write_server_key_exchange()`.

13.108.3.9 **int rsa_pkcs1_verify** (*rsa_context* * *ctx*, *int mode*, *int hash_id*, *int hashlen*, *unsigned char* * *hash*, *unsigned char* * *sig*)

Do a public RSA and check the message digest.

Parameters:

ctx points to an RSA public key
mode RSA_PUBLIC or RSA_PRIVATE
hash_id RSA_RAW, RSA_MD{2,4,5} or RSA_SHA{1,256}
hashlen message digest length (for RSA_RAW only)
hash buffer holding the message digest
sig buffer holding the ciphertext

Returns:

0 if the verify operation was successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The "sig" buffer must be as large as the size of *ctx*->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 468 of file *rsa.c*.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, buf, int, rsa_context::len, rsa_context::padding, RSA_MD2, RSA_MD4, RSA_MD5, RSA_PKCS_V15, rsa_private(), rsa_public(), RSA_PUBLIC, RSA_RAW, RSA_SHA1, RSA_SIGN, XYSSL_ERR_RSA_BAD_INPUT_DATA, XYSSL_ERR_RSA_INVALID_PADDING, and XYSSL_ERR_RSA_VERIFY_FAILED.

Referenced by main(), rsa_self_test(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), and x509parse_verify().

13.108.3.10 **int rsa_private** (*rsa_context* * *ctx*, *unsigned char* * *input*, *unsigned char* * *output*)

Do an RSA private key operation.

Parameters:

ctx RSA context
input input buffer
output output buffer

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 221 of file *rsa.c*.

References rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::len, mpi_add_mpi(), MPI_CHK, mpi_cmp_mpi(), mpi_exp_mod(), mpi_free(), mpi_init(), mpi_mod_mpi(), mpi_mul_mpi(), mpi_read_binary(), mpi_sub_mpi(), mpi_write_binary(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, rsa_context::RN, rsa_context::RP, rsa_context::RQ, XYSSL_ERR_RSA_BAD_INPUT_DATA, and XYSSL_ERR_RSA_PRIVATE_FAILED.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

13.108.3.11 `int rsa_public (rsa_context * ctx, unsigned char * input, unsigned char * output)`

Do an RSA public key operation.

Parameters:

ctx RSA context
input input buffer
output output buffer

Returns:

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Note:

This function does NOT take care of message padding. Also, be sure to set `input[0] = 0`.
The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 187 of file `rsa.c`.

References `rsa_context::E`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_read_binary()`, `mpi_write_binary()`, `rsa_context::N`, `rsa_context::RN`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_PUBLIC_FAILED`.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

13.108.3.12 `int rsa_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 627 of file `rsa.c`.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `KEY_LEN`, `rsa_context::len`, `mpi_read_string()`, `rsa_context::N`, `rsa_context::P`, `PT_LEN`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `RSA_D`, `RSA_DP`, `RSA_DQ`, `RSA_E`, `rsa_free()`, `RSA_N`, `RSA_P`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, `rsa_pkcs1_verify()`, `RSA_PRIVATE`, `RSA_PT`, `RSA_PUBLIC`, `RSA_Q`, `RSA_QP`, `RSA_SHA1`, and `sha1()`.

Referenced by `main()`.

13.109 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha1.h File Reference

Data Structures

- struct [sha1_context](#)
SHA-1 context structure.

Functions

- void [sha1_starts](#) ([sha1_context](#) *ctx)
SHA-1 context setup.
- void [sha1_update](#) ([sha1_context](#) *ctx, unsigned char *input, int ilen)
SHA-1 process buffer.
- void [sha1_finish](#) ([sha1_context](#) *ctx, unsigned char output[20])
SHA-1 final digest.
- void [sha1](#) (unsigned char *input, int ilen, unsigned char output[20])
Output = SHA-1(input buffer).
- int [sha1_file](#) (char *path, unsigned char output[20])
Output = SHA-1(file contents).
- void [sha1_hmac_starts](#) ([sha1_context](#) *ctx, unsigned char *key, int keylen)
SHA-1 HMAC context setup.
- void [sha1_hmac_update](#) ([sha1_context](#) *ctx, unsigned char *input, int ilen)
SHA-1 HMAC process buffer.
- void [sha1_hmac_finish](#) ([sha1_context](#) *ctx, unsigned char output[20])
SHA-1 HMAC final digest.
- void [sha1_hmac](#) (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char output[20])
Output = HMAC-SHA-1(hmac key, input buffer).
- int [sha1_self_test](#) (int verbose)
Checkup routine.

13.109.1 Detailed Description

Definition in file [sha1.h](#).

13.109.2 Function Documentation

13.109.2.1 void sha1 (unsigned char * *input*, int *ilen*, unsigned char *output*[20])

Output = SHA-1(*input* buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output SHA-1 checksum result

Definition at line 313 of file sha1.c.

References sha1_finish(), sha1_starts(), and sha1_update().

Referenced by main(), rsa_self_test(), sha1_hmac_starts(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_finished(), ssl_parse_server_key_exchange(), ssl_write_finished(), ssl_write_server_key_exchange(), and x509_hash().

13.109.2.2 int sha1_file (char * *path*, unsigned char *output*[20])

Output = SHA-1(*file* contents).

Parameters:

path input file name
output SHA-1 checksum result

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha1.c.

References buf, f, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by main(), and sha1_wrapper().

13.109.2.3 void sha1_finish (sha1_context * *ctx*, unsigned char *output*[20])

SHA-1 final digest.

Parameters:

ctx SHA-1 context
output SHA-1 checksum result

Definition at line 284 of file sha1.c.

References PUT_ULONG_BE, sha1_padding, sha1_update(), sha1_context::state, and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_hmac_finish(), sha1_self_test(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

13.109.2.4 void sha1_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char *output*[20])

Output = HMAC-SHA-1(hmac key, input buffer).

Parameters:

key HMAC secret key
keylen length of the HMAC key
input buffer holding the data
ilen length of the input data
output HMAC-SHA-1 result

Definition at line 413 of file sha1.c.

References sha1_hmac_finish(), sha1_hmac_starts(), and sha1_hmac_update().

Referenced by ssl_decrypt_buf(), ssl_encrypt_buf(), and tls1_prf().

13.109.2.5 void sha1_hmac_finish (sha1_context * *ctx*, unsigned char *output*[20])

SHA-1 HMAC final digest.

Parameters:

ctx HMAC context
output SHA-1 HMAC checksum result

Definition at line 397 of file sha1.c.

References sha1_context::opad, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by sha1_hmac(), and sha1_self_test().

13.109.2.6 void sha1_hmac_starts (sha1_context * *ctx*, unsigned char * *key*, int *keylen*)

SHA-1 HMAC context setup.

Parameters:

ctx HMAC context to be initialized
key HMAC secret key
keylen length of the HMAC key

Definition at line 359 of file sha1.c.

References sha1_context::ipad, sha1_context::opad, sha1(), sha1_starts(), and sha1_update().

Referenced by sha1_hmac(), and sha1_self_test().

13.109.2.7 void sha1_hmac_update (sha1_context * *ctx*, unsigned char * *input*, int *ilen*)

SHA-1 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

Definition at line 389 of file sha1.c.

References sha1_update().

Referenced by sha1_hmac(), and sha1_self_test().

13.109.2.8 int sha1_self_test (int verbose)

Checksum routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 521 of file sha1.c.

References buf, sha1_finish(), sha1_hmac_finish(), sha1_hmac_starts(), sha1_hmac_test_buf, sha1_hmac_test_buflen, sha1_hmac_test_key, sha1_hmac_test_keylen, sha1_hmac_test_sum, sha1_hmac_update(), sha1_starts(), sha1_test_buf, sha1_test_buflen, sha1_test_sum, and sha1_update().

Referenced by main().

13.109.2.9 void sha1_starts (sha1_context * ctx)

SHA-1 context setup.

Parameters:

ctx context to be initialized

Definition at line 61 of file sha1.c.

References sha1_context::state, and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_hmac_finish(), sha1_hmac_starts(), sha1_self_test(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_init(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

13.109.2.10 void sha1_update (sha1_context * ctx, unsigned char * input, int ilen)

SHA-1 process buffer.

Parameters:

ctx SHA-1 context

input buffer holding the data

ilen length of the input data

Definition at line 232 of file sha1.c.

References `sha1_context::buffer`, `sha1_process()`, and `sha1_context::total`.

Referenced by `sha1()`, `sha1_file()`, `sha1_finish()`, `sha1_hmac_finish()`, `sha1_hmac_starts()`, `sha1_hmac_update()`, `sha1_self_test()`, `ssl_calc_finished()`, `ssl_calc_verify()`, `ssl_derive_keys()`, `ssl_mac_sha1()`, `ssl_parse_client_hello()`, `ssl_parse_server_key_exchange()`, `ssl_read_record()`, `ssl_write_record()`, and `ssl_write_server_key_exchange()`.

13.110 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha2.h File Reference

Data Structures

- struct [sha2_context](#)
SHA-256 context structure.

Functions

- void [sha2_starts](#) ([sha2_context](#) *ctx, [int](#) is224)
SHA-256 context setup.
- void [sha2_update](#) ([sha2_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-256 process buffer.
- void [sha2_finish](#) ([sha2_context](#) *ctx, unsigned char output[32])
SHA-256 final digest.
- void [sha2](#) (unsigned char *input, [int](#) ilen, unsigned char output[32], [int](#) is224)
Output = SHA-256(input buffer).
- [int](#) [sha2_file](#) (char *path, unsigned char output[32], [int](#) is224)
Output = SHA-256(file contents).
- void [sha2_hmac_starts](#) ([sha2_context](#) *ctx, unsigned char *key, [int](#) keylen, [int](#) is224)
SHA-256 HMAC context setup.
- void [sha2_hmac_update](#) ([sha2_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-256 HMAC process buffer.
- void [sha2_hmac_finish](#) ([sha2_context](#) *ctx, unsigned char output[32])
SHA-256 HMAC final digest.
- void [sha2_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char output[32], [int](#) is224)
Output = HMAC-SHA-256(hmac key, input buffer).
- [int](#) [sha2_self_test](#) ([int](#) verbose)
Checkup routine.

13.110.1 Detailed Description

Definition in file [sha2.h](#).

13.110.2 Function Documentation

13.110.2.1 void sha2 (unsigned char * *input*, int *ilen*, unsigned char *output*[32], int *is224*)

Output = SHA-256(input buffer).

Parameters:

input buffer holding the data

ilen length of the input data

output SHA-224/256 checksum result

is224 0 = use SHA256, 1 = use SHA224

Definition at line 314 of file sha2.c.

References sha2_finish(), sha2_starts(), and sha2_update().

Referenced by main(), and sha2_hmac_starts().

13.110.2.2 int sha2_file (char * *path*, unsigned char *output*[32], int *is224*)

Output = SHA-256(file contents).

Parameters:

path input file name

output SHA-224/256 checksum result

is224 0 = use SHA256, 1 = use SHA224

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 329 of file sha2.c.

References buf, f, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by sha2_wrapper().

13.110.2.3 void sha2_finish (sha2_context * *ctx*, unsigned char *output*[32])

SHA-256 final digest.

Parameters:

ctx SHA-256 context

output SHA-224/256 checksum result

Definition at line 280 of file sha2.c.

References sha2_context::is224, PUT_ULONG_BE, sha2_padding, sha2_update(), sha2_context::state, and sha2_context::total.

Referenced by aes_en_de(), main(), sha2(), sha2_file(), sha2_hmac_finish(), and sha2_self_test().

13.110.2.4 `void sha2_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char output[32], int is224)`

Output = HMAC-SHA-256(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key

input buffer holding the data

ilen length of the input data

output HMAC-SHA-224/256 result

is224 0 = use SHA256, 1 = use SHA224

Definition at line 420 of file sha2.c.

References sha2_hmac_finish(), sha2_hmac_starts(), and sha2_hmac_update().

13.110.2.5 `void sha2_hmac_finish (sha2_context * ctx, unsigned char output[32])`

SHA-256 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-224/256 HMAC checksum result

Definition at line 400 of file sha2.c.

References sha2_context::is224, sha2_context::opad, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by aes_en_de(), main(), sha2_hmac(), and sha2_self_test().

13.110.2.6 `void sha2_hmac_starts (sha2_context * ctx, unsigned char * key, int keylen, int is224)`

SHA-256 HMAC context setup.

Parameters:

ctx HMAC context to be initialized

key HMAC secret key

keylen length of the HMAC key

is224 0 = use SHA256, 1 = use SHA224

Definition at line 361 of file sha2.c.

References sha2_context::ipad, sha2_context::opad, sha2(), sha2_starts(), and sha2_update().

Referenced by aes_en_de(), main(), sha2_hmac(), and sha2_self_test().

13.110.2.7 void sha2_hmac_update (sha2_context * *ctx*, unsigned char * *input*, int *ilen*)

SHA-256 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

Definition at line 392 of file sha2.c.

References sha2_update().

Referenced by aes_en_de(), main(), sha2_hmac(), and sha2_self_test().

13.110.2.8 int sha2_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 599 of file sha2.c.

References buf, sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), sha2_hmac_test_buf, sha2_hmac_test_buflen, sha2_hmac_test_key, sha2_hmac_test_keylen, sha2_hmac_test_sum, sha2_hmac_update(), sha2_starts(), sha2_test_buf, sha2_test_buflen, sha2_test_sum, and sha2_update().

Referenced by main().

13.110.2.9 void sha2_starts (sha2_context * *ctx*, int *is224*)

SHA-256 context setup.

Parameters:

ctx context to be initialized

is224 0 = use SHA256, 1 = use SHA224

Definition at line 61 of file sha2.c.

References sha2_context::is224, sha2_context::state, and sha2_context::total.

Referenced by aes_en_de(), main(), sha2(), sha2_file(), sha2_hmac_finish(), sha2_hmac_starts(), and sha2_self_test().

13.110.2.10 void sha2_update (sha2_context * *ctx*, unsigned char * *input*, int *ilen*)

SHA-256 process buffer.

Parameters:

ctx SHA-256 context

input buffer holding the data

ilen length of the input data

Definition at line 228 of file sha2.c.

References sha2_context::buffer, sha2_process(), and sha2_context::total.

Referenced by aes_en_de(), main(), sha2(), sha2_file(), sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), sha2_hmac_update(), and sha2_self_test().

13.111 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/sha4.h File Reference

Data Structures

- struct [sha4_context](#)
SHA-512 context structure.

Defines

- #define [UL64](#)(x) x##ULL
- #define [int64](#) long long

Functions

- void [sha4_starts](#) ([sha4_context](#) *ctx, [int](#) is384)
SHA-512 context setup.
- void [sha4_update](#) ([sha4_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-512 process buffer.
- void [sha4_finish](#) ([sha4_context](#) *ctx, unsigned char output[64])
SHA-512 final digest.
- void [sha4](#) (unsigned char *input, [int](#) ilen, unsigned char output[64], [int](#) is384)
Output = SHA-512(input buffer).
- [int](#) [sha4_file](#) (char *path, unsigned char output[64], [int](#) is384)
Output = SHA-512(file contents).
- void [sha4_hmac_starts](#) ([sha4_context](#) *ctx, unsigned char *key, [int](#) keylen, [int](#) is384)
SHA-512 HMAC context setup.
- void [sha4_hmac_update](#) ([sha4_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-512 HMAC process buffer.
- void [sha4_hmac_finish](#) ([sha4_context](#) *ctx, unsigned char output[64])
SHA-512 HMAC final digest.
- void [sha4_hmac](#) (unsigned char *key, [int](#) keylen, unsigned char *input, [int](#) ilen, unsigned char output[64], [int](#) is384)
Output = HMAC-SHA-512(hmac key, input buffer).
- [int](#) [sha4_self_test](#) ([int](#) verbose)
Checkup routine.

13.111.1 Detailed Description

Definition in file [sha4.h](#).

13.111.2 Define Documentation

13.111.2.1 `#define int64 long long`

Definition at line 12 of file sha4.h.

Referenced by sha4_finish(), sha4_process(), and sha4_update().

13.111.2.2 `#define UL64(x) x##ULL`

Definition at line 11 of file sha4.h.

Referenced by sha4_starts().

13.111.3 Function Documentation

13.111.3.1 `void sha4(unsigned char *input, int ilen, unsigned char output[64], int is384)`

Output = SHA-512(input buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output SHA-384/512 checksum result
is384 0 = use SHA512, 1 = use SHA384

Definition at line 312 of file sha4.c.

References sha4_finish(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac_starts().

13.111.3.2 `int sha4_file(char *path, unsigned char output[64], int is384)`

Output = SHA-512(file contents).

Parameters:

path input file name
output SHA-384/512 checksum result
is384 0 = use SHA512, 1 = use SHA384

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha4.c.

References buf, f, sha4_finish(), sha4_starts(), and sha4_update().

13.111.3.3 void sha4_finish (sha4_context * ctx, unsigned char output[64])

SHA-512 final digest.

Parameters:

ctx SHA-512 context

output SHA-384/512 checksum result

Definition at line 276 of file sha4.c.

References int, int64, sha4_context::is384, PUT_UINT64_BE, sha4_padding, sha4_update(), sha4_context::state, and sha4_context::total.

Referenced by sha4(), sha4_file(), sha4_hmac_finish(), and sha4_self_test().

13.111.3.4 void sha4_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char output[64], int is384)

Output = HMAC-SHA-512(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key

input buffer holding the data

ilen length of the input data

output HMAC-SHA-384/512 result

is384 0 = use SHA512, 1 = use SHA384

Definition at line 419 of file sha4.c.

References sha4_hmac_finish(), sha4_hmac_starts(), and sha4_hmac_update().

13.111.3.5 void sha4_hmac_finish (sha4_context * ctx, unsigned char output[64])

SHA-512 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-384/512 HMAC checksum result

Definition at line 399 of file sha4.c.

References sha4_context::is384, sha4_context::opad, sha4_finish(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac(), and sha4_self_test().

13.111.3.6 void sha4_hmac_starts (sha4_context * ctx, unsigned char * key, int keylen, int is384)

SHA-512 HMAC context setup.

Parameters:

ctx HMAC context to be initialized
is384 0 = use SHA512, 1 = use SHA384
key HMAC secret key
keylen length of the HMAC key

Definition at line 359 of file sha4.c.

References sha4_context::ipad, sha4_context::opad, sha4(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac(), and sha4_self_test().

13.111.3.7 void sha4_hmac_update (sha4_context * ctx, unsigned char * input, int ilen)

SHA-512 HMAC process buffer.

Parameters:

ctx HMAC context
input buffer holding the data
ilen length of the input data

Definition at line 390 of file sha4.c.

References sha4_update().

Referenced by sha4_hmac(), and sha4_self_test().

13.111.3.8 int sha4_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 654 of file sha4.c.

References buf, sha4_finish(), sha4_hmac_finish(), sha4_hmac_starts(), sha4_hmac_test_buf, sha4_hmac_test_buflen, sha4_hmac_test_key, sha4_hmac_test_keylen, sha4_hmac_test_sum, sha4_hmac_update(), sha4_starts(), sha4_test_buf, sha4_test_buflen, sha4_test_sum, and sha4_update().

Referenced by main().

13.111.3.9 void sha4_starts (sha4_context * ctx, int is384)

SHA-512 context setup.

Parameters:

ctx context to be initialized
is384 0 = use SHA512, 1 = use SHA384

Definition at line 116 of file sha4.c.

References sha4_context::is384, sha4_context::state, sha4_context::total, and UL64.

Referenced by sha4(), sha4_file(), sha4_hmac_finish(), sha4_hmac_starts(), and sha4_self_test().

13.111.3.10 void sha4_update (sha4_context * *ctx*, unsigned char * *input*, int *ilen*)

SHA-512 process buffer.

Parameters:

ctx SHA-512 context

input buffer holding the data

ilen length of the input data

Definition at line 221 of file sha4.c.

References sha4_context::buffer, int, int64, sha4_process(), and sha4_context::total.

Referenced by sha4(), sha4_file(), sha4_finish(), sha4_hmac_finish(), sha4_hmac_starts(), sha4_hmac_update(), and sha4_self_test().

13.112 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/ssl.h File Reference

```
#include <time.h>
#include "xyssl/net.h"
#include "xyssl/dhm.h"
#include "xyssl/rsa.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include "xyssl/x509.h"
```

Data Structures

- struct [_ssl_session](#)
- struct [_ssl_context](#)

Defines

- #define [XYSSL_ERR_SSL_FEATURE_UNAVAILABLE](#) -0x1000
- #define [XYSSL_ERR_SSL_BAD_INPUT_DATA](#) -0x1800
- #define [XYSSL_ERR_SSL_INVALID_MAC](#) -0x2000
- #define [XYSSL_ERR_SSL_INVALID_RECORD](#) -0x2800
- #define [XYSSL_ERR_SSL_INVALID_MODULUS_SIZE](#) -0x3000
- #define [XYSSL_ERR_SSL_UNKNOWN_CIPHER](#) -0x3800
- #define [XYSSL_ERR_SSL_NO_CIPHER_CHOSEN](#) -0x4000
- #define [XYSSL_ERR_SSL_NO_SESSION_FOUND](#) -0x4800
- #define [XYSSL_ERR_SSL_NO_CLIENT_CERTIFICATE](#) -0x5000
- #define [XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE](#) -0x5800
- #define [XYSSL_ERR_SSL_CERTIFICATE_REQUIRED](#) -0x6000
- #define [XYSSL_ERR_SSL_PRIVATE_KEY_REQUIRED](#) -0x6800
- #define [XYSSL_ERR_SSL_CA_CHAIN_REQUIRED](#) -0x7000
- #define [XYSSL_ERR_SSL_UNEXPECTED_MESSAGE](#) -0x7800
- #define [XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE](#) -0x8000
- #define [XYSSL_ERR_SSL_PEER_VERIFY_FAILED](#) -0x8800
- #define [XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY](#) -0x9000
- #define [XYSSL_ERR_SSL_BAD_HS_CLIENT_HELLO](#) -0x9800
- #define [XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO](#) -0xA000
- #define [XYSSL_ERR_SSL_BAD_HS_CERTIFICATE](#) -0xA800
- #define [XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_REQUEST](#) -0xB000
- #define [XYSSL_ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE](#) -0xB800
- #define [XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO_DONE](#) -0xC000
- #define [XYSSL_ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE](#) -0xC800
- #define [XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_VERIFY](#) -0xD000
- #define [XYSSL_ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC](#) -0xD800
- #define [XYSSL_ERR_SSL_BAD_HS_FINISHED](#) -0xE000
- #define [SSL_MAJOR_VERSION_3](#) 3

- `#define SSL_MINOR_VERSION_0` 0
- `#define SSL_MINOR_VERSION_1` 1
- `#define SSL_MINOR_VERSION_2` 2
- `#define SSL_IS_CLIENT` 0
- `#define SSL_IS_SERVER` 1
- `#define SSL_COMPRESS_NULL` 0
- `#define SSL_VERIFY_NONE` 0
- `#define SSL_VERIFY_OPTIONAL` 1
- `#define SSL_VERIFY_REQUIRED` 2
- `#define SSL_MAX_CONTENT_LEN` 16384
- `#define SSL_BUFFER_LEN` (SSL_MAX_CONTENT_LEN + 512)
- `#define SSL_RSA_RC4_128_MD5` 4
- `#define SSL_RSA_RC4_128_SHA` 5
- `#define SSL_RSA_DES_168_SHA` 10
- `#define SSL_EDH_RSA_DES_168_SHA` 22
- `#define SSL_RSA_AES_128_SHA` 47
- `#define SSL_RSA_AES_256_SHA` 53
- `#define SSL_EDH_RSA_AES_256_SHA` 57
- `#define SSL_MSG_CHANGE_CIPHER_SPEC` 20
- `#define SSL_MSG_ALERT` 21
- `#define SSL_MSG_HANDSHAKE` 22
- `#define SSL_MSG_APPLICATION_DATA` 23
- `#define SSL_ALERT_CLOSE_NOTIFY` 0
- `#define SSL_ALERT_WARNING` 1
- `#define SSL_ALERT_FATAL` 2
- `#define SSL_ALERT_NO_CERTIFICATE` 41
- `#define SSL_HS_HELLO_REQUEST` 0
- `#define SSL_HS_CLIENT_HELLO` 1
- `#define SSL_HS_SERVER_HELLO` 2
- `#define SSL_HS_CERTIFICATE` 11
- `#define SSL_HS_SERVER_KEY_EXCHANGE` 12
- `#define SSL_HS_CERTIFICATE_REQUEST` 13
- `#define SSL_HS_SERVER_HELLO_DONE` 14
- `#define SSL_HS_CERTIFICATE_VERIFY` 15
- `#define SSL_HS_CLIENT_KEY_EXCHANGE` 16
- `#define SSL_HS_FINISHED` 20
- `#define TLS_EXT_SERVERNAME` 0
- `#define TLS_EXT_SERVERNAME_HOSTNAME` 0

Typedefs

- `typedef struct _ssl_session ssl_session`
- `typedef struct _ssl_context ssl_context`

Enumerations

- enum `ssl_states` {
`SSL_HELLO_REQUEST`, `SSL_CLIENT_HELLO`, `SSL_SERVER_HELLO`, `SSL_SERVER_CERTIFICATE`,
`SSL_SERVER_KEY_EXCHANGE`, `SSL_CERTIFICATE_REQUEST`, `SSL_SERVER_HELLO_DONE`, `SSL_CLIENT_CERTIFICATE`,
`SSL_CLIENT_KEY_EXCHANGE`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`,
`SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_FLUSH_BUFFERS`, `SSL_HANDSHAKE_OVER` }

Functions

- int `ssl_init` (`ssl_context` *ssl)
Initialize an SSL context.
- void `ssl_set_endpoint` (`ssl_context` *ssl, int endpoint)
Set the current endpoint type.
- void `ssl_set_authmode` (`ssl_context` *ssl, int authmode)
Set the certificate verification mode.
- void `ssl_set_rng` (`ssl_context` *ssl, int(*f_rng)(void *), void *p_rng)
Set the random number generator callback.
- void `ssl_set_dbg` (`ssl_context` *ssl, void(*f_dbg)(void *, int, char *), void *p_dbg)
Set the debug callback.
- void `ssl_set_bio` (`ssl_context` *ssl, int(*f_rcv)(void *, unsigned char *, int), void *p_rcv, int(*f_send)(void *, unsigned char *, int), void *p_send)
Set the underlying BIO read and write callbacks.
- void `ssl_set_scb` (`ssl_context` *ssl, int(*s_get)(`ssl_context` *), int(*s_set)(`ssl_context` *))
Set the session callbacks (server-side only).
- void `ssl_set_session` (`ssl_context` *ssl, int resume, int timeout, `ssl_session` *session)
Set the session resuming flag, timeout and data.
- void `ssl_set_ciphers` (`ssl_context` *ssl, int *ciphers)
Set the list of allowed ciphersuites.
- void `ssl_set_ca_chain` (`ssl_context` *ssl, `x509_cert` *ca_chain, char *peer_cn)
Set the data required to verify peer certificate.
- void `ssl_set_own_cert` (`ssl_context` *ssl, `x509_cert` *own_cert, `rsa_context` *rsa_key)
Set own certificate and private key.
- int `ssl_set_dh_param` (`ssl_context` *ssl, char *dhm_P, char *dhm_G)

Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).

- `int ssl_set_hostname (ssl_context *ssl, char *hostname)`
Set hostname for ServerName TLS Extension.
- `int ssl_get_bytes_avail (ssl_context *ssl)`
Return the number of data bytes available to read.
- `int ssl_get_verify_result (ssl_context *ssl)`
Return the result of the certificate verification.
- `char * ssl_get_cipher (ssl_context *ssl)`
Return the name of the current cipher.
- `int ssl_handshake (ssl_context *ssl)`
Perform the SSL handshake.
- `int ssl_read (ssl_context *ssl, unsigned char *buf, int len)`
Read at most 'len' application data bytes.
- `int ssl_write (ssl_context *ssl, unsigned char *buf, int len)`
Write exactly 'len' application data bytes.
- `int ssl_close_notify (ssl_context *ssl)`
Notify the peer that the connection is being closed.
- `void ssl_free (ssl_context *ssl)`
Free an SSL context.
- `int ssl_handshake_client (ssl_context *ssl)`
- `int ssl_handshake_server (ssl_context *ssl)`
- `int ssl_derive_keys (ssl_context *ssl)`
- `void ssl_calc_verify (ssl_context *ssl, unsigned char hash[36])`
- `int ssl_read_record (ssl_context *ssl)`
- `int ssl_fetch_input (ssl_context *ssl, int nb_want)`
- `int ssl_write_record (ssl_context *ssl)`
- `int ssl_flush_output (ssl_context *ssl)`
- `int ssl_parse_certificate (ssl_context *ssl)`
- `int ssl_write_certificate (ssl_context *ssl)`
- `int ssl_parse_change_cipher_spec (ssl_context *ssl)`
- `int ssl_write_change_cipher_spec (ssl_context *ssl)`
- `int ssl_parse_finished (ssl_context *ssl)`
- `int ssl_write_finished (ssl_context *ssl)`

Variables

- `int ssl_default_ciphers []`

13.112.1 Detailed Description

Definition in file [ssl.h](#).

13.112.2 Define Documentation

13.112.2.1 #define SSL_ALERT_CLOSE_NOTIFY 0

Definition at line 87 of file [ssl.h](#).

Referenced by [ssl_close_notify\(\)](#), and [ssl_read_record\(\)](#).

13.112.2.2 #define SSL_ALERT_FATAL 2

Definition at line 89 of file [ssl.h](#).

Referenced by [ssl_read_record\(\)](#).

13.112.2.3 #define SSL_ALERT_NO_CERTIFICATE 41

Definition at line 90 of file [ssl.h](#).

Referenced by [ssl_parse_certificate\(\)](#), and [ssl_write_certificate\(\)](#).

13.112.2.4 #define SSL_ALERT_WARNING 1

Definition at line 88 of file [ssl.h](#).

Referenced by [ssl_close_notify\(\)](#), [ssl_parse_certificate\(\)](#), [ssl_read_record\(\)](#), and [ssl_write_certificate\(\)](#).

13.112.2.5 #define SSL_BUFFER_LEN (SSL_MAX_CONTENT_LEN + 512)

Definition at line 66 of file [ssl.h](#).

Referenced by [ssl_free\(\)](#), and [ssl_init\(\)](#).

13.112.2.6 #define SSL_COMPRESS_NULL 0

Definition at line 54 of file [ssl.h](#).

Referenced by [ssl_parse_server_hello\(\)](#), [ssl_write_client_hello\(\)](#), and [ssl_write_server_hello\(\)](#).

13.112.2.7 #define SSL_EDH_RSA_AES_256_SHA 57

Definition at line 77 of file [ssl.h](#).

Referenced by [main\(\)](#), [ssl_derive_keys\(\)](#), [ssl_get_cipher\(\)](#), [ssl_parse_client_key_exchange\(\)](#), [ssl_parse_server_key_exchange\(\)](#), [ssl_write_client_key_exchange\(\)](#), and [ssl_write_server_key_exchange\(\)](#).

13.112.2.8 #define SSL_EDH_RSA_DES_168_SHA 22

Definition at line 74 of file ssl.h.

Referenced by main(), ssl_derive_keys(), ssl_get_cipher(), ssl_parse_client_key_exchange(), ssl_parse_server_key_exchange(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

13.112.2.9 #define SSL_HS_CERTIFICATE 11

Definition at line 95 of file ssl.h.

Referenced by ssl_parse_certificate(), and ssl_write_certificate().

13.112.2.10 #define SSL_HS_CERTIFICATE_REQUEST 13

Definition at line 97 of file ssl.h.

Referenced by ssl_parse_certificate_request(), and ssl_write_certificate_request().

13.112.2.11 #define SSL_HS_CERTIFICATE_VERIFY 15

Definition at line 99 of file ssl.h.

Referenced by ssl_parse_certificate_verify(), and ssl_write_certificate_verify().

13.112.2.12 #define SSL_HS_CLIENT_HELLO 1

Definition at line 93 of file ssl.h.

Referenced by ssl_parse_client_hello(), and ssl_write_client_hello().

13.112.2.13 #define SSL_HS_CLIENT_KEY_EXCHANGE 16

Definition at line 100 of file ssl.h.

Referenced by ssl_parse_client_key_exchange(), and ssl_write_client_key_exchange().

13.112.2.14 #define SSL_HS_FINISHED 20

Definition at line 101 of file ssl.h.

Referenced by ssl_parse_finished(), and ssl_write_finished().

13.112.2.15 #define SSL_HS_HELLO_REQUEST 0

Definition at line 92 of file ssl.h.

13.112.2.16 #define SSL_HS_SERVER_HELLO 2

Definition at line 94 of file ssl.h.

Referenced by ssl_parse_server_hello(), and ssl_write_server_hello().

13.112.2.17 #define SSL_HS_SERVER_HELLO_DONE 14

Definition at line 98 of file ssl.h.

Referenced by ssl_parse_server_hello_done(), and ssl_write_server_hello_done().

13.112.2.18 #define SSL_HS_SERVER_KEY_EXCHANGE 12

Definition at line 96 of file ssl.h.

Referenced by ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

13.112.2.19 #define SSL_IS_CLIENT 0

Definition at line 52 of file ssl.h.

Referenced by main(), ssl_calc_finished(), ssl_derive_keys(), ssl_handshake(), ssl_parse_finished(), ssl_test(), ssl_write_certificate(), and ssl_write_finished().

13.112.2.20 #define SSL_IS_SERVER 1

Definition at line 53 of file ssl.h.

Referenced by main(), ssl_handshake(), ssl_parse_certificate(), ssl_parse_finished(), and ssl_test().

13.112.2.21 #define SSL_MAJOR_VERSION_3 3

Definition at line 47 of file ssl.h.

Referenced by ssl_parse_client_hello(), ssl_parse_server_hello(), and ssl_write_client_hello().

13.112.2.22 #define SSL_MAX_CONTENT_LEN 16384

Definition at line 60 of file ssl.h.

Referenced by ssl_read_record(), ssl_write(), and ssl_write_certificate().

13.112.2.23 #define SSL_MINOR_VERSION_0 0

SSL v3.0

Definition at line 48 of file ssl.h.

Referenced by ssl_calc_finished(), ssl_calc_verify(), ssl_decrypt_buf(), ssl_derive_keys(), ssl_encrypt_buf(), ssl_parse_certificate(), ssl_parse_client_key_exchange(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_read_record(), ssl_write_certificate(), ssl_write_client_hello(), ssl_write_client_key_exchange(), and ssl_write_finished().

13.112.2.24 #define SSL_MINOR_VERSION_1 1

TLS v1.0

Definition at line 49 of file ssl.h.

Referenced by `ssl_parse_client_hello()`, `ssl_parse_server_hello()`, `ssl_read_record()`, and `ssl_write_client_hello()`.

13.112.2.25 #define SSL_MINOR_VERSION_2 2

TLS v1.1

Definition at line 50 of file `ssl.h`.

13.112.2.26 #define SSL_MSG_ALERT 21

Definition at line 83 of file `ssl.h`.

Referenced by `ssl_close_notify()`, `ssl_parse_certificate()`, `ssl_read_record()`, and `ssl_write_certificate()`.

13.112.2.27 #define SSL_MSG_APPLICATION_DATA 23

Definition at line 85 of file `ssl.h`.

Referenced by `ssl_read()`, and `ssl_write()`.

13.112.2.28 #define SSL_MSG_CHANGE_CIPHER_SPEC 20

Definition at line 82 of file `ssl.h`.

Referenced by `ssl_parse_change_cipher_spec()`, and `ssl_write_change_cipher_spec()`.

13.112.2.29 #define SSL_MSG_HANDSHAKE 22

Definition at line 84 of file `ssl.h`.

Referenced by `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `ssl_read_record()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_record()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

13.112.2.30 #define SSL_RSA_AES_128_SHA 47

Definition at line 75 of file `ssl.h`.

Referenced by `main()`, `ssl_derive_keys()`, and `ssl_get_cipher()`.

13.112.2.31 #define SSL_RSA_AES_256_SHA 53

Definition at line 76 of file `ssl.h`.

Referenced by `main()`, `ssl_derive_keys()`, and `ssl_get_cipher()`.

13.112.2.32 #define SSL_RSA_DES_168_SHA 10

Definition at line 73 of file ssl.h.

Referenced by main(), ssl_derive_keys(), and ssl_get_cipher().

13.112.2.33 #define SSL_RSA_RC4_128_MD5 4

Definition at line 71 of file ssl.h.

Referenced by main(), ssl_derive_keys(), and ssl_get_cipher().

13.112.2.34 #define SSL_RSA_RC4_128_SHA 5

Definition at line 72 of file ssl.h.

Referenced by main(), ssl_derive_keys(), and ssl_get_cipher().

13.112.2.35 #define SSL_VERIFY_NONE 0

Definition at line 56 of file ssl.h.

Referenced by main(), ssl_parse_certificate(), ssl_test(), and ssl_write_certificate_request().

13.112.2.36 #define SSL_VERIFY_OPTIONAL 1

Definition at line 57 of file ssl.h.

Referenced by main(), and ssl_parse_certificate().

13.112.2.37 #define SSL_VERIFY_REQUIRED 2

Definition at line 58 of file ssl.h.

Referenced by ssl_parse_certificate().

13.112.2.38 #define TLS_EXT_SERVERNAME 0

Definition at line 106 of file ssl.h.

Referenced by ssl_write_client_hello().

13.112.2.39 #define TLS_EXT_SERVERNAME_HOSTNAME 0

Definition at line 107 of file ssl.h.

Referenced by ssl_write_client_hello().

13.112.2.40 #define XYSSL_ERR_SSL_BAD_HS_CERTIFICATE -0xA800

Definition at line 35 of file ssl.h.

Referenced by ssl_parse_certificate().

13.112.2.41 #define XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_REQUEST -0xB000

Definition at line 36 of file ssl.h.

13.112.2.42 #define XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_VERIFY -0xD000

Definition at line 40 of file ssl.h.

Referenced by ssl_parse_certificate_verify().

13.112.2.43 #define XYSSL_ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC -0xD800

Definition at line 41 of file ssl.h.

Referenced by ssl_parse_change_cipher_spec().

13.112.2.44 #define XYSSL_ERR_SSL_BAD_HS_CLIENT_HELLO -0x9800

Definition at line 33 of file ssl.h.

Referenced by ssl_parse_client_hello().

13.112.2.45 #define XYSSL_ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE -0xC800

Definition at line 39 of file ssl.h.

Referenced by ssl_parse_client_key_exchange().

13.112.2.46 #define XYSSL_ERR_SSL_BAD_HS_FINISHED -0xE000

Definition at line 42 of file ssl.h.

Referenced by ssl_parse_finished().

13.112.2.47 #define XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO -0xA000

Definition at line 34 of file ssl.h.

Referenced by ssl_parse_server_hello().

13.112.2.48 #define XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO_DONE -0xC000

Definition at line 38 of file ssl.h.

Referenced by ssl_parse_server_hello_done().

13.112.2.49 #define XYSSL_ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE -0xB800

Definition at line 37 of file ssl.h.

Referenced by ssl_parse_server_key_exchange().

13.112.2.50 #define XYSSL_ERR_SSL_BAD_INPUT_DATA -0x1800

Definition at line 17 of file ssl.h.

Referenced by ssl_handshake_client(), ssl_handshake_server(), ssl_set_hostname(), and tls1_prf().

13.112.2.51 #define XYSSL_ERR_SSL_CA_CHAIN_REQUIRED -0x7000

Definition at line 28 of file ssl.h.

Referenced by ssl_parse_certificate().

13.112.2.52 #define XYSSL_ERR_SSL_CERTIFICATE_REQUIRED -0x6000

Definition at line 26 of file ssl.h.

Referenced by ssl_write_certificate().

13.112.2.53 #define XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE -0x5800

Definition at line 25 of file ssl.h.

Referenced by ssl_write_certificate().

13.112.2.54 #define XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE -0x8000

Definition at line 30 of file ssl.h.

Referenced by ssl_read_record().

13.112.2.55 #define XYSSL_ERR_SSL_FEATURE_UNAVAILABLE -0x1000

Definition at line 16 of file ssl.h.

Referenced by ssl_decrypt_buf(), ssl_derive_keys(), ssl_encrypt_buf(), ssl_handshake(), ssl_parse_client_key_exchange(), ssl_parse_server_key_exchange(), ssl_write_client_key_exchange(), and ssl_write_server_key_exchange().

13.112.2.56 #define XYSSL_ERR_SSL_INVALID_MAC -0x2000

Definition at line 18 of file ssl.h.

Referenced by ssl_decrypt_buf().

13.112.2.57 #define XYSSL_ERR_SSL_INVALID_MODULUS_SIZE -0x3000

Definition at line 20 of file ssl.h.

13.112.2.58 #define XYSSL_ERR_SSL_INVALID_RECORD -0x2800

Definition at line 19 of file ssl.h.

Referenced by ssl_read_record().

13.112.2.59 #define XYSSL_ERR_SSL_NO_CIPHER_CHOSEN -0x4000

Definition at line 22 of file ssl.h.

Referenced by ssl_parse_client_hello().

13.112.2.60 #define XYSSL_ERR_SSL_NO_CLIENT_CERTIFICATE -0x5000

Definition at line 24 of file ssl.h.

Referenced by ssl_parse_certificate().

13.112.2.61 #define XYSSL_ERR_SSL_NO_SESSION_FOUND -0x4800

Definition at line 23 of file ssl.h.

13.112.2.62 #define XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY -0x9000

Definition at line 32 of file ssl.h.

Referenced by main(), ssl_read_record(), and ssl_test().

13.112.2.63 #define XYSSL_ERR_SSL_PEER_VERIFY_FAILED -0x8800

Definition at line 31 of file ssl.h.

13.112.2.64 #define XYSSL_ERR_SSL_PRIVATE_KEY_REQUIRED -0x6800

Definition at line 27 of file ssl.h.

Referenced by ssl_write_certificate_verify().

13.112.2.65 #define XYSSL_ERR_SSL_UNEXPECTED_MESSAGE -0x7800

Definition at line 29 of file ssl.h.

Referenced by ssl_parse_certificate(), ssl_parse_certificate_request(), ssl_parse_change_cipher_spec(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_parse_server_hello_done(), ssl_parse_server_key_exchange(), and ssl_read().

13.112.2.66 #define XYSSL_ERR_SSL_UNKNOWN_CIPHER -0x3800

Definition at line 21 of file ssl.h.

13.112.3 Typedef Documentation

13.112.3.1 typedef struct _ssl_context ssl_context

Definition at line 134 of file ssl.h.

13.112.3.2 typedef struct _ssl_session ssl_session

Definition at line 133 of file ssl.h.

13.112.4 Enumeration Type Documentation

13.112.4.1 enum ssl_states

Enumerator:

SSL_HELLO_REQUEST
SSL_CLIENT_HELLO
SSL_SERVER_HELLO
SSL_SERVER_CERTIFICATE
SSL_SERVER_KEY_EXCHANGE
SSL_CERTIFICATE_REQUEST
SSL_SERVER_HELLO_DONE
SSL_CLIENT_CERTIFICATE
SSL_CLIENT_KEY_EXCHANGE
SSL_CERTIFICATE_VERIFY
SSL_CLIENT_CHANGE_CIPHER_SPEC
SSL_CLIENT_FINISHED
SSL_SERVER_CHANGE_CIPHER_SPEC
SSL_SERVER_FINISHED
SSL_FLUSH_BUFFERS
SSL_HANDSHAKE_OVER

Definition at line 112 of file ssl.h.

13.112.5 Function Documentation

13.112.5.1 void ssl_calc_verify (ssl_context * *ssl*, unsigned char *hash*[36])

Definition at line 335 of file ssl_tls.c.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_session::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::session`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, and `SSL_MINOR_VERSION_0`.

Referenced by `ssl_parse_certificate_verify()`, and `ssl_write_certificate_verify()`.

13.112.5.2 int ssl_close_notify (ssl_context * ssl)

Notify the peer that the connection is being closed.

Definition at line 1904 of file ssl_tls.c.

References `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_WARNING`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_ALERT`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.3 int ssl_derive_keys (ssl_context * ssl)

Definition at line 106 of file ssl_tls.c.

References `aes_setkey_dec()`, `aes_setkey_enc()`, `arc4_setup()`, `_ssl_session::cipher`, `_ssl_context::ctx_dec`, `_ssl_context::ctx_enc`, `des3_set3key_dec()`, `des3_set3key_enc()`, `_ssl_context::endpoint`, `_ssl_context::iv_dec`, `_ssl_context::iv_enc`, `_ssl_context::ivlen`, `_ssl_context::keylen`, `_ssl_context::mac_dec`, `_ssl_context::mac_enc`, `_ssl_context::maclen`, `_ssl_session::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `_ssl_context::pmslen`, `_ssl_context::premaster`, `_ssl_context::randbytes`, `_ssl_context::resume`, `_ssl_context::session`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_EDH_RSA_AES_256_SHA`, `SSL_EDH_RSA_DES_168_SHA`, `ssl_get_cipher()`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `SSL_RSA_AES_128_SHA`, `SSL_RSA_AES_256_SHA`, `SSL_RSA_DES_168_SHA`, `SSL_RSA_RC4_128_MD5`, `SSL_RSA_RC4_128_SHA`, `tls1_prf()`, and `XYSSL_ERR_SSL_FEATURE_UNAVAILABLE`.

Referenced by `ssl_parse_client_key_exchange()`, `ssl_parse_server_hello()`, `ssl_write_client_key_exchange()`, and `ssl_write_server_hello()`.

13.112.5.4 int ssl_fetch_input (ssl_context * ssl, int nb_want)

Definition at line 727 of file ssl_tls.c.

References `_ssl_context::f_recv`, `_ssl_context::in_hdr`, `_ssl_context::in_left`, `_ssl_context::p_recv`, `SSL_DEBUG_MSG`, and `SSL_DEBUG_RET`.

Referenced by `ssl_parse_client_hello()`, and `ssl_read_record()`.

13.112.5.5 int ssl_flush_output (ssl_context * ssl)

Definition at line 756 of file ssl_tls.c.

References `buf`, `_ssl_context::f_send`, `_ssl_context::out_hdr`, `_ssl_context::out_left`, `_ssl_context::out_msglen`, `_ssl_context::p_send`, `SSL_DEBUG_MSG`, and `SSL_DEBUG_RET`.

Referenced by `ssl_close_notify()`, `ssl_handshake_client()`, `ssl_handshake_server()`, `ssl_write()`, and `ssl_write_record()`.

13.112.5.6 void ssl_free (ssl_context * ssl)

Free an SSL context.

Definition at line 1938 of file ssl_tls.c.

References `_ssl_context::dhm_ctx`, `dhm_free()`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::out_ctr`, `_ssl_context::peer_cert`, `SSL_BUFFER_LEN`, `SSL_DEBUG_MSG`, and `x509_free()`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.7 `int ssl_get_bytes_avail (ssl_context * ssl)`

Return the number of data bytes available to read.

Parameters:

ssl SSL context

Returns:

how many bytes are available in the read buffer

Definition at line 1691 of file `ssl_tls.c`.

References `_ssl_context::in_msglen`, and `_ssl_context::in_offt`.

13.112.5.8 `char* ssl_get_cipher (ssl_context * ssl)`

Return the name of the current cipher.

Parameters:

ssl SSL context

Returns:

a string containing the cipher name

Definition at line 1701 of file `ssl_tls.c`.

References `_ssl_session::cipher`, `_ssl_context::session`, `SSL_EDH_RSA_AES_256_SHA`, `SSL_EDH_RSA_DES_168_SHA`, `SSL_RSA_AES_128_SHA`, `SSL_RSA_AES_256_SHA`, `SSL_RSA_DES_168_SHA`, `SSL_RSA_RC4_128_MD5`, and `SSL_RSA_RC4_128_SHA`.

Referenced by `main()`, and `ssl_derive_keys()`.

13.112.5.9 `int ssl_get_verify_result (ssl_context * ssl)`

Return the result of the certificate verification.

Parameters:

ssl SSL context

Returns:

0 if successful, or a combination of: `BADCERT_EXPIRED` `BADCERT_REVOKED` `BADCERT_CN_MISMATCH` `BADCERT_NOT_TRUSTED`

Definition at line 1696 of file `ssl_tls.c`.

References `_ssl_context::verify_result`.

Referenced by `main()`.

13.112.5.10 int ssl_handshake (ssl_context * ssl)

Perform the SSL handshake.

Parameters:

ssl SSL context

Returns:

0 if successful, XYSSL_ERR_NET_TRY_AGAIN, or a specific SSL error code.

Definition at line 1767 of file ssl_tls.c.

References `_ssl_context::endpoint`, `SSL_DEBUG_MSG`, `ssl_handshake_client()`, `ssl_handshake_server()`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, and `XYSSL_ERR_SSL_FEATURE_UNAVAILABLE`.

Referenced by `main()`, `ssl_read()`, and `ssl_write()`.

13.112.5.11 int ssl_handshake_client (ssl_context * ssl)

Definition at line 660 of file ssl_cli.c.

References `SSL_CERTIFICATE_REQUEST`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CERTIFICATE`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`, `SSL_CLIENT_HELLO`, `SSL_CLIENT_KEY_EXCHANGE`, `SSL_DEBUG_MSG`, `SSL_FLUSH_BUFFERS`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_HELLO_REQUEST`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `SSL_SERVER_CERTIFICATE`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_SERVER_HELLO`, `SSL_SERVER_HELLO_DONE`, `SSL_SERVER_KEY_EXCHANGE`, `ssl_write_certificate()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_handshake()`.

13.112.5.12 int ssl_handshake_server (ssl_context * ssl)

Definition at line 819 of file ssl_srv.c.

References `SSL_CERTIFICATE_REQUEST`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CERTIFICATE`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`, `SSL_CLIENT_HELLO`, `SSL_CLIENT_KEY_EXCHANGE`, `SSL_DEBUG_MSG`, `SSL_FLUSH_BUFFERS`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_HELLO_REQUEST`, `ssl_parse_certificate()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_hello()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `SSL_SERVER_CERTIFICATE`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_SERVER_HELLO`, `SSL_SERVER_HELLO_DONE`, `SSL_SERVER_KEY_EXCHANGE`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_change_cipher_spec()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, `ssl_write_server_key_exchange()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_handshake()`.

13.112.5.13 int ssl_init (ssl_context * ssl)

Initialize an SSL context.

Parameters:

ssl SSL context

Returns:

0 if successful, or 1 if memory allocation failed

Definition at line 1542 of file ssl_tls.c.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::in_hdr`, `_ssl_context::in_msg`, `md5_starts()`, `_ssl_context::out_ctr`, `_ssl_context::out_hdr`, `_ssl_context::out_msg`, `sha1_starts()`, `SSL_BUFFER_LEN`, and `SSL_DEBUG_MSG`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.14 `int ssl_parse_certificate(ssl_context * ssl)`

Definition at line 1140 of file ssl_tls.c.

References `_ssl_context::authmode`, `_ssl_context::ca_chain`, `_ssl_context::endpoint`, `_ssl_context::in_hslen`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `int`, `_ssl_context::minor_ver`, `_ssl_context::peer_cert`, `_ssl_context::peer_cn`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_SERVER`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `SSL_VERIFY_NONE`, `SSL_VERIFY_OPTIONAL`, `SSL_VERIFY_REQUIRED`, `_ssl_context::state`, `_ssl_context::verify_result`, `x509parse_cert()`, `x509parse_verify()`, `XYSSL_ERR_SSL_BAD_HS_CERTIFICATE`, `XYSSL_ERR_SSL_CA_CHAIN_REQUIRED`, `XYSSL_ERR_SSL_NO_CLIENT_CERTIFICATE`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.112.5.15 `int ssl_parse_change_cipher_spec(ssl_context * ssl)`

Definition at line 1311 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_read_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.112.5.16 `int ssl_parse_finished(ssl_context * ssl)`

Definition at line 1480 of file ssl_tls.c.

References `buf`, `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hslen`, `_ssl_context::in_msg`, `_ssl_context::in_msgtype`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_BAD_HS_FINISHED`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.112.5.17 int ssl_read (ssl_context * ssl, unsigned char * buf, int len)

Read at most 'len' application data bytes.

Parameters:

ssl SSL context
buf buffer that will hold the data
len how many bytes must be read

Returns:

This function returns the number of bytes read, or a negative error code.

Definition at line 1791 of file ssl_tls.c.

References `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::in_offt`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_APPLICATION_DATA`, `ssl_read_record()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.18 int ssl_read_record (ssl_context * ssl)

Definition at line 842 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hdr`, `_ssl_context::in_hhlen`, `_ssl_context::in_left`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `sha1_update()`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_FATAL`, `SSL_ALERT_WARNING`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_decrypt_buf()`, `ssl_fetch_input()`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MINOR_VERSION_1`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE`, `XYSSL_ERR_SSL_INVALID_RECORD`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

Referenced by `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, and `ssl_read()`.

13.112.5.19 void ssl_set_authmode (ssl_context * ssl, int authmode)

Set the certificate verification mode.

Parameters:

ssl SSL context
mode can be:

`SSL_VERIFY_NONE`: peer certificate is not checked (default), this is insecure and SHOULD be avoided.

`SSL_VERIFY_OPTIONAL`: peer certificate is checked, however the handshake continues even if verification failed; `ssl_get_verify_result()` can be called after the handshake is complete.

`SSL_VERIFY_REQUIRED`: peer *must* present a valid certificate, handshake is aborted if verification failed.

Definition at line 1589 of file ssl_tls.c.

References `_ssl_context::authmode`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.20 `void ssl_set_bio (ssl_context * ssl, int(*) (void *, unsigned char *, int) f_recv, void * p_recv, int(*) (void *, unsigned char *, int) f_send, void * p_send)`

Set the underlying BIO read and write callbacks.

Parameters:

ssl SSL context

f_recv read callback

p_recv read parameter

f_send write callback

p_send write parameter

Definition at line 1610 of file ssl_tls.c.

References `_ssl_context::f_recv`, `_ssl_context::f_send`, `_ssl_context::p_recv`, and `_ssl_context::p_send`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.21 `void ssl_set_ca_chain (ssl_context * ssl, x509_cert * ca_chain, char * peer_cn)`

Set the data required to verify peer certificate.

Parameters:

ssl SSL context

ca_chain trusted CA chain

peer_cn expected peer CommonName (or NULL)

Note:

TODO: add two more parameters: depth and crl

Definition at line 1641 of file ssl_tls.c.

References `_ssl_context::ca_chain`, and `_ssl_context::peer_cn`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.22 `void ssl_set_ciphers (ssl_context * ssl, int * ciphers)`

Set the list of allowed ciphersuites.

Parameters:

ssl SSL context

ciphers 0-terminated list of allowed ciphers

Definition at line 1636 of file `ssl_tls.c`.

References `_ssl_context::ciphers`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.23 `void ssl_set_dbg (ssl_context * ssl, void(*)(void *, int, char *) f_dbg, void * p_dbg)`

Set the debug callback.

Parameters:

ssl SSL context

f_dbg debug function

p_dbg debug parameter

Definition at line 1602 of file `ssl_tls.c`.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.24 `int ssl_set_dh_param (ssl_context * ssl, char * dhm_P, char * dhm_G)`

Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).

Parameters:

ssl SSL context

dhm_P Diffie-Hellman-Merkle modulus

dhm_G Diffie-Hellman-Merkle generator

Returns:

0 if successful

Definition at line 1655 of file `ssl_tls.c`.

References `_ssl_context::dhm_ctx`, `dhm_context::G`, `mpi_read_string()`, `dhm_context::P`, and `SSL_DEBUG_RET`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.25 `void ssl_set_endpoint (ssl_context * ssl, int endpoint)`

Set the current endpoint type.

Parameters:

ssl SSL context

endpoint must be `SSL_IS_CLIENT` or `SSL_IS_SERVER`

Definition at line 1584 of file `ssl_tls.c`.

References `_ssl_context::endpoint`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.26 `int ssl_set_hostname (ssl_context * ssl, char * hostname)`

Set hostname for ServerName TLS Extension.

Parameters:

ssl SSL context
hostname the server hostname

Returns:

0 if successful

Definition at line 1674 of file ssl_tls.c.

References `_ssl_context::hostname`, `_ssl_context::hostname_len`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `main()`.

13.112.5.27 `void ssl_set_own_cert (ssl_context * ssl, x509_cert * own_cert, rsa_context * rsa_key)`

Set own certificate and private key.

Parameters:

ssl SSL context
own_cert own public certificate
rsa_key own private RSA key

Definition at line 1648 of file ssl_tls.c.

References `_ssl_context::own_cert`, and `_ssl_context::rsa_key`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.28 `void ssl_set_rng (ssl_context * ssl, int(*) (void *) f_rng, void * p_rng)`

Set the random number generator callback.

Parameters:

ssl SSL context
f_rng RNG function
p_rng RNG parameter

Definition at line 1594 of file ssl_tls.c.

References `_ssl_context::f_rng`, and `_ssl_context::p_rng`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.29 `void ssl_set_scb (ssl_context * ssl, int(*) (ssl_context *) s_get, int(*) (ssl_context *) s_set)`

Set the session callbacks (server-side only).

Parameters:

ssl SSL context
s_get session get callback
s_set session set callback

Definition at line 1620 of file ssl_tls.c.

References `_ssl_context::s_get`, and `_ssl_context::s_set`.

Referenced by `main()`.

13.112.5.30 void ssl_set_session (ssl_context * ssl, int resume, int timeout, ssl_session * session)

Set the session resuming flag, timeout and data.

Parameters:

ssl SSL context
resume if 0 (default), the session will not be resumed
timeout session timeout in seconds, or 0 (no timeout)
session session context

Definition at line 1628 of file ssl_tls.c.

References `_ssl_context::resume`, `_ssl_context::session`, and `_ssl_context::timeout`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.31 int ssl_write (ssl_context * ssl, unsigned char * buf, int len)

Write exactly 'len' application data bytes.

Parameters:

ssl SSL context
buf buffer holding the data
len how many bytes must be written

Returns:

This function returns the number of bytes written, or a negative error code.

Note:

When this function returns `XYSSL_ERR_NET_TRY_AGAIN`, it must be called later with the **same** arguments, until it returns a positive value.

Definition at line 1857 of file ssl_tls.c.

References `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MAX_CONTENT_LEN`, `SSL_MSG_APPLICATION_DATA`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

13.112.5.32 int ssl_write_certificate (ssl_context * ssl)

Definition at line 1044 of file ssl_tls.c.

References `_ssl_context::client_auth`, `_ssl_context::endpoint`, `_x509_buf::len`, `_ssl_context::minor_ver`, `_x509_cert::next`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::own_cert`, `_x509_buf::p`, `_x509_cert::raw`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_CLIENT`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_CERTIFICATE_REQUIRED`, and `XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.112.5.33 int ssl_write_change_cipher_spec (ssl_context * ssl)

Definition at line 1287 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.112.5.34 int ssl_write_finished (ssl_context * ssl)

Definition at line 1433 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.112.5.35 int ssl_write_record (ssl_context * ssl)

Definition at line 786 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::out_hdr`, `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_encrypt_buf()`, `ssl_flush_output()`, and `SSL_MSG_HANDSHAKE`.

Referenced by `ssl_close_notify()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

13.112.6 Variable Documentation

13.112.6.1 `int ssl_default_ciphers[]`

Definition at line 1739 of file `ssl_tls.c`.

Referenced by `main()`, and `ssl_test()`.

13.113 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/timing.h File Reference

Data Structures

- struct [hr_time](#)
timer structure

Functions

- unsigned long [hardclock](#) (void)
Return the CPU cycle counter value.
- unsigned long [get_timer](#) (struct [hr_time](#) *val, int reset)
Return the elapsed time in milliseconds.
- void [set_alarm](#) (int seconds)
Setup an alarm clock.
- void [m_sleep](#) (int milliseconds)
Sleep for a certain amount of time.

Variables

- int [alarmed](#)

13.113.1 Detailed Description

Definition in file [timing.h](#).

13.113.2 Function Documentation

13.113.2.1 unsigned long get_timer (struct hr_time * val, int reset)

Return the elapsed time in milliseconds.

Parameters:

- val* points to a timer structure
- reset* if set to 1, the timer is restarted

Definition at line 205 of file [timing.c](#).

References [_hr_time::start](#).

Referenced by [ssl_test\(\)](#).

13.113.2.2 unsigned long hardclock (void)

Return the CPU cycle counter value.

Definition at line 136 of file timing.c.

References tv_init.

Referenced by main().

13.113.2.3 void m_sleep (int *milliseconds*)

Sleep for a certain amount of time.

Definition at line 238 of file timing.c.

13.113.2.4 void set_alarm (int *seconds*)

Setup an alarm clock.

Parameters:

seconds delay before the "alarmed" flag is set

Definition at line 231 of file timing.c.

References sighandler().

Referenced by main().

13.113.3 Variable Documentation**13.113.3.1 int alarmed**

Definition at line 159 of file timing.c.

Referenced by main().

13.114 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/include/xyssl/x509.h File Reference

```
#include "xyssl/rsa.h"
```

Data Structures

- struct [_x509_buf](#)
- struct [_x509_name](#)
- struct [_x509_time](#)
- struct [_x509_cert](#)
- struct [_x509_node](#)
- struct [_x509_raw](#)

Defines

- #define [XYSSL_ERR_ASN1_OUT_OF_DATA](#) -0x0014
- #define [XYSSL_ERR_ASN1_UNEXPECTED_TAG](#) -0x0016
- #define [XYSSL_ERR_ASN1_INVALID_LENGTH](#) -0x0018
- #define [XYSSL_ERR_ASN1_LENGTH_MISMATCH](#) -0x001A
- #define [XYSSL_ERR_ASN1_INVALID_DATA](#) -0x001C
- #define [XYSSL_ERR_X509_FEATURE_UNAVAILABLE](#) -0x0020
- #define [XYSSL_ERR_X509_CERT_INVALID_PEM](#) -0x0040
- #define [XYSSL_ERR_X509_CERT_INVALID_FORMAT](#) -0x0060
- #define [XYSSL_ERR_X509_CERT_INVALID_VERSION](#) -0x0080
- #define [XYSSL_ERR_X509_CERT_INVALID_SERIAL](#) -0x00A0
- #define [XYSSL_ERR_X509_CERT_INVALID_ALG](#) -0x00C0
- #define [XYSSL_ERR_X509_CERT_INVALID_NAME](#) -0x00E0
- #define [XYSSL_ERR_X509_CERT_INVALID_DATE](#) -0x0100
- #define [XYSSL_ERR_X509_CERT_INVALID_PUBKEY](#) -0x0120
- #define [XYSSL_ERR_X509_CERT_INVALID_SIGNATURE](#) -0x0140
- #define [XYSSL_ERR_X509_CERT_INVALID_EXTENSIONS](#) -0x0160
- #define [XYSSL_ERR_X509_CERT_UNKNOWN_VERSION](#) -0x0180
- #define [XYSSL_ERR_X509_CERT_UNKNOWN_SIG_ALG](#) -0x01A0
- #define [XYSSL_ERR_X509_CERT_UNKNOWN_PK_ALG](#) -0x01C0
- #define [XYSSL_ERR_X509_CERT_SIG_MISMATCH](#) -0x01E0
- #define [XYSSL_ERR_X509_CERT_VERIFY_FAILED](#) -0x0200
- #define [XYSSL_ERR_X509_KEY_INVALID_PEM](#) -0x0220
- #define [XYSSL_ERR_X509_KEY_INVALID_VERSION](#) -0x0240
- #define [XYSSL_ERR_X509_KEY_INVALID_FORMAT](#) -0x0260
- #define [XYSSL_ERR_X509_KEY_INVALID_ENC_IV](#) -0x0280
- #define [XYSSL_ERR_X509_KEY_UNKNOWN_ENC_ALG](#) -0x02A0
- #define [XYSSL_ERR_X509_KEY_PASSWORD_REQUIRED](#) -0x02C0
- #define [XYSSL_ERR_X509_KEY_PASSWORD_MISMATCH](#) -0x02E0
- #define [XYSSL_ERR_X509_POINT_ERROR](#) -0x0300
- #define [XYSSL_ERR_X509_VALUE_TO_LENGTH](#) -0x0320
- #define [BADCERT_EXPIRED](#) 1
- #define [BADCERT_REVOKED](#) 2
- #define [BADCERT_CN_MISMATCH](#) 4

- #define [BADCERT_NOT_TRUSTED](#) 8
- #define [ASN1_BOOLEAN](#) 0x01
- #define [ASN1_INTEGER](#) 0x02
- #define [ASN1_BIT_STRING](#) 0x03
- #define [ASN1_OCTET_STRING](#) 0x04
- #define [ASN1_NULL](#) 0x05
- #define [ASN1_OID](#) 0x06
- #define [ASN1_UTF8_STRING](#) 0x0C
- #define [ASN1_SEQUENCE](#) 0x10
- #define [ASN1_SET](#) 0x11
- #define [ASN1_PRINTABLE_STRING](#) 0x13
- #define [ASN1_T61_STRING](#) 0x14
- #define [ASN1_IA5_STRING](#) 0x16
- #define [ASN1_UTC_TIME](#) 0x17
- #define [ASN1_UNIVERSAL_STRING](#) 0x1C
- #define [ASN1_BMP_STRING](#) 0x1E
- #define [ASN1_PRIMITIVE](#) 0x00
- #define [ASN1_CONSTRUCTED](#) 0x20
- #define [ASN1_CONTEXT_SPECIFIC](#) 0x80
- #define [X520_COMMON_NAME](#) 3
- #define [X520_COUNTRY](#) 6
- #define [X520_LOCALITY](#) 7
- #define [X520_STATE](#) 8
- #define [X520_ORGANIZATION](#) 10
- #define [X520_ORG_UNIT](#) 11
- #define [PKCS9_EMAIL](#) 1
- #define [X509_OUTPUT_DER](#) 0x01
- #define [X509_OUTPUT_PEM](#) 0x02
- #define [PEM_LINE_LENGTH](#) 72
- #define [X509_ISSUER](#) 0x01
- #define [X509_SUBJECT](#) 0x02
- #define [OID_X520](#) "\x55\x04"
- #define [OID_CN](#) "\x55\x04\x03"
- #define [OID_PKCS1](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x01"
- #define [OID_PKCS1_RSA](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x01"
- #define [OID_PKCS1_RSA_SHA](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x05"
- #define [OID_PKCS9](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x09"
- #define [OID_PKCS9_EMAIL](#) "\x2A\x86\x48\x86\xF7\x0D\x01\x09\x01"

Typedefs

- typedef struct [_x509_buf](#) [x509_buf](#)
- typedef struct [_x509_name](#) [x509_name](#)
- typedef struct [_x509_time](#) [x509_time](#)
- typedef struct [_x509_cert](#) [x509_cert](#)
- typedef struct [_x509_node](#) [x509_node](#)
- typedef struct [_x509_raw](#) [x509_raw](#)

Functions

- `int x509parse_cert(x509_cert *crt, unsigned char *buf, int buflen)`
Parse one or more certificates and add them to the chained list.
- `int x509parse_crtfile(x509_cert *crt, char *path)`
Load one or more certificates and add them to the chained list.
- `int x509parse_key(rsa_context *rsa, unsigned char *buf, int buflen, unsigned char *pwd, int pwrlen)`
Parse a private RSA key.
- `int x509parse_keyfile(rsa_context *rsa, char *path, char *password)`
Load and parse a private RSA key.
- `int x509parse_dn_gets(char *buf, char *end, x509_name *dn)`
Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.
- `char * x509parse_cert_info(char *prefix, x509_cert *crt)`
Returns an informational string about the certificate.
- `int x509parse_expired(x509_cert *crt)`
Return 0 if the certificate is still valid, or BADCERT_EXPIRED.
- `int x509parse_verify(x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)`
Verify the certificate signature.
- `void x509_free(x509_cert *crt)`
Unallocate all certificate data.
- `int x509_self_test(int verbose)`
Checkup routine.

13.114.1 Detailed Description

Definition in file [x509.h](#).

13.114.2 Define Documentation

13.114.2.1 `#define ASN1_BIT_STRING 0x03`

Definition at line 51 of file [x509.h](#).

Referenced by [x509_get_pubkey\(\)](#), and [x509_get_sig\(\)](#).

13.114.2.2 `#define ASN1_BMP_STRING 0x1E`

Definition at line 63 of file [x509.h](#).

Referenced by [x509_get_name\(\)](#).

13.114.2.3 #define ASN1_BOOLEAN 0x01

Definition at line 49 of file x509.h.

Referenced by `asn1_get_bool()`.

13.114.2.4 #define ASN1_CONSTRUCTED 0x20

Definition at line 65 of file x509.h.

Referenced by `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_uid()`, `x509_get_version()`, `x509parse_crt()`, and `x509parse_key()`.

13.114.2.5 #define ASN1_CONTEXT_SPECIFIC 0x80

Definition at line 66 of file x509.h.

Referenced by `x509_get_ext()`, `x509_get_serial()`, `x509_get_uid()`, and `x509_get_version()`.

13.114.2.6 #define ASN1_IA5_STRING 0x16

Definition at line 60 of file x509.h.

Referenced by `x509_get_name()`.

13.114.2.7 #define ASN1_INTEGER 0x02

Definition at line 50 of file x509.h.

Referenced by `asn1_get_int()`, `asn1_get_mpi()`, and `x509_get_serial()`.

13.114.2.8 #define ASN1_NULL 0x05

Definition at line 53 of file x509.h.

Referenced by `x509_get_alg()`.

13.114.2.9 #define ASN1_OCTET_STRING 0x04

Definition at line 52 of file x509.h.

Referenced by `x509_get_ext()`.

13.114.2.10 #define ASN1_OID 0x06

Definition at line 54 of file x509.h.

Referenced by `x509_get_alg()`, and `x509_get_name()`.

13.114.2.11 #define ASN1_PRIMITIVE 0x00

Definition at line 64 of file x509.h.

Referenced by x509_get_serial().

13.114.2.12 #define ASN1_PRINTABLE_STRING 0x13

Definition at line 58 of file x509.h.

Referenced by x509_get_name().

13.114.2.13 #define ASN1_SEQUENCE 0x10

Definition at line 56 of file x509.h.

Referenced by x509_get_alg(), x509_get_dates(), x509_get_ext(), x509_get_name(), x509_get_pubkey(), x509parse_crt(), and x509parse_key().

13.114.2.14 #define ASN1_SET 0x11

Definition at line 57 of file x509.h.

Referenced by x509_get_name().

13.114.2.15 #define ASN1_T61_STRING 0x14

Definition at line 59 of file x509.h.

Referenced by x509_get_name().

13.114.2.16 #define ASN1_UNIVERSAL_STRING 0x1C

Definition at line 62 of file x509.h.

Referenced by x509_get_name().

13.114.2.17 #define ASN1.UTC_TIME 0x17

Definition at line 61 of file x509.h.

Referenced by x509_get_dates().

13.114.2.18 #define ASN1_UTF8_STRING 0x0C

Definition at line 55 of file x509.h.

Referenced by x509_get_name().

13.114.2.19 #define BADCERT_CN_MISMATCH 4

Definition at line 43 of file x509.h.

Referenced by main(), and x509parse_verify().

13.114.2.20 #define BADCERT_EXPIRED 1

Definition at line 41 of file x509.h.

Referenced by main(), and x509parse_expired().

13.114.2.21 #define BADCERT_NOT_TRUSTED 8

Definition at line 44 of file x509.h.

Referenced by main(), and x509parse_verify().

13.114.2.22 #define BADCERT_REVOKED 2

Definition at line 42 of file x509.h.

Referenced by main().

13.114.2.23 #define OID_CN "\x55\x04\x03"

Definition at line 86 of file x509.h.

Referenced by x509parse_verify().

13.114.2.24 #define OID_PKCS1 "\x2A\x86\x48\x86\xF7\x0D\x01\x01"

Definition at line 87 of file x509.h.

Referenced by x509parse_crt().

13.114.2.25 #define OID_PKCS1_RSA "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x01"

Definition at line 88 of file x509.h.

Referenced by x509_get_pubkey().

13.114.2.26 #define OID_PKCS1_RSA_SHA "\x2A\x86\x48\x86\xF7\x0D\x01\x01\x05"

Definition at line 89 of file x509.h.

13.114.2.27 #define OID_PKCS9 "\x2A\x86\x48\x86\xF7\x0D\x01\x09"

Definition at line 90 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.28 #define OID_PKCS9_EMAIL "\x2A\x86\x48\x86\xF7\x0D\x01\x09\x01"

Definition at line 91 of file x509.h.

13.114.2.29 #define OID_X520 "\x55\x04"

Definition at line 85 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.30 #define PEM_LINE_LENGTH 72

Definition at line 81 of file x509.h.

13.114.2.31 #define PKCS9_EMAIL 1

Definition at line 77 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.32 #define X509_ISSUER 0x01

Definition at line 82 of file x509.h.

13.114.2.33 #define X509_OUTPUT_DER 0x01

Definition at line 79 of file x509.h.

13.114.2.34 #define X509_OUTPUT_PEM 0x02

Definition at line 80 of file x509.h.

13.114.2.35 #define X509_SUBJECT 0x02

Definition at line 83 of file x509.h.

13.114.2.36 #define X520_COMMON_NAME 3

Definition at line 71 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.37 #define X520_COUNTRY 6

Definition at line 72 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.38 #define X520_LOCALITY 7

Definition at line 73 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.39 #define X520_ORG_UNIT 11

Definition at line 76 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.40 #define X520_ORGANIZATION 10

Definition at line 75 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.41 #define X520_STATE 8

Definition at line 74 of file x509.h.

Referenced by x509parse_dn_gets().

13.114.2.42 #define XYSSL_ERR_ASN1_INVALID_DATA -0x001C

Definition at line 13 of file x509.h.

13.114.2.43 #define XYSSL_ERR_ASN1_INVALID_LENGTH -0x0018

Definition at line 11 of file x509.h.

Referenced by asn1_get_bool(), asn1_get_int(), and asn1_get_len().

13.114.2.44 #define XYSSL_ERR_ASN1_LENGTH_MISMATCH -0x001A

Definition at line 12 of file x509.h.

Referenced by x509_get_alg(), x509_get_dates(), x509_get_ext(), x509_get_name(), x509_get_pubkey(), x509_get_version(), x509parse_crt(), and x509parse_key().

13.114.2.45 #define XYSSL_ERR_ASN1_OUT_OF_DATA -0x0014

Definition at line 9 of file x509.h.

Referenced by asn1_get_len(), asn1_get_tag(), x509_get_name(), x509_get_pubkey(), and x509_get_serial().

13.114.2.46 #define XYSSL_ERR_ASN1_UNEXPECTED_TAG -0x0016

Definition at line 10 of file x509.h.

Referenced by asn1_get_tag(), x509_get_ext(), x509_get_name(), x509_get_serial(), x509_get_uid(), and x509_get_version().

13.114.2.47 #define XYSSL_ERR_X509_CERT_INVALID_ALG -0x00C0

Definition at line 20 of file x509.h.

Referenced by x509_get_alg().

13.114.2.48 #define XYSSL_ERR_X509_CERT_INVALID_DATE -0x0100

Definition at line 22 of file x509.h.

Referenced by x509_get_dates().

13.114.2.49 #define XYSSL_ERR_X509_CERT_INVALID_EXTENSIONS -0x0160

Definition at line 25 of file x509.h.

Referenced by x509_get_ext().

13.114.2.50 #define XYSSL_ERR_X509_CERT_INVALID_FORMAT -0x0060

Definition at line 17 of file x509.h.

Referenced by x509parse_crt().

13.114.2.51 #define XYSSL_ERR_X509_CERT_INVALID_NAME -0x00E0

Definition at line 21 of file x509.h.

Referenced by x509_get_name().

13.114.2.52 #define XYSSL_ERR_X509_CERT_INVALID_PEM -0x0040

Definition at line 16 of file x509.h.

Referenced by x509parse_crt().

13.114.2.53 #define XYSSL_ERR_X509_CERT_INVALID_PUBKEY -0x0120

Definition at line 23 of file x509.h.

Referenced by x509_get_pubkey().

13.114.2.54 #define XYSSL_ERR_X509_CERT_INVALID_SERIAL -0x00A0

Definition at line 19 of file x509.h.

Referenced by x509_get_serial().

13.114.2.55 #define XYSSL_ERR_X509_CERT_INVALID_SIGNATURE -0x0140

Definition at line 24 of file x509.h.

Referenced by x509_get_sig().

13.114.2.56 #define XYSSL_ERR_X509_CERT_INVALID_VERSION -0x0080

Definition at line 18 of file x509.h.

Referenced by x509_get_version().

13.114.2.57 #define XYSSL_ERR_X509_CERT_SIG_MISMATCH -0x01E0

Definition at line 29 of file x509.h.

Referenced by x509parse_crt().

13.114.2.58 #define XYSSL_ERR_X509_CERT_UNKNOWN_PK_ALG -0x01C0

Definition at line 28 of file x509.h.

Referenced by x509_get_pubkey().

13.114.2.59 #define XYSSL_ERR_X509_CERT_UNKNOWN_SIG_ALG -0x01A0

Definition at line 27 of file x509.h.

Referenced by x509parse_crt().

13.114.2.60 #define XYSSL_ERR_X509_CERT_UNKNOWN_VERSION -0x0180

Definition at line 26 of file x509.h.

Referenced by x509parse_crt().

13.114.2.61 #define XYSSL_ERR_X509_CERT_VERIFY_FAILED -0x0200

Definition at line 30 of file x509.h.

Referenced by x509parse_verify().

13.114.2.62 #define XYSSL_ERR_X509_FEATURE_UNAVAILABLE -0x0020

Definition at line 15 of file x509.h.

Referenced by x509parse_key().

13.114.2.63 #define XYSSL_ERR_X509_KEY_INVALID_ENC_IV -0x0280

Definition at line 34 of file x509.h.

Referenced by x509_get_iv(), and x509parse_key().

13.114.2.64 #define XYSSL_ERR_X509_KEY_INVALID_FORMAT -0x0260

Definition at line 33 of file x509.h.

Referenced by x509parse_key().

13.114.2.65 #define XYSSL_ERR_X509_KEY_INVALID_PEM -0x0220

Definition at line 31 of file x509.h.

Referenced by x509parse_key().

13.114.2.66 #define XYSSL_ERR_X509_KEY_INVALID_VERSION -0x0240

Definition at line 32 of file x509.h.

Referenced by x509parse_key().

13.114.2.67 #define XYSSL_ERR_X509_KEY_PASSWORD_MISMATCH -0x02E0

Definition at line 37 of file x509.h.

Referenced by x509parse_key().

13.114.2.68 #define XYSSL_ERR_X509_KEY_PASSWORD_REQUIRED -0x02C0

Definition at line 36 of file x509.h.

Referenced by x509parse_key().

13.114.2.69 #define XYSSL_ERR_X509_KEY_UNKNOWN_ENC_ALG -0x02A0

Definition at line 35 of file x509.h.

Referenced by x509parse_key().

13.114.2.70 #define XYSSL_ERR_X509_POINT_ERROR -0x0300

Definition at line 38 of file x509.h.

13.114.2.71 #define XYSSL_ERR_X509_VALUE_TO_LENGTH -0x0320

Definition at line 39 of file x509.h.

13.114.3 Typedef Documentation

13.114.3.1 `typedef struct _x509_buf x509_buf`

13.114.3.2 `typedef struct _x509_cert x509_cert`

13.114.3.3 `typedef struct _x509_name x509_name`

13.114.3.4 `typedef struct _x509_node x509_node`

13.114.3.5 `typedef struct _x509_raw x509_raw`

13.114.3.6 `typedef struct _x509_time x509_time`

13.114.4 Function Documentation

13.114.4.1 `void x509_free (x509_cert * crt)`

Unallocate all certificate data.

Definition at line 1613 of file x509parse.c.

References `_x509_cert::issuer`, `_x509_buf::len`, `_x509_cert::next`, `_x509_name::next`, `_x509_buf::p`, `_x509_cert::raw`, `_x509_cert::rsa`, `rsa_free()`, and `_x509_cert::subject`.

Referenced by `main()`, `ssl_free()`, `ssl_test()`, `x509_self_test()`, and `x509parse_cert()`.

13.114.4.2 `int x509_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1675 of file x509parse.c.

References `rsa_free()`, `test_ca_cert`, `test_ca_key`, `test_ca_pwd`, `test_cli_cert`, `x509_free()`, `x509parse_cert()`, `x509parse_key()`, and `x509parse_verify()`.

Referenced by `main()`.

13.114.4.3 `char* x509parse_cert_info (char * prefix, x509_cert * crt)`

Returns an informational string about the certificate.

Definition at line 1399 of file x509parse.c.

References `buf`, `_x509_time::day`, `_x509_time::hour`, `_x509_cert::issuer`, `_x509_buf::len`, `_x509_time::min`, `_x509_time::mon`, `mpi::n`, `rsa_context::N`, `_x509_buf::p`, `_x509_cert::rsa`, `RSA_MD2`, `RSA_MD4`, `RSA_MD5`, `RSA_SHA1`, `_x509_time::sec`, `_x509_cert::serial`, `_x509_cert::sig_oid1`, `_x509_cert::subject`, `_x509_cert::valid_from`, `_x509_cert::valid_to`, `_x509_cert::version`, `x509parse_dn_gets()`, and `_x509_time::year`.

Referenced by `debug_print_cert()`, and `main()`.

13.114.4.4 int x509parse_crt (x509_cert * crt, unsigned char * buf, int buflen)

Parse one or more certificates and add them to the chained list.

Parameters:

- chain* points to the start of the chain
- buf* buffer holding the certificate data
- buflen* size of the buffer

Returns:

- 0 if successful, or a specific X509 error code

Definition at line 647 of file x509parse.c.

References ASN1_CONSTRUCTED, asn1_get_tag(), ASN1_SEQUENCE, base64_decode(), _x509_cert::ca_istrue, rsa_context::E, _x509_cert::issuer, _x509_cert::issuer_id, _x509_cert::issuer_raw, rsa_context::len, _x509_buf::len, _x509_cert::max_pathlen, mpi_size(), rsa_context::N, _x509_cert::next, OID_PKCS1, _x509_buf::p, _x509_cert::pk_oid, _x509_cert::raw, _x509_cert::rsa, rsa_check_pubkey(), _x509_cert::serial, _x509_cert::sig, _x509_cert::sig_oid1, _x509_cert::sig_oid2, _x509_cert::subject, _x509_cert::subject_id, _x509_cert::subject_raw, _x509_cert::tbs, _x509_cert::v3_ext, _x509_cert::valid_from, _x509_cert::valid_to, _x509_cert::version, x509_free(), x509_get_alg(), x509_get_dates(), x509_get_ext(), x509_get_name(), x509_get_pubkey(), x509_get_serial(), x509_get_sig(), x509_get_uid(), x509_get_version(), x509parse_crt(), XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_BASE64_INVALID_CHARACTER, XYSSL_ERR_X509_CERT_INVALID_FORMAT, XYSSL_ERR_X509_CERT_INVALID_PEM, XYSSL_ERR_X509_CERT_SIG_MISMATCH, XYSSL_ERR_X509_CERT_UNKNOWN_SIG_ALG, and XYSSL_ERR_X509_CERT_UNKNOWN_VERSION.

Referenced by main(), ssl_parse_certificate(), ssl_test(), x509_self_test(), x509parse_crt(), and x509parse_crtfile().

13.114.4.5 int x509parse_crtfile (x509_cert * crt, char * path)

Load one or more certificates and add them to the chained list.

Parameters:

- chain* points to the start of the chain
- path* filename to read the certificates from

Returns:

- 0 if successful, or a specific X509 error code

Definition at line 979 of file x509parse.c.

References buf, f, and x509parse_crt().

13.114.4.6 int x509parse_dn_gets (char * buf, char * end, x509_name * dn)

Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.

Definition at line 1316 of file x509parse.c.

References `_x509_buf::len`, `_x509_name::next`, `_x509_name::oid`, `OID_PKCS9`, `OID_X520`, `_x509_buf::p`, `PKCS9_EMAIL`, `_x509_name::val`, `X520_COMMON_NAME`, `X520_COUNTRY`, `X520_LOCALITY`, `X520_ORG_UNIT`, `X520_ORGANIZATION`, and `X520_STATE`.

Referenced by `x509parse_cert_info()`.

13.114.4.7 `int x509parse_expired(x509_cert * crt)`

Return 0 if the certificate is still valid, or `BADCERT_EXPIRED`.

Definition at line 1458 of file `x509parse.c`.

References `BADCERT_EXPIRED`, `_x509_time::day`, `_x509_time::mon`, `_x509_cert::valid_to`, and `_x509_time::year`.

Referenced by `x509parse_verify()`.

13.114.4.8 `int x509parse_key(rsa_context * rsa, unsigned char * buf, int buflen, unsigned char * pwd, int pwrlen)`

Parse a private RSA key.

Parameters:

rsa RSA context to be initialized
buf input buffer
buflen size of the buffer
pwd password for decryption (optional)
pwrlen size of the password

Returns:

0 if successful, or a specific X509 error code

Definition at line 1082 of file `x509parse.c`.

References `ASN1_CONSTRUCTED`, `asn1_get_int()`, `asn1_get_mpi()`, `asn1_get_tag()`, `ASN1_SEQUENCE`, `base64_decode()`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::len`, `mpi_size()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_free()`, `rsa_context::ver`, `x509_des3_decrypt()`, `x509_get_iv()`, `XYSSL_ERR_ASN1_LENGTH_MISMATCH`, `XYSSL_ERR_BASE64_INVALID_CHARACTER`, `XYSSL_ERR_X509_FEATURE_UNAVAILABLE`, `XYSSL_ERR_X509_KEY_INVALID_ENC_IV`, `XYSSL_ERR_X509_KEY_INVALID_FORMAT`, `XYSSL_ERR_X509_KEY_INVALID_PEM`, `XYSSL_ERR_X509_KEY_INVALID_VERSION`, `XYSSL_ERR_X509_KEY_PASSWORD_MISMATCH`, `XYSSL_ERR_X509_KEY_PASSWORD_REQUIRED`, and `XYSSL_ERR_X509_KEY_UNKNOWN_ENC_ALG`.

Referenced by `main()`, `ssl_test()`, `x509_self_test()`, and `x509parse_keyfile()`.

13.114.4.9 `int x509parse_keyfile(rsa_context * rsa, char * path, char * password)`

Load and parse a private RSA key.

Parameters:

rsa RSA context to be initialized

path filename to read the private key from
pwd password to decrypt the file (can be NULL)

Returns:

0 if successful, or a specific X509 error code

Definition at line 1269 of file x509parse.c.

References buf, f, and x509parse_key().

13.114.4.10 int x509parse_verify (x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)

Verify the certificate signature.

Parameters:

crt a certificate to be verified
trust_ca the trusted CA chain
cn expected Common Name (can be set to NULL if the CN must not be verified)
flags result of the verification

Returns:

0 if successful or XYSSL_ERR_X509_SIG_VERIFY_FAILED, in which case *flags will have one or more of the following values set: BADCERT_EXPIRED -- BADCERT_REVOKED -- BADCERT_CN_MISMATCH -- BADCERT_NOT_TRUSTED

Note:

TODO: add two arguments, depth and crl

Definition at line 1503 of file x509parse.c.

References BADCERT_CN_MISMATCH, BADCERT_NOT_TRUSTED, _x509_cert::ca_istrue, cur, _x509_cert::issuer_raw, _x509_buf::len, _x509_cert::max_pathlen, _x509_cert::next, _x509_name::next, _x509_name::oid, OID_CN, _x509_buf::p, _x509_cert::rsa, rsa_pkcs1_verify(), RSA_PUBLIC, _x509_cert::sig, _x509_cert::sig_oid1, _x509_cert::subject, _x509_cert::subject_raw, _x509_cert::tbs, _x509_name::val, _x509_cert::version, x509_hash(), x509parse_expired(), and XYSSL_ERR_X509_CERT_VERIFY_FAILED.

Referenced by ssl_parse_certificate(), and x509_self_test().

13.115 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/aes.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/aes.h"
#include "xyssl/padlock.h"
#include <string.h>
#include <stdio.h>
```

Defines

- #define [GET_ULONG_LE](#)(n, b, i)
- #define [PUT_ULONG_LE](#)(n, b, i)
- #define [ROTL8](#)(x) ((x << 8) & 0xFFFFFFFF) | (x >> 24)
- #define [XTIME](#)(x) ((x << 1) ^ ((x & 0x80) ? 0x1B : 0x00))
- #define [MUL](#)(x, y) ((x && y) ? pow[(log[x]+log[y]) % 255] : 0)
- #define [AES_FROUND](#)(X0, X1, X2, X3, Y0, Y1, Y2, Y3)
- #define [AES_RROUND](#)(X0, X1, X2, X3, Y0, Y1, Y2, Y3)

Functions

- static void [aes_gen_tables](#) (void)
- void [aes_setkey_enc](#) ([aes_context](#) *ctx, unsigned char *key, [int](#) keysize)
AES key schedule (encryption).
- void [aes_setkey_dec](#) ([aes_context](#) *ctx, unsigned char *key, [int](#) keysize)
AES key schedule (decryption).
- void [aes_crypt_ecb](#) ([aes_context](#) *ctx, [int](#) mode, unsigned char input[16], unsigned char output[16])
AES-ECB block encryption/decryption.
- void [aes_crypt_cbc](#) ([aes_context](#) *ctx, [int](#) mode, [int](#) length, unsigned char iv[16], unsigned char *input, unsigned char *output)
AES-CBC buffer encryption/decryption.
- void [aes_crypt_cfb](#) ([aes_context](#) *ctx, [int](#) mode, [int](#) length, [int](#) *iv_off, unsigned char iv[16], unsigned char *input, unsigned char *output)
AES-CFB buffer encryption/decryption.
- [int](#) [aes_self_test](#) ([int](#) verbose)
Checkup routine.

Variables

- static unsigned char [FSb](#) [256]
- static unsigned long [FT0](#) [256]
- static unsigned long [FT1](#) [256]
- static unsigned long [FT2](#) [256]
- static unsigned long [FT3](#) [256]
- static unsigned char [RSb](#) [256]
- static unsigned long [RT0](#) [256]
- static unsigned long [RT1](#) [256]
- static unsigned long [RT2](#) [256]
- static unsigned long [RT3](#) [256]
- static unsigned long [RCON](#) [10]
- static `int` [aes_init_done](#) = 0
- static const unsigned char [aes_test_ecb_dec](#) [3][16]
- static const unsigned char [aes_test_ecb_enc](#) [3][16]
- static const unsigned char [aes_test_cbc_dec](#) [3][16]
- static const unsigned char [aes_test_cbc_enc](#) [3][16]
- static const unsigned char [aes_test_cfb_dec](#) [3][16]
- static const unsigned char [aes_test_cfb_enc](#) [3][16]

13.115.1 Define Documentation

13.115.1.1 `#define` AES_FROUND(X0, X1, X2, X3, Y0, Y1, Y2, Y3)

Value:

```
{
    X0 = *RK++ ^ FT0[ ( Y0          ) & 0xFF ] ^ \
              FT1[ ( Y1 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y2 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y3 >> 24 ) & 0xFF ]; \
    X1 = *RK++ ^ FT0[ ( Y1          ) & 0xFF ] ^ \
              FT1[ ( Y2 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y3 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y0 >> 24 ) & 0xFF ]; \
    X2 = *RK++ ^ FT0[ ( Y2          ) & 0xFF ] ^ \
              FT1[ ( Y3 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y0 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y1 >> 24 ) & 0xFF ]; \
    X3 = *RK++ ^ FT0[ ( Y3          ) & 0xFF ] ^ \
              FT1[ ( Y0 >>  8 ) & 0xFF ] ^ \
              FT2[ ( Y1 >> 16 ) & 0xFF ] ^ \
              FT3[ ( Y2 >> 24 ) & 0xFF ]; \
}
```

Definition at line 590 of file aes.c.

Referenced by `aes_crypt_ecb()`.

13.115.1.2 #define AES_RROUND(X0, X1, X2, X3, Y0, Y1, Y2, Y3)**Value:**

```

{
    X0 = *RK++ ^ RT0[ ( Y0          ) & 0xFF ] ^ \
                RT1[ ( Y3 >>  8 ) & 0xFF ] ^ \
                RT2[ ( Y2 >> 16 ) & 0xFF ] ^ \
                RT3[ ( Y1 >> 24 ) & 0xFF ]; \
    X1 = *RK++ ^ RT0[ ( Y1          ) & 0xFF ] ^ \
                RT1[ ( Y0 >>  8 ) & 0xFF ] ^ \
                RT2[ ( Y3 >> 16 ) & 0xFF ] ^ \
                RT3[ ( Y2 >> 24 ) & 0xFF ]; \
    X2 = *RK++ ^ RT0[ ( Y2          ) & 0xFF ] ^ \
                RT1[ ( Y1 >>  8 ) & 0xFF ] ^ \
                RT2[ ( Y0 >> 16 ) & 0xFF ] ^ \
                RT3[ ( Y3 >> 24 ) & 0xFF ]; \
    X3 = *RK++ ^ RT0[ ( Y3          ) & 0xFF ] ^ \
                RT1[ ( Y2 >>  8 ) & 0xFF ] ^ \
                RT2[ ( Y1 >> 16 ) & 0xFF ] ^ \
                RT3[ ( Y0 >> 24 ) & 0xFF ]; \
}

```

Definition at line 613 of file aes.c.

Referenced by aes_crypt_ecb().

13.115.1.3 #define GET_ULONG_LE(n, b, i)**Value:**

```

{
    (n) = ( (unsigned long) (b)[(i)      ]          ) \
    | ( (unsigned long) (b)[(i) + 1] <<  8 ) \
    | ( (unsigned long) (b)[(i) + 2] << 16 ) \
    | ( (unsigned long) (b)[(i) + 3] << 24 ); \
}

```

Definition at line 40 of file aes.c.

Referenced by aes_crypt_ecb(), aes_setkey_enc(), and md5_process().

13.115.1.4 #define MUL(x, y) ((x && y) ? pow[(log[x]+log[y]) % 255] : 0)

Definition at line 354 of file aes.c.

Referenced by aes_gen_tables().

13.115.1.5 #define PUT_ULONG_LE(n, b, i)**Value:**

```

{
    (b)[(i)      ] = (unsigned char) ( (n)          ); \
    (b)[(i) + 1] = (unsigned char) ( (n) >>  8 ); \
    (b)[(i) + 2] = (unsigned char) ( (n) >> 16 ); \
    (b)[(i) + 3] = (unsigned char) ( (n) >> 24 ); \
}

```

Definition at line 50 of file aes.c.

Referenced by aes_crypt_ecb(), and md5_finish().

13.115.1.6 **#define ROTL8(x) ((x << 8) & 0xFFFFFFFF) | (x >> 24)**

Definition at line 352 of file aes.c.

Referenced by aes_gen_tables().

13.115.1.7 **#define XTIME(x) ((x << 1) ^ ((x & 0x80) ? 0x1B : 0x00))**

Definition at line 353 of file aes.c.

Referenced by aes_gen_tables().

13.115.2 Function Documentation

13.115.2.1 **void aes_crypt_cbc (aes_context * ctx, int mode, int length, unsigned char iv[16], unsigned char * input, unsigned char * output)**

AES-CBC buffer encryption/decryption.

Parameters:

ctx AES context
mode AES_ENCRYPT or AES_DECRYPT
length length of the input data
iv initialization vector (updated after use)
input buffer holding the input data
output buffer holding the output data

Definition at line 732 of file aes.c.

References aes_crypt_ecb(), and AES_DECRYPT.

Referenced by aes_self_test(), main(), ssl_decrypt_buf(), and ssl_encrypt_buf().

13.115.2.2 **void aes_crypt_cfb (aes_context * ctx, int mode, int length, int * iv_off, unsigned char iv[16], unsigned char * input, unsigned char * output)**

AES-CFB buffer encryption/decryption.

Parameters:

ctx AES context
mode AES_ENCRYPT or AES_DECRYPT
length length of the input data
iv_off offset in IV (updated after use)
iv initialization vector (updated after use)
input buffer holding the input data

output buffer holding the output data

Definition at line 787 of file aes.c.

References aes_crypt_ecb(), AES_DECRYPT, and AES_ENCRYPT.

Referenced by aes_self_test().

13.115.2.3 void aes_crypt_ecb (aes_context * ctx, int mode, unsigned char input[16], unsigned char output[16])

AES-ECB block encryption/decryption.

Parameters:

ctx AES context

mode AES_ENCRYPT or AES_DECRYPT

input 16-byte input block

output 16-byte output block

Definition at line 639 of file aes.c.

References AES_DECRYPT, AES_FROUND, AES_RROUND, FSb, GET_ULONG_LE, aes_context::nr, PUT_ULONG_LE, aes_context::rk, and RSb.

Referenced by aes_crypt_cbc(), aes_crypt_cfb(), aes_en_de(), aes_self_test(), and main().

13.115.2.4 static void aes_gen_tables (void) [static]

Definition at line 358 of file aes.c.

References FSb, FT0, FT1, FT2, FT3, MUL, RCON, ROTL8, RSb, RT0, RT1, RT2, RT3, and XTIME.

Referenced by aes_setkey_enc().

13.115.2.5 int aes_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 902 of file aes.c.

References aes_crypt_cbc(), aes_crypt_cfb(), aes_crypt_ecb(), AES_DECRYPT, aes_setkey_dec(), aes_setkey_enc(), aes_test_cbc_dec, aes_test_cbc_enc, aes_test_cfb_dec, aes_test_cfb_enc, aes_test_ecb_dec, aes_test_ecb_enc, buf, and prv.

Referenced by main().

13.115.2.6 void aes_setkey_dec (aes_context * ctx, unsigned char * key, int keysize)

AES key schedule (decryption).

Parameters:

ctx AES context to be initialized
key decryption key
keysize must be 128, 192 or 256

Definition at line 542 of file aes.c.

References aes_setkey_enc(), aes_context::buf, FSb, aes_context::nr, aes_context::rk, RT0, RT1, RT2, and RT3.

Referenced by aes_en_de(), aes_self_test(), main(), and ssl_derive_keys().

13.115.2.7 void aes_setkey_enc (aes_context * ctx, unsigned char * key, int keysize)

AES key schedule (encryption).

Parameters:

ctx AES context to be initialized
key encryption key
keysize must be 128, 192 or 256

Definition at line 439 of file aes.c.

References aes_gen_tables(), aes_init_done, aes_context::buf, FSb, GET_ULONG_LE, aes_context::nr, RCON, and aes_context::rk.

Referenced by aes_en_de(), aes_self_test(), aes_setkey_dec(), main(), and ssl_derive_keys().

13.115.3 Variable Documentation**13.115.3.1 int aes_init_done = 0 [static]**

Definition at line 356 of file aes.c.

Referenced by aes_setkey_enc().

13.115.3.2 const unsigned char aes_test_cbc_dec[3][16] [static]**Initial value:**

```
{
    { 0xFA, 0xCA, 0x37, 0xE0, 0xB0, 0xC8, 0x53, 0x73,
      0xDF, 0x70, 0x6E, 0x73, 0xF7, 0xC9, 0xAF, 0x86 },
    { 0x5D, 0xF6, 0x78, 0xDD, 0x17, 0xBA, 0x4E, 0x75,
      0xB6, 0x17, 0x68, 0xC6, 0xAD, 0xEF, 0x7C, 0x7B },
    { 0x48, 0x04, 0xE1, 0x81, 0x8F, 0xE6, 0x29, 0x75,
      0x19, 0xA3, 0xE8, 0x8C, 0x57, 0x31, 0x04, 0x13 }
}
```

Definition at line 856 of file aes.c.

Referenced by aes_self_test().

13.115.3.3 const unsigned char aes_test_cbc_enc[3][16] [static]**Initial value:**

```
{
    { 0x8A, 0x05, 0xFC, 0x5E, 0x09, 0x5A, 0xF4, 0x84,
      0x8A, 0x08, 0xD3, 0x28, 0xD3, 0x68, 0x8E, 0x3D },
    { 0x7B, 0xD9, 0x66, 0xD5, 0x3A, 0xD8, 0xC1, 0xBB,
      0x85, 0xD2, 0xAD, 0xFA, 0xE8, 0x7B, 0xB1, 0x04 },
    { 0xFE, 0x3C, 0x53, 0x65, 0x3E, 0x2F, 0x45, 0xB5,
      0x6F, 0xCD, 0x88, 0xB2, 0xCC, 0x89, 0x8F, 0xF0 }
}
```

Definition at line 866 of file aes.c.

Referenced by aes_self_test().

13.115.3.4 const unsigned char aes_test_cfb_dec[3][16] [static]**Initial value:**

```
{
    { 0xBA, 0x75, 0x0C, 0xC9, 0x77, 0xF8, 0xD4, 0xE1,
      0x3E, 0x0F, 0xB5, 0x46, 0x2E, 0xA6, 0x33, 0xF6 },
    { 0xDB, 0x40, 0x4A, 0x98, 0x7B, 0xAA, 0xA3, 0xF3,
      0x92, 0x35, 0xAD, 0x58, 0x09, 0x9B, 0xFF, 0x6E },
    { 0xA8, 0x17, 0x41, 0x0E, 0x76, 0x71, 0x60, 0xE5,
      0xFD, 0x37, 0xC5, 0x43, 0xCC, 0xC8, 0xD6, 0xDA }
}
```

Definition at line 879 of file aes.c.

Referenced by aes_self_test().

13.115.3.5 const unsigned char aes_test_cfb_enc[3][16] [static]**Initial value:**

```
{
    { 0x45, 0x62, 0xC5, 0xA1, 0xF9, 0x10, 0x8F, 0xE0,
      0x87, 0x24, 0x25, 0x68, 0xB5, 0x12, 0xF3, 0x8B },
    { 0xB8, 0xD4, 0xD5, 0x09, 0xF5, 0xEE, 0x08, 0x38,
      0x48, 0x9B, 0x9D, 0xAD, 0x11, 0xB4, 0x2E, 0xD2 },
    { 0xE9, 0x10, 0x80, 0xDA, 0xEE, 0x2D, 0x81, 0xD9,
      0x41, 0x78, 0x91, 0xD5, 0x98, 0x78, 0xE1, 0xFA }
}
```

Definition at line 889 of file aes.c.

Referenced by aes_self_test().

13.115.3.6 const unsigned char aes_test_ecb_dec[3][16] [static]**Initial value:**

```

{
    { 0x44, 0x41, 0x6A, 0xC2, 0xD1, 0xF5, 0x3C, 0x58,
      0x33, 0x03, 0x91, 0x7E, 0x6B, 0xE9, 0xEB, 0xE0 },
    { 0x48, 0xE3, 0x1E, 0x9E, 0x25, 0x67, 0x18, 0xF2,
      0x92, 0x29, 0x31, 0x9C, 0x19, 0xF1, 0x5B, 0xA4 },
    { 0x05, 0x8C, 0xCF, 0xFD, 0xBB, 0xCB, 0x38, 0x2D,
      0x1F, 0x6F, 0x56, 0x58, 0x5D, 0x8A, 0x4A, 0xDE }
}

```

Definition at line 836 of file aes.c.

Referenced by aes_self_test().

13.115.3.7 const unsigned char aes_test_ecb_enc[3][16] [static]

Initial value:

```

{
    { 0xC3, 0x4C, 0x05, 0x2C, 0xC0, 0xDA, 0x8D, 0x73,
      0x45, 0x1A, 0xFE, 0x5F, 0x03, 0xBE, 0x29, 0x7F },
    { 0xF3, 0xF6, 0x75, 0x2A, 0xE8, 0xD7, 0x83, 0x11,
      0x38, 0xF0, 0x41, 0x56, 0x06, 0x31, 0xB1, 0x14 },
    { 0x8B, 0x79, 0xEE, 0xCC, 0x93, 0xA0, 0xEE, 0x5D,
      0xFF, 0x30, 0xB4, 0xEA, 0x21, 0x63, 0x6D, 0xA4 }
}

```

Definition at line 846 of file aes.c.

Referenced by aes_self_test().

13.115.3.8 unsigned char FSb[256] [static]

Definition at line 329 of file aes.c.

Referenced by aes_crypt_ecb(), aes_gen_tables(), aes_setkey_dec(), and aes_setkey_enc().

13.115.3.9 unsigned long FT0[256] [static]

Definition at line 330 of file aes.c.

Referenced by aes_gen_tables().

13.115.3.10 unsigned long FT1[256] [static]

Definition at line 331 of file aes.c.

Referenced by aes_gen_tables().

13.115.3.11 unsigned long FT2[256] [static]

Definition at line 332 of file aes.c.

Referenced by aes_gen_tables().

13.115.3.12 unsigned long FT3[256] [static]

Definition at line 333 of file aes.c.

Referenced by aes_gen_tables().

13.115.3.13 unsigned long RCON[10] [static]

Definition at line 347 of file aes.c.

Referenced by aes_gen_tables(), and aes_setkey_enc().

13.115.3.14 unsigned char RSb[256] [static]

Definition at line 338 of file aes.c.

Referenced by aes_crypt_ecb(), and aes_gen_tables().

13.115.3.15 unsigned long RT0[256] [static]

Definition at line 339 of file aes.c.

Referenced by aes_gen_tables(), and aes_setkey_dec().

13.115.3.16 unsigned long RT1[256] [static]

Definition at line 340 of file aes.c.

Referenced by aes_gen_tables(), and aes_setkey_dec().

13.115.3.17 unsigned long RT2[256] [static]

Definition at line 341 of file aes.c.

Referenced by aes_gen_tables(), and aes_setkey_dec().

13.115.3.18 unsigned long RT3[256] [static]

Definition at line 342 of file aes.c.

Referenced by aes_gen_tables(), and aes_setkey_dec().

13.116 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/arc4.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/arc4.h"
#include <string.h>
#include <stdio.h>
```

Functions

- void [arc4_setup](#) ([arc4_context](#) *ctx, unsigned char *key, [int](#) keylen)
ARC4 key schedule.
- void [arc4_crypt](#) ([arc4_context](#) *ctx, unsigned char *buf, [int](#) buflen)
ARC4 cipher function.
- [int](#) [arc4_self_test](#) ([int](#) verbose)

Variables

- static const unsigned char [arc4_test_key](#) [3][8]
- static const unsigned char [arc4_test_pt](#) [3][8]
- static const unsigned char [arc4_test_ct](#) [3][8]

13.116.1 Function Documentation

13.116.1.1 void [arc4_crypt](#) ([arc4_context](#) *ctx, unsigned char *buf, [int](#) buflen)

ARC4 cipher function.

Parameters:

ctx ARC4 context
buf buffer to be processed
buflen amount of data in buf

Definition at line 63 of file arc4.c.

References [arc4_context::m](#), [arc4_context::x](#), and [arc4_context::y](#).

Referenced by [arc4_self_test\(\)](#), [main\(\)](#), [ssl_decrypt_buf\(\)](#), and [ssl_encrypt_buf\(\)](#).

13.116.1.2 [int](#) [arc4_self_test](#) ([int](#) verbose)

Definition at line 122 of file arc4.c.

References [arc4_crypt\(\)](#), [arc4_setup\(\)](#), [arc4_test_ct](#), [arc4_test_key](#), [arc4_test_pt](#), and [buf](#).

Referenced by [main\(\)](#).

13.116.1.3 void arc4_setup (arc4_context * *ctx*, unsigned char * *key*, int *keylen*)

ARC4 key schedule.

Parameters:

ctx ARC4 context to be initialized

key the secret key

keylen length of the key

Definition at line 35 of file arc4.c.

References arc4_context::m, arc4_context::x, and arc4_context::y.

Referenced by arc4_self_test(), main(), and ssl_derive_keys().

13.116.2 Variable Documentation

13.116.2.1 const unsigned char arc4_test_ct[3][8] [static]

Initial value:

```
{
    { 0x75, 0xB7, 0x87, 0x80, 0x99, 0xE0, 0xC5, 0x96 },
    { 0x74, 0x94, 0xC2, 0xE7, 0x10, 0x4B, 0x08, 0x79 },
    { 0xDE, 0x18, 0x89, 0x41, 0xA3, 0x37, 0x5D, 0x3A }
}
```

Definition at line 112 of file arc4.c.

Referenced by arc4_self_test().

13.116.2.2 const unsigned char arc4_test_key[3][8] [static]

Initial value:

```
{
    { 0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF },
    { 0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF },
    { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }
}
```

Definition at line 98 of file arc4.c.

Referenced by arc4_self_test().

13.116.2.3 const unsigned char arc4_test_pt[3][8] [static]

Initial value:

```
{
    { 0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF },
    { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 },
    { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }
}
```

Definition at line 105 of file arc4.c.

Referenced by arc4_self_test().

13.117 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/base64.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/base64.h"
#include <string.h>
#include <stdio.h>
```

Functions

- `int base64_encode` (unsigned char *dst, int *dlen, unsigned char *src, int slen)
Encode a buffer into base64 format.
- `int base64_decode` (unsigned char *dst, int *dlen, unsigned char *src, int slen)
Decode a base64-formatted buffer.
- `int base64_self_test` (int verbose)
Checkup routine.

Variables

- static const unsigned char `base64_enc_map` [64]
- static const unsigned char `base64_dec_map` [128]
- static const unsigned char `base64_test_dec` [64]
- static const unsigned char `base64_test_enc` []

13.117.1 Function Documentation

13.117.1.1 `int base64_decode` (unsigned char *dst, int *dlen, unsigned char *src, int slen)

Decode a base64-formatted buffer.

Parameters:

dst destination buffer
dlen size of the buffer
src source buffer
slen amount of data to be decoded

Returns:

0 if successful, XYSSL_ERR_BASE64_BUFFER_TOO_SMALL, or XYSSL_ERR_BASE64_INVALID_DATA if the input data is not correct. *dlen is always updated to reflect the amount of data that has (or would have) been written.

Note:

Call this function with *dlen = 0 to obtain the required buffer size in *dlen

Definition at line 121 of file base64.c.

References `base64_dec_map`, `XYSSL_ERR_BASE64_BUFFER_TOO_SMALL`, and `XYSSL_ERR_BASE64_INVALID_CHARACTER`.

Referenced by `base64_self_test()`, `x509parse_crt()`, and `x509parse_key()`.

13.117.1.2 int base64_encode (unsigned char * *dst*, int * *dlen*, unsigned char * *src*, int *slen*)

Encode a buffer into base64 format.

Parameters:

dst destination buffer

dlen size of the buffer

src source buffer

slen amount of data to be encoded

Returns:

0 if successful, or `XYSSL_ERR_BASE64_BUFFER_TOO_SMALL`. `*dlen` is always updated to reflect the amount of data that has (or would have) been written.

Note:

Call this function with `*dlen = 0` to obtain the required buffer size in `*dlen`

Definition at line 58 of file base64.c.

References base64_enc_map, and XYSSL_ERR_BASE64_BUFFER_TOO_SMALL.

Referenced by `base64_self_test()`.

13.117.1.3 int base64_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 206 of file base64.c.

References `base64_decode()`, `base64_encode()`, `base64_test_dec`, and `base64_test_enc`.

Referenced by main().

13.117.2 Variable Documentation

13.117.2.1 `const unsigned char base64_dec_map[128]` `[static]`

Initial value:

$$\{127, 127, 127, 127, 127, 127, 127, 127, 127, 127, 127,$$

```

127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
127, 127, 127, 127, 127, 127, 127, 127, 127, 127,
127, 127, 127, 62, 127, 127, 127, 63, 52, 53,
54, 55, 56, 57, 58, 59, 60, 61, 127, 127,
127, 64, 127, 127, 127, 0, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 11, 12, 13, 14,
15, 16, 17, 18, 19, 20, 21, 22, 23, 24,
25, 127, 127, 127, 127, 127, 127, 26, 27, 28,
29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
39, 40, 41, 42, 43, 44, 45, 46, 47, 48,
49, 50, 51, 127, 127, 127, 127, 127, 127
}

```

Definition at line 38 of file base64.c.

Referenced by base64_decode().

13.117.2.2 `const unsigned char base64_enc_map[64]` `[static]`

Initial value:

```

{
    'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',
    'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T',
    'U', 'V', 'W', 'X', 'Y', 'Z', 'a', 'b', 'c', 'd',
    'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n',
    'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x',
    'y', 'z', '0', '1', '2', '3', '4', '5', '6', '7',
    '8', '9', '+', '/'
}

```

Definition at line 27 of file base64.c.

Referenced by base64_encode().

13.117.2.3 `const unsigned char base64_test_dec[64]` `[static]`

Initial value:

```

{
    0x24, 0x48, 0x6E, 0x56, 0x87, 0x62, 0x5A, 0xBD,
    0xBF, 0x17, 0xD9, 0xA2, 0xC4, 0x17, 0x1A, 0x01,
    0x94, 0xED, 0x8F, 0x1E, 0x11, 0xB3, 0xD7, 0x09,
    0x0C, 0xB6, 0xE9, 0x10, 0x6F, 0x22, 0xEE, 0x13,
    0xCA, 0xB3, 0x07, 0x05, 0x76, 0xC9, 0xFA, 0x31,
    0x6C, 0x08, 0x34, 0xFF, 0x8D, 0xC2, 0x6C, 0x38,
    0x00, 0x43, 0xE9, 0x54, 0x97, 0xAF, 0x50, 0x4B,
    0xD1, 0x41, 0xBA, 0x95, 0x31, 0x5A, 0x0B, 0x97
}

```

Definition at line 187 of file base64.c.

Referenced by base64_self_test().

13.117.2.4 `const unsigned char base64_test_enc[]` `[static]`

Initial value:

```
"JEhuVodiWr2/F9mixBcaAZTtjx4Rs9cJDLbpEG8i7hPK"  
"swcFdsn6MWwINP+Nwmw4AEPpVJevUEvRQbqVMVoLlw=="
```

Definition at line 199 of file base64.c.

Referenced by base64_self_test().

13.118 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/bignum.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/bignum.h"
#include "xyssl/bn_mul.h"
#include <string.h>
#include <stdlib.h>
#include <stdarg.h>
```

Defines

- #define `ciL` ((int) sizeof(t_int))
- #define `biL` (ciL << 3)
- #define `biH` (ciL << 2)
- #define `BITS_TO_LIMBS`(i) (((i) + biL - 1) / biL)
- #define `CHARS_TO_LIMBS`(i) (((i) + ciL - 1) / ciL)

Functions

- void `mpi_init` (mpi *X,...)
Initialize one or more mpi.
- void `mpi_free` (mpi *X,...)
Unallocate one or more mpi.
- int `mpi_grow` (mpi *X, int nlimbs)
Enlarge to the specified number of limbs.
- int `mpi_copy` (mpi *X, mpi *Y)
Copy the contents of Y into X.
- void `mpi_swap` (mpi *X, mpi *Y)
Swap the contents of X and Y.
- int `mpi_lset` (mpi *X, int z)
Set value from integer.
- int `mpi_lsb` (mpi *X)
Return the number of least significant bits.
- int `mpi_msb` (mpi *X)
Return the number of most significant bits.
- int `mpi_size` (mpi *X)
Return the total size in bytes.

- static `int mpi_get_digit (t_int *d, int radix, char c)`
- `int mpi_read_string (mpi *X, int radix, char *s)`
Import from an ASCII string.
- static `int mpi_write_hlp (mpi *X, int radix, char **p)`
- `int mpi_write_string (mpi *X, int radix, char *s, int *slen)`
Export into an ASCII string.
- `int mpi_read_file (mpi *X, int radix, FILE *fin)`
Read X from an opened file.
- `int mpi_read_mystring (mpi *X, int radix, char *s)`
- `int mpi_write_file (char *p, mpi *X, int radix, FILE *fout)`
Write X into an opened file, or stdout.
- `int mpi_read_binary (mpi *X, unsigned char *buf, int buflen)`
Import X from unsigned binary data, big endian.
- `int mpi_write_binary (mpi *X, unsigned char *buf, int buflen)`
Export X into unsigned binary data, big endian.
- `int mpi_shift_l (mpi *X, int count)`
Left-shift: $X \ll = \text{count}$.
- `int mpi_shift_r (mpi *X, int count)`
Right-shift: $X \gg = \text{count}$.
- `int mpi_cmp_abs (mpi *X, mpi *Y)`
Compare unsigned values.
- `int mpi_cmp_mpi (mpi *X, mpi *Y)`
Compare signed values.
- `int mpi_cmp_int (mpi *X, int z)`
Compare signed values.
- `int mpi_add_abs (mpi *X, mpi *A, mpi *B)`
Unsigned addition: $X = |A| + |B|$.
- static void `mpi_sub_hlp (int n, t_int *s, t_int *d)`
- `int mpi_sub_abs (mpi *X, mpi *A, mpi *B)`
Unsigned subtraction: $X = |A| - |B|$.
- `int mpi_add_mpi (mpi *X, mpi *A, mpi *B)`
Signed addition: $X = A + B$.
- `int mpi_sub_mpi (mpi *X, mpi *A, mpi *B)`
Signed subtraction: $X = A - B$.
- `int mpi_add_int (mpi *X, mpi *A, int b)`

Signed addition: $X = A + b$.

- `int mpi_sub_int (mpi *X, mpi *A, int b)`

Signed subtraction: $X = A - b$.

- `static void mpi_mul_hlp (int i, t_int *s, t_int *d, t_int b)`
- `int mpi_mul_mpi (mpi *X, mpi *A, mpi *B)`

*Baseline multiplication: $X = A * B$.*

- `int mpi_mul_int (mpi *X, mpi *A, t_int b)`

*Baseline multiplication: $X = A * b$.*

- `int mpi_div_mpi (mpi *Q, mpi *R, mpi *A, mpi *B)`

*Division by mpi: $A = Q * B + R$.*

- `int mpi_div_int (mpi *Q, mpi *R, mpi *A, int b)`

*Division by int: $A = Q * b + R$.*

- `int mpi_mod_mpi (mpi *R, mpi *A, mpi *B)`

Modulo: $R = A \bmod B$.

- `int mpi_mod_int (t_int *r, mpi *A, int b)`

Modulo: $r = A \bmod b$.

- `static void mpi_montg_init (t_int *mm, mpi *N)`
- `static void mpi_montmul (mpi *A, mpi *B, mpi *N, t_int mm, mpi *T)`
- `static void mpi_montred (mpi *A, mpi *N, t_int mm, mpi *T)`
- `int mpi_exp_mod (mpi *X, mpi *A, mpi *E, mpi *N, mpi *_RR)`

Sliding-window exponentiation: $X = A^E \bmod N$.

- `int mpi_gcd (mpi *G, mpi *A, mpi *B)`

Greatest common divisor: $G = \gcd(A, B)$.

- `int mpi_inv_mod (mpi *X, mpi *A, mpi *N)`

Modular inverse: $X = A^{-1} \bmod N$.

- `int mpi_is_prime (mpi *X, int(*f_rng)(void *), void *p_rng)`

Miller-Rabin primality test.

- `int mpi_gen_prime (mpi *X, int nbits, int dh_flag, int(*f_rng)(void *), void *p_rng)`

Prime number generation.

- `int mpi_self_test (int verbose)`

Checkup routine.

Variables

- `static const int small_prime []`

13.118.1 Define Documentation

13.118.1.1 `#define biH (ciL << 2)`

Definition at line 41 of file bignum.c.

Referenced by `mpi_div_mpi()`, and `mpi_mod_int()`.

13.118.1.2 `#define biL (ciL << 3)`

Definition at line 40 of file bignum.c.

Referenced by `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_lsb()`, `mpi_montg_init()`, `mpi_msb()`, `mpi_shift_l()`, and `mpi_shift_r()`.

13.118.1.3 `#define BITS_TO_LIMBS(i) (((i) + biL - 1) / biL)`

Definition at line 46 of file bignum.c.

Referenced by `mpi_gen_prime()`, `mpi_read_string()`, and `mpi_shift_l()`.

13.118.1.4 `#define CHARS_TO_LIMBS(i) (((i) + ciL - 1) / ciL)`

Definition at line 47 of file bignum.c.

Referenced by `mpi_read_binary()`.

13.118.1.5 `#define ciL ((int) sizeof(t_int))`

Definition at line 39 of file bignum.c.

Referenced by `mpi_copy()`, `mpi_free()`, `mpi_gen_prime()`, `mpi_grow()`, `mpi_is_prime()`, `mpi_lset()`, `mpi_montmul()`, `mpi_read_binary()`, `mpi_read_string()`, `mpi_write_binary()`, and `mpi_write_string()`.

13.118.2 Function Documentation

13.118.2.1 `int mpi_add_abs (mpi * X, mpi * A, mpi * B)`

Unsigned addition: $X = |A| + |B|$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 694 of file bignum.c.

References `MPI_CHK`, `mpi_copy()`, `mpi_grow()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, and `mpi_sub_mpi()`.

13.118.2.2 `int mpi_add_int (mpi * X, mpi * A, int b)`

Signed addition: $X = A + b$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 860 of file bignum.c.

References `mpi_add_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_gen_prime()`, and `mpi_read_string()`.

13.118.2.3 int mpi_add_mpi (mpi * X, mpi * A, mpi * B)

Signed addition: $X = A + B$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 798 of file bignum.c.

References `mpi_add_abs()`, `MPI_CHK`, `mpi_cmp_abs()`, `mpi_sub_abs()`, and `mpi::s`.

Referenced by `mpi_add_int()`, `mpi_div_mpi()`, `mpi_inv_mod()`, `mpi_mod_mpi()`, and `rsa_private()`.

13.118.2.4 int mpi_cmp_abs (mpi * X, mpi * Y)

Compare unsigned values.

Returns:

1 if $|X|$ is greater than $|Y|$, -1 if $|X|$ is lesser than $|Y|$ or 0 if $|X|$ is equal to $|Y|$

Definition at line 615 of file bignum.c.

References `mpi::n`, and `mpi::p`.

Referenced by `mpi_add_mpi()`, `mpi_div_mpi()`, `mpi_montmul()`, `mpi_sub_abs()`, and `mpi_sub_mpi()`.

13.118.2.5 int mpi_cmp_int (mpi * X, int z)

Compare signed values.

Returns:

1 if X is greater than z, -1 if X is lesser than z or 0 if X is equal to z

Definition at line 678 of file bignum.c.

References `mpi_cmp_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mod_mpi()`, `mpi_write_hlp()`, `rsa_check_privkey()`, and `rsa_gen_key()`.

13.118.2.6 int mpi_cmp_mpi (mpi * X, mpi * Y)

Compare signed values.

Returns:

1 if X is greater than Y, -1 if X is lesser than Y or 0 if X is equal to Y

Definition at line 645 of file bignum.c.

References `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_cmp_int()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mod_mpi()`, `mpi_self_test()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

13.118.2.7 int mpi_copy (mpi * X, mpi * Y)

Copy the contents of Y into X.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 128 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_grow()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `mpi_add_abs()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_sub_abs()`, and `mpi_write_string()`.

13.118.2.8 int mpi_div_int (mpi * Q, mpi * R, mpi * A, int b)

Division by int: $A = Q * b + R$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if $b == 0$

Note:

Either Q or R can be NULL.

Definition at line 1173 of file bignum.c.

References `mpi_div_mpi()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `main()`, and `mpi_write_hlp()`.

13.118.2.9 int mpi_div_mpi (mpi * Q, mpi * R, mpi * A, mpi * B)

Division by `mpi`: $A = Q * B + R$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if $B == 0$

Note:

Either Q or R can be NULL.

Definition at line 1008 of file bignum.c.

References biH, biL, mpi_add_mpi(), MPI_CHK, mpi_cmp_abs(), mpi_cmp_int(), mpi_cmp_mpi(), mpi_copy(), mpi_free(), mpi_grow(), mpi_init(), mpi_lset(), mpi_msb(), mpi_mul_int(), mpi_shift_l(), mpi_shift_r(), mpi_sub_mpi(), mpi::n, mpi::p, mpi::s, and XYSSL_ERR_MPI_DIVISION_BY_ZERO.

Referenced by mpi_div_int(), mpi_mod_mpi(), and mpi_self_test().

13.118.2.10 int mpi_exp_mod (mpi * X, mpi * A, mpi * E, mpi * N, mpi * _RR)

Sliding-window exponentiation: $X = A^E \bmod N$.

Returns:

0 if successful, 1 if memory allocation failed, XYSSL_ERR_MPI_BAD_INPUT_DATA if N is negative or even

Note:

_RR is used to avoid re-computing $R \cdot R \bmod N$ across multiple calls, which speeds up things a bit. It can be set to NULL if the extra performance is unneeded.

Definition at line 1328 of file bignum.c.

References biL, MPI_CHK, mpi_cmp_int(), mpi_cmp_mpi(), mpi_copy(), mpi_free(), mpi_grow(), mpi_init(), mpi_lset(), mpi_mod_mpi(), mpi_montg_init(), mpi_montmul(), mpi_montred(), mpi_msb(), mpi_shift_l(), mpi::n, mpi::p, and XYSSL_ERR_MPI_BAD_INPUT_DATA.

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_make_public(), main(), mpi_is_prime(), mpi_self_test(), rsa_private(), and rsa_public().

13.118.2.11 void mpi_free (mpi * X, ...)

Unallocate one or more [mpi](#).

Definition at line 73 of file bignum.c.

References ciL, mpi::n, mpi::p, and mpi::s.

Referenced by dhm_free(), main(), mpi_div_mpi(), mpi_exp_mod(), mpi_gcd(), mpi_gen_prime(), mpi_inv_mod(), mpi_is_prime(), mpi_mul_mpi(), mpi_read_string(), mpi_self_test(), mpi_sub_abs(), mpi_write_string(), rsa_check_privkey(), rsa_free(), rsa_gen_key(), rsa_private(), and rsa_public().

13.118.2.12 int mpi_gcd (mpi * G, mpi * A, mpi * B)

Greatest common divisor: $G = \gcd(A, B)$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 1507 of file bignum.c.

References MPI_CHK, mpi_cmp_int(), mpi_cmp_mpi(), mpi_copy(), mpi_free(), mpi_init(), mpi_lset(), mpi_mul_mpi(), mpi_shift_r(), mpi_sub_abs(), mpi::p, and mpi::s.

Referenced by mpi_inv_mod(), rsa_check_privkey(), and rsa_gen_key().

13.118.2.13 `int mpi_gen_prime (mpi * X, int nbits, int dh_flag, int(*) (void *) f_rng, void * p_rng)`

Prime number generation.

Parameters:

X destination [mpi](#)
nbits required size of X in bits
dh_flag if 1, then (X-1)/2 will be prime too
f_rng RNG function
p_rng RNG parameter

Returns:

0 if successful (probably prime), 1 if memory allocation failed, XYSSL_ERR_MPI_BAD_INPUT_DATA if nbits is < 3

Definition at line 1778 of file bignum.c.

References BITS_TO_LIMBS, ciL, mpi_add_int(), MPI_CHK, mpi_free(), mpi_grow(), mpi_init(), mpi_is_prime(), mpi_lset(), mpi_msb(), mpi_shift_l(), mpi_shift_r(), mpi_sub_int(), mpi::p, XYSSL_ERR_MPI_BAD_INPUT_DATA, and XYSSL_ERR_MPI_NOT_ACCEPTABLE.

Referenced by main(), and rsa_gen_key().

13.118.2.14 `static int mpi_get_digit (t_int * d, int radix, char c) [static]`

Definition at line 226 of file bignum.c.

References XYSSL_ERR_MPI_INVALID_CHARACTER.

Referenced by mpi_read_file(), mpi_read_mystring(), and mpi_read_string().

13.118.2.15 `int mpi_grow (mpi * X, int nlimbs)`

Enlarge to the specified number of limbs.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 100 of file bignum.c.

References ciL, mpi::n, and mpi::p.

Referenced by dhm_make_params(), dhm_make_public(), mpi_add_abs(), mpi_copy(), mpi_div_mpi(), mpi_exp_mod(), mpi_gen_prime(), mpi_is_prime(), mpi_lset(), mpi_mul_mpi(), mpi_read_binary(), mpi_read_string(), and mpi_shift_l().

13.118.2.16 `void mpi_init (mpi * X, ...)`

Initialize one or more [mpi](#).

Definition at line 52 of file bignum.c.

References mpi::n, mpi::p, and mpi::s.

Referenced by `main()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_inv_mod()`, `mpi_is_prime()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_self_test()`, `mpi_sub_abs()`, `mpi_write_string()`, `rsa_check_privkey()`, `rsa_gen_key()`, `rsa_private()`, and `rsa_public()`.

13.118.2.17 `int mpi_inv_mod (mpi * X, mpi * A, mpi * N)`

Modular inverse: $X = A^{-1} \bmod N$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_BAD_INPUT_DATA` if `N` is negative or nil `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `A` has no inverse mod `N`

Definition at line 1549 of file `bignum.c`.

References `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_shift_r()`, `mpi_sub_mpi()`, `mpi::p`, `XYSSL_ERR_MPI_BAD_INPUT_DATA`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, `mpi_self_test()`, and `rsa_gen_key()`.

13.118.2.18 `int mpi_is_prime (mpi * X, int(*) (void *) f_rng, void * p_rng)`

Miller-Rabin primality test.

Returns:

0 if successful (probably prime), 1 if memory allocation failed, `XYSSL_ERR_MPI_NOT_ACCEPTABLE` if `X` is not prime

Definition at line 1667 of file `bignum.c`.

References `ciL`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_copy()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_grow()`, `mpi_init()`, `mpi_lsb()`, `mpi_mod_int()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_shift_r()`, `mpi_sub_int()`, `mpi::n`, `mpi::p`, `R`, `mpi::s`, `small_prime`, and `XYSSL_ERR_MPI_NOT_ACCEPTABLE`.

Referenced by `main()`, and `mpi_gen_prime()`.

13.118.2.19 `int mpi_lsb (mpi * X)`

Return the number of least significant bits.

Definition at line 185 of file `bignum.c`.

References `biL`, `int`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_is_prime()`.

13.118.2.20 `int mpi_lset (mpi * X, int z)`

Set value from integer.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 167 of file bignum.c.

References `ciL`, `MPI_CHK`, `mpi_grow()`, `mpi::n`, `mpi::p`, and `mpi::s`.

Referenced by `dhm_make_params()`, `dhm_make_public()`, `mpi_div_mpi()`, `mpi_exp_mod()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_inv_mod()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_read_string()`, and `rsa_gen_key()`.

13.118.2.21 `int mpi_mod_int (t_int * r, mpi * A, int b)`

Modulo: $r = A \bmod b$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if $b == 0$

Definition at line 1209 of file bignum.c.

References `biH`, `mpi::n`, `mpi::p`, and `XYSSL_ERR_MPI_DIVISION_BY_ZERO`.

Referenced by `mpi_is_prime()`, and `mpi_write_hlp()`.

13.118.2.22 `int mpi_mod_mpi (mpi * R, mpi * A, mpi * B)`

Modulo: $R = A \bmod B$.

Returns:

0 if successful, 1 if memory allocation failed, `XYSSL_ERR_MPI_DIVISION_BY_ZERO` if $B == 0$

Definition at line 1189 of file bignum.c.

References `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_div_mpi()`, and `mpi_sub_mpi()`.

Referenced by `mpi_exp_mod()`, `mpi_inv_mod()`, `mpi_is_prime()`, `rsa_check_privkey()`, `rsa_gen_key()`, and `rsa_private()`.

13.118.2.23 `static void mpi_montg_init (t_int * mm, mpi * N) [static]`

Definition at line 1259 of file bignum.c.

References `biL`, and `mpi::p`.

Referenced by `mpi_exp_mod()`.

13.118.2.24 `static void mpi_montmul (mpi * A, mpi * B, mpi * N, t_int mm, mpi * T) [static]`

Definition at line 1277 of file bignum.c.

References `ciL`, `mpi_cmp_abs()`, `mpi_mul_hlp()`, `mpi_sub_hlp()`, `mpi::n`, and `mpi::p`.

Referenced by `mpi_exp_mod()`, and `mpi_montred()`.

13.118.2.25 static void mpi_montred (mpi * A, mpi * N, t_int mm, mpi * T) [static]

Definition at line 1314 of file bignum.c.

References mpi_montmul(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_exp_mod().

13.118.2.26 int mpi_msb (mpi * X)

Return the number of most significant bits.

Definition at line 200 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by d2i_RSA_PUBKEY(), main(), mpi_div_mpi(), mpi_exp_mod(), mpi_gen_prime(), mpi_is_prime(), mpi_shift_l(), mpi_size(), mpi_write_string(), rsa_check_pubkey(), rsa_decryption(), rsa_encryption(), and rsa_gen_key().

13.118.2.27 static void mpi_mul_hlp (int i, t_int * s, t_int * d, t_int b) [static]

Definition at line 892 of file bignum.c.

References MULADDC_CORE, MULADDC_INIT, and MULADDC_STOP.

Referenced by mpi_montmul(), and mpi_mul_mpi().

13.118.2.28 int mpi_mul_int (mpi * X, mpi * A, t_int b)

Baseline multiplication: $X = A * b$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 992 of file bignum.c.

References mpi_mul_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by mpi_div_mpi(), and mpi_read_string().

13.118.2.29 int mpi_mul_mpi (mpi * X, mpi * A, mpi * B)

Baseline multiplication: $X = A * B$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 956 of file bignum.c.

References MPI_CHK, mpi_copy(), mpi_free(), mpi_grow(), mpi_init(), mpi_lset(), mpi_mul_hlp(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), mpi_gcd(), mpi_is_prime(), mpi_mul_int(), mpi_self_test(), rsa_check_privkey(), rsa_gen_key(), and rsa_private().

13.118.2.30 `int mpi_read_binary (mpi * X, unsigned char * buf, int buflen)`

Import X from unsigned binary data, big endian.

Parameters:

X destination [mpi](#)
buf input buffer
buflen input buffer size

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 484 of file bignum.c.

References `CHARS_TO_LIMBS`, `ciL`, `MPI_CHK`, `mpi_grow()`, `mpi_lset()`, and `mpi::p`.

Referenced by `asn1_get_mpi()`, `d2i_RSA_PUBKEY()`, `dhm_read_bignum()`, `dhm_read_public()`, `rsa_private()`, and `rsa_public()`.

13.118.2.31 `int mpi_read_file (mpi * X, int radix, FILE * fin)`

Read X from an opened file.

Parameters:

X destination [mpi](#)
radix input numeric base
fin input file handle

Returns:

0 if successful, or an `XYSSL_ERR_MPI_XXX` error code

Definition at line 391 of file bignum.c.

References `mpi_get_digit()`, `mpi_read_string()`, and `XYSSL_ERR_MPI_FILE_IO_ERROR`.

Referenced by `main()`.

13.118.2.32 `int mpi_read_mystring (mpi * X, int radix, char * s)`

Definition at line 418 of file bignum.c.

References `mpi_get_digit()`, and `mpi_read_string()`.

Referenced by `rsa_decryption()`, and `rsa_encryption()`.

13.118.2.33 `int mpi_read_string (mpi * X, int radix, char * s)`

Import from an ASCII string.

Parameters:

X destination [mpi](#)

radix input numeric base
s null-terminated string buffer

Returns:

0 if successful, or an XYSSL_ERR_MPI_XXX error code

Definition at line 243 of file bignum.c.

References BITS_TO_LIMBS, ciL, int, mpi_add_int(), MPI_CHK, mpi_free(), mpi_get_digit(), mpi_grow(), mpi_init(), mpi_lset(), mpi_mul_int(), mpi::p, mpi::s, and XYSSL_ERR_MPI_BAD_INPUT_DATA.

Referenced by main(), mpi_read_file(), mpi_read_mystring(), mpi_self_test(), rsa_self_test(), and ssl_set_dh_param().

13.118.2.34 int mpi_self_test (int *verbose*)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1854 of file bignum.c.

References MPI_CHK, mpi_cmp_mpi(), mpi_div_mpi(), mpi_exp_mod(), mpi_free(), mpi_init(), mpi_inv_mod(), mpi_mul_mpi(), and mpi_read_string().

Referenced by main().

13.118.2.35 int mpi_shift_l (mpi * *X*, int *count*)

Left-shift: $X \ll= \text{count}$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 526 of file bignum.c.

References biL, BITS_TO_LIMBS, MPI_CHK, mpi_grow(), mpi_msb(), mpi::n, and mpi::p.

Referenced by mpi_div_mpi(), mpi_exp_mod(), and mpi_gen_prime().

13.118.2.36 int mpi_shift_r (mpi * *X*, int *count*)

Right-shift: $X \gg= \text{count}$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 575 of file bignum.c.

References biL, mpi::n, and mpi::p.

Referenced by dhm_make_params(), dhm_make_public(), mpi_div_mpi(), mpi_gcd(), mpi_gen_prime(), mpi_inv_mod(), and mpi_is_prime().

13.118.2.37 int mpi_size (mpi * *X*)

Return the total size in bytes.

Definition at line 218 of file bignum.c.

References mpi_msb().

Referenced by dhm_calc_secret(), dhm_make_params(), dhm_read_params(), mpi_write_binary(), x509parse_crt(), and x509parse_key().

13.118.2.38 int mpi_sub_abs (mpi * *X*, mpi * *A*, mpi * *B*)

Unsigned subtraction: $X = |A| - |B|$.

Returns:

0 if successful, XYSSL_ERR_MPI_NEGATIVE_VALUE if *B* is greater than *A*

Definition at line 761 of file bignum.c.

References MPI_CHK, mpi_cmp_abs(), mpi_copy(), mpi_free(), mpi_init(), mpi_sub_hlp(), mpi::n, mpi::p, and XYSSL_ERR_MPI_NEGATIVE_VALUE.

Referenced by mpi_add_mpi(), mpi_gcd(), and mpi_sub_mpi().

13.118.2.39 static void mpi_sub_hlp (int *n*, t_int * *s*, t_int * *d*) [static]

Definition at line 740 of file bignum.c.

Referenced by mpi_montmul(), and mpi_sub_abs().

13.118.2.40 int mpi_sub_int (mpi * *X*, mpi * *A*, int *b*)

Signed subtraction: $X = A - b$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 876 of file bignum.c.

References mpi_sub_mpi(), mpi::n, mpi::p, and mpi::s.

Referenced by main(), mpi_gen_prime(), mpi_is_prime(), rsa_check_privkey(), and rsa_gen_key().

13.118.2.41 int mpi_sub_mpi (mpi * *X*, mpi * *A*, mpi * *B*)

Signed subtraction: $X = A - B$.

Returns:

0 if successful, 1 if memory allocation failed

Definition at line 829 of file bignum.c.

References mpi_add_abs(), MPI_CHK, mpi_cmp_abs(), mpi_sub_abs(), and mpi::s.

Referenced by mpi_div_mpi(), mpi_inv_mod(), mpi_mod_mpi(), mpi_sub_int(), and rsa_private().

13.118.2.42 void mpi_swap (mpi * *X*, mpi * *Y*)

Swap the contents of *X* and *Y*.

Definition at line 155 of file bignum.c.

Referenced by rsa_gen_key().

13.118.2.43 int mpi_write_binary (mpi * *X*, unsigned char * *buf*, int *buflen*)

Export *X* into unsigned binary data, big endian.

Parameters:

X source [mpi](#)

buf output buffer

buflen output buffer size

Returns:

0 if successful, XYSSL_ERR_MPI_BUFFER_TOO_SMALL if *buf* isn't large enough

Note:

Call this function with **buflen* = 0 to obtain the minimum required buffer size in **buflen*.

Definition at line 506 of file bignum.c.

References ciL, mpi_size(), mpi::p, and XYSSL_ERR_MPI_BUFFER_TOO_SMALL.

Referenced by dhm_calc_secret(), dhm_make_public(), rsa_private(), and rsa_public().

13.118.2.44 int mpi_write_file (char * *p*, mpi * *X*, int *radix*, FILE * *fout*)

Write *X* into an opened file, or stdout.

Parameters:

p prefix, can be NULL

X source [mpi](#)

radix output numeric base

fout output file handle

Returns:

0 if successful, or an XYSSL_ERR_MPI_XXX error code

Note:

Set *fout* == NULL to print *X* on the console.

Definition at line 447 of file bignum.c.

References MPI_CHK, mpi_write_string(), and XYSSL_ERR_MPI_FILE_IO_ERROR.

Referenced by generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), and main().

13.118.2.45 static int mpi_write_hlp (mpi * *X*, int *radix*, char *p*) [static]**

Definition at line 301 of file bignum.c.

References MPI_CHK, mpi_cmp_int(), mpi_div_int(), mpi_mod_int(), and XYSSL_ERR_MPI_BAD_INPUT_DATA.

Referenced by mpi_write_string().

13.118.2.46 int mpi_write_string (mpi * *X*, int *radix*, char * *s*, int * *slen*)

Export into an ASCII string.

Parameters:

X source [mpi](#)
radix output numeric base
s string buffer
slen string buffer size

Returns:

0 if successful, or an XYSSL_ERR_MPI_XXX error code

Note:

Call this function with *slen = 0 to obtain the minimum required buffer size in *slen.

Definition at line 328 of file bignum.c.

References ciL, MPI_CHK, mpi_copy(), mpi_free(), mpi_init(), mpi_msb(), mpi_write_hlp(), mpi::n, mpi::p, mpi::s, XYSSL_ERR_MPI_BAD_INPUT_DATA, and XYSSL_ERR_MPI_BUFFER_TOO_SMALL.

Referenced by mpi_write_file().

13.118.3 Variable Documentation**13.118.3.1 const int small_prime[] [static]****Initial value:**

```
{
    3,    5,    7,   11,   13,   17,   19,   23,
   29,   31,   37,   41,   43,   47,   53,   59,
   61,   67,   71,   73,   79,   83,   89,   97,
  101,  103,  107,  109,  113,  127,  131,  137,
  139,  149,  151,  157,  163,  167,  173,  179,
  181,  191,  193,  197,  199,  211,  223,  227,
  229,  233,  239,  241,  251,  257,  263,  269,
  271,  277,  281,  283,  293,  307,  311,  313,
  317,  331,  337,  347,  349,  353,  359,  367,
  373,  379,  383,  389,  397,  401,  409,  419,
  421,  431,  433,  439,  443,  449,  457,  461,
  463,  467,  479,  487,  491,  499,  503,  509,
  521,  523,  541,  547,  557,  563,  569,  571,
  577,  587,  593,  599,  601,  607,  613,  617,
```

```
        619,  631,  641,  643,  647,  653,  659,  661,  
        673,  677,  683,  691,  701,  709,  719,  727,  
        733,  739,  743,  751,  757,  761,  769,  773,  
        787,  797,  809,  811,  821,  823,  827,  829,  
        839,  853,  857,  859,  863,  877,  881,  883,  
        887,  907,  911,  919,  929,  937,  941,  947,  
        953,  967,  971,  977,  983,  991,  997, -103  
    }
```

Definition at line 1639 of file bignum.c.

Referenced by mpi_is_prime().

13.119 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/certs.c File Reference

```
#include "xyssl/config.h"
```

Variables

- char `test_ca_cert` []
- char `test_ca_key` []
- char `test_ca_pwd` [] = "test"
- char `test_srv_cert` []
- char `test_srv_key` []
- char `test_cli_cert` []
- char `test_cli_key` []
- char `xyssl_ca_cert` []

13.119.1 Variable Documentation

13.119.1.1 char test_ca_cert[]

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDpTCCAo2gAwIBAgIBADANBgkqhkiG9w0BAQUFADBFMQswCQYDVQQGEwJGUjEO\r\n"
"MAwGA1UEBxMFUGFyaXMxDjAMBgNVBAoTBVh5U1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDAxOFoXDTE3MDcwNzA1MDAxOFowRTElMAkGA1UE\r\n"
"BhMCRlIxZjAMBgNVBAcTBVBhcm1zMQ4wDAYDVQQKEwVYeVNTTDEWMBQGA1UEAxMN\r\n"
"WH1TU0wgVGZvdCBBDQTCASiWdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAM+k\r\n"
"gt70fIPiYqmvXr+9uPmWoN405eoSzxdiRLLCqL4V/Ts0E/H+JNfHS4DHLAgxrJu\r\n"
"+ZiadvSJuHkI6eliMkAh5SU1DqaF3jrrFdJCooM6a077M4CRkE1tdAeZdf+BYp0q\r\n"
"BeMU9Y2+j7ibsQaPAizunbXLf4QdteExCwhYRJ8OVXSEasNt339gJzTD6kOhES3b\r\n"
"lEN3qb61qFaJ5MLHTix5uNVc2rvbOizV5oLhJNqm52AKOp1ltv6WTiI8loagvAc\r\n"
"jlhEZRNB9el5SL6Jai/uFcqXKzfXNKW3FYpQHfGobmiMfGt1lUSBj3F2mrqEC7gC\r\n"
"whY/FDvAI64/k5LZAFkCAwEAAaOBnzCBnDAMBGNVHRMEBTADAQH/MB0GA1UdDgQW\r\n"
"BBS87h+Y6Porg+SkfV7DdXKTMdkyZzBtBgNVHSMEZjBkgBS87h+Y6Porg+SkfV7D\r\n"
"dXKTMdkyZ6FJpEcwRTElMAkGA1UEBhMCRlIxZjAMBgNVBAcTBVBhcm1zMQ4wDAYD\r\n"
"VQKKewVYeVNTTDEWMBQGA1UEAxMNWH1TU0wgVGZvdCBBDQYIBADANBgkqhkiG9w0B\r\n"
"AQUFAAOCAQEAIHdohONCg6KAAhWDSmfEgSbKUI8/Zr/d56uw42H00sj/uKPQzUco\r\n"
"3Mx2BYElm1itg7q5OhrkB7J4ZB78EtNZM84nV+y6od3YpR0Z9VUxCx7948MozYRy\r\n"
"TKF5x/lKHXl1PJKfEO4clKdWTFAtWtGhewXrHJQ8C+ENh2Up2wTVh3Z+pEzuZNv3\r\n"
"u/JYu1H+vkt3l1WCy/9mxUnu+anW1DzxPWnjy41x6Mi0BD2qfKBWLjVS+7v6ALcj\r\n"
"S2oRWwr4LUvXT7z9BBAvw2eJQD+a4uAya6EURG7AsAvr5MnWn/r0wLWmBJ6fBlYp\r\n"
"FlkOmamOFvstLMf74rLX+LGKeJ/nwui5FQ==\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 25 of file certs.c.

Referenced by `main()`, `ssl_test()`, and `x509_self_test()`.

13.119.1.2 char test_ca_key[]

Definition at line 49 of file certs.c.

Referenced by `x509_self_test()`.

13.119.1.3 char test_ca_pwd[] = "test"

Definition at line 81 of file certs.c.

Referenced by x509_self_test().

13.119.1.4 char test_cli_crt[]

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDPTCAiWgAwIBAgIBATANBgkqhkiG9w0BAQUFADBQMswCQYDVQQGEwJGUjE0\r\n"
"MAwGA1UEBxMFUGFyaXMxdjAMBgNVBAoTBVh5U1NMMRYwFAYDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDEyMFoXDTA4MDcwNjA1MDEyMFowMDELMAkGA1UE\r\n"
"BhMCR1IxdjAMBgNVBAoTBVh5U1NMMREwDwYDVQQDEwhKb2UgVXNlcjCCASIdQYJ\r\n"
"KoZIHvcNAQEBAQADggEPADCCAQoCggEBAKZkg8ANl6kGmLGqKc6KUHb9IsZwb2+K\r\n"
"jBw83Qb0KuvPVnu3MzEcXfV0Z83g0PL/z8ob5PKr8HP6bVYzhsD65imcCDEEVPk2\r\n"
"9x0XGTggGjB601Fd8aTShUWE4NLrKw6YNXTXgTndvhHNxXwqmdNVLkmZjj3ZwYUc\r\n"
"eEE8eE5jHs8cMDXJLMCwgKIM7Sax220hSHQHKwifVO4/Fdw5G+Suys8PhMX2jDXM\r\n"
"ICFwq8ld+bZGoNUtgp48FWhAMfJyTEaHh9LC46KkqGSDRIzx7/4cPB6QqrpzJN0o\r\n"
"Kr8KH7vdRDTFDm023D4C5l0Bw/2aC76DhEJpB2bGA4iIszJs+F/PIL8CAwEAAaNN\r\n"
"MEswCQYDVROTBAlwADAdBgNVHQ4EFgQUiWX1IvjrDYgt0zz5Sq16x01k0o4wHwYD\r\n"
"VR0jBBBgwFoAUvO4fm0j6K4PkpH1ew3VykzHZMmcwDQYJKoZIhvcNAQEFBQADggEB\r\n"
"AGdqD7VthJmC+oeemuHk2TQX2wZNU+GsC+RLjtlencnky95KnljGvMtCznyLkS5D\r\n"
"faJLKfR1No8pk5GrdsccgyIuQx5WnHNv4QBZmMsmvDICxzRQaxuPFHbS4aLXldL\r\n"
"yOWm524qkMHPCKvA86blYsEkksGDV47fF9ZkOQ8nkh7Z4eY4/5TwqTY72ww5g4NL\r\n"
"6DZtWpcpGbX99NRANvzcq9D+ElxkgHnH4YWafOKBclSgqrutbRLi2uZx/QpvuF+i\r\n"
"sUbe+HFPmWuU5lBv/oOhQkz0VD+HusYtXWS2lG88ct40aNLy2CkYUugdTR/b9Uea\r\n"
"p/i862sL/1040qlQ0xv5N7U=\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 134 of file certs.c.

Referenced by main(), and x509_self_test().

13.119.1.5 char test_cli_key[]

Initial value:

```
"-----BEGIN RSA PRIVATE KEY-----\r\n"
"MIIEowIBAAKCAQEApmsDwa2XqQaYsaopzopQdv0ixnBvb4qMHDzdBvQq689We7cz\r\n"
"MRxcW85nzeDQ8v/Pyhvk8qvw/ptVjOGwPrmKZwIMQRU+Tb2vRcZOCaAMHrTUV3x\r\n"
"pNKFRTg0usrDpgldNeBMl2+Ec3FfCqZ0lUuSZmOPdnBhRwxQTx4TmMezxwNcks\r\n"
"wLCAogztJrHbY6FIdAcrCJ9U7j8V3Dkb5K7Kzw+ExfaMNCwgIXCryV35tkag1S2C\r\n"
"njwVaEAX8nJMRoeH0sLjoqSoZINEjPHv/hw8HpCqunMk3SgqvYQfu91ENMUOY7bc\r\n"
"PgLMXQHD/ZoLvoOEQmKHZsYDiIizMmz4X88gvwIDAQABAoIBAEOBnBKAjDVN+j4ax\r\n"
"lDjEwZKqxVkmAUXDBDyDrCjxRoWY2gz7YW1ALUMUbeV0f05v1zVrwbkUKKZeVBx\r\n"
"QA9zRw28H8A6tfvolHgrIcx4dixMh3ePC+DVDJ6zglvKV2ipAwBufKYIrX0r4Io2\r\n"
"ZqUrNg9CeEYNlkHWceaN12rhYwO82pgHxnB1p5pI42pY7lzyLgSddf5n+M5UBOJI\r\n"
"gsNCKvbGdv7WQPVFTRDiRgEnCJ3rI8oPSK6MOUWJw3rh2hbKx+ex8NPvEKbzEXiU\r\n"
"p5jlAlbHIWP5sYBbAlYviFtryAV4fyfLcWPfoqa33Oozofjlwoj0Aixz+6rerLjZ\r\n"
"cpTSrAECgYEA2oCffUo6HH3Lq9oeWhFCOyG3YjZmFrJaJwJHnvroX9/pXHqYKog\r\n"
"TeHcjUJBtFzW0klcetYbZCFqT8v9nf0uPlgaiVGctXf1MSbFXDUFKkYBiFwzdWMT\r\n"
"Ysmvfhf82jMWZ8ecsXTyDRl858R5WPZ52qEsCc5X2un7QENm6FtVT8CgYEAwwKS\r\n"
"zQNzuoJETqZX7AalMk3JM8Fdam+Qm5LNMcbvbkKI8HKMS1VMuqaR0XdAX/imXAx\r\n"
"PlVhSsmoSDbsMpxBEZlptpCen/GocqITxANTakrBHxqb2aQ5EEu7SgzfHZWse3\r\n"
"vQEYfctFB1PdcdZUDzk4/w7WmyivpYtCWoaHlIECgYEA0UYZ+1UJfVpapRj+swMP\r\n"
"DrQbo7i7t7lUaFYLNpFX2OPLTWC5txqn1OruTu5VHDqE+5hneDNUUTT3uOg4B2q\r\n"
"mdmmaNjh2M6wz0e0BVfexhNQynqMaqTe32IOM8DFs3L0xacgg7JfVn6P7CeQGOVe\r\n"
"wc96kICw6ZxhtJSqpOGipt8CgYBI/0Pw+IXxJK4nNSpe+u4vCYP5mUI9hKEFYCbt\r\n"
"qKwvyAUknn/zgiIQ+z/iSErFMPmlwXjvWi0gL/qPb+Pp4hCLX8u2zNhY08Ppx4Gin\r\n"
```

```
"Ej+pAntWxq+kHyfKEI5dyRwV/snfvlqwjy404JsSF3VMhIMdYDPzbb72Qnni5w5l\r\n"
"j0OeAQKBgBqt9jJMd1JdpemC2dm0BuuDIz2h3/MH+CMjfaDLenVpKykn17B6N92h\r\n"
"k1MesqK3RQzDGwauDw431LQw0R69onn9fCM3wJw2yEC6wC9sF8I8hsNZbt64yZhZ\r\n"
"4Bi2YRTiHhpEuBqKlHHLDFHneo3SMYh8PU/PDQQcyWGHHUi9z1RE\r\n"
"-----END RSA PRIVATE KEY-----\r\n"
```

Definition at line 156 of file certs.c.

Referenced by main().

13.119.1.6 char test_srv_cert[]

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIIDPjCCAiaGAWIBAgIBAJANBgkqhkiG9w0BAQUFADBQMswCQQYDVQQGEWJGUjE0\r\n"
"MAwGA1UEBxMFUGFyaXMxZjAMBGNVBAoTBVh5U1NMMRYwFAyDVQQDEw1YeVNTTCBU\r\n"
"ZXN0IENBMB4XDTA3MDcwNzA1MDEyOVoxDTA4MDcwNjA1MDEyOVowMTELMakGA1UE\r\n"
"BhMCRlIxZjAMBGNVBAoTBVh5U1NMMRIwEAYDVQQDEwlsb2NhbGhvc3QwgGEiMA0G\r\n"
"CSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC40PDcGTgmHkt6noXDfkjVuymjiNYB\r\n"
"gttiL7uAlKe3tXStacEecQek/OJxYqYr7ffcWals29LL6HbKpi0xLZKBbD9ACkDh\r\n"
"1Z/SvHlyQPILJdYb9DMw+kzZds5myXUjzn7Aem1YjoxMZUAMyc34i2900X2pL0v2\r\n"
"SfCEJ9Ym4MOnZxYl217+dX9ZbkGIgrT6uY2IYK4boDwxBTcyT8i/NPsVsIMwtWPM\r\n"
"rnQMr+XbgS98sUzcZE70Pe1TlV9Iy8j/8d2OiFo+qTyMu/6UpM2s3gdkQkMzx+Sm\r\n"
"4QitRUjzmEXeUePRUjEgHiv7vz069xuVBzrks36w5BXiVAhLke/OTKVPAGMBAAGj\r\n"
"TTBLMAkGA1UdEwQCAAAwHQYDVROBBYEFNkOyCTx64SDdPySGW1/tzD7/WMSMB8G\r\n"
"AlUdIwQYMBaAFzLzuH5jo+iuD5KR9XsN1cpMx2TJnMA0GCSqGSIb3DQEBBQUAA4IB\r\n"
"AQBelJv5t+suayq5Lo5bJneHjNZfgg8EigDQ7Nqaosv1QZAsh2N34Gg5YdkGyVdg\r\n"
"s32I/K5aaywyUbG9qVXQxCM2T95qBqyK56h9yJoZKWQD9H//+zB8kCK/16WvRfv3\r\n"
"V7eSR19qOFwLHe+1qGh2YhxduFYi+fm4D36dGxqC2A34tZjo0QPHKtIeqM0kJy\r\n"
"zzL65T1bJQKkyTurHofFv0jW9ZFG2wkGysVgCY5fjuLi1do/sWUaXd2987iNfa+K\r\n"
"FrHsTi6urSfZuG1ZNxDXDHEE7Q2snAvvev+KR7DD9X4DJGcPX9gA4CGJj+9ZzyAA\r\n"
"ZTGpOzk1hIH44RFs2lJMZR1E\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 83 of file certs.c.

Referenced by main(), and ssl_test().

13.119.1.7 char test_srv_key[]

Initial value:

```
"-----BEGIN RSA PRIVATE KEY-----\r\n"
"MIIEowIBAAKCAQEAuNDw3Bk4Jh5Lep6Fw35I1bsp04jWAYI7Yi+7gNSnt7V0rWnB\r\n"
"HnEHpPzicWKmK+333FmpUtvSy+h2yqYtMS2SgWw/QApA4dWf0rx5ckDyCyXWG/Qz\r\n"
"MPpM2XbOZsl1I85+wHptWI6MTGVADMnN+ItvdNF9qS9L9knwniFWJuDDp2cWJdte\r\n"
"/nV/WW5ICIK0+rmNiGCuG6A8MW03Mk/IvzT7FbIjMLVjzK50DK/124EvfLFM3GRO\r\n"
"9D3tU5VfSMvI/HdjohaPqk8jLv+1KTNrN4HZEJDM8fkpuEIrUVI85hF31Hj0Vix\r\n"
"IBYL+7890vcblQc65LN+sOQV4lQIS5HvzkylTwIDAQABAoIBABeah8h0aBlmMRmd\r\n"
"+VN4Y3D4kf7UcRCMQ21Mz1Oq1Si/QgGLyiBLK0DFE16LzNE7eTZpNRjh/lAQhmt\r\n"
"QcpQGa/x1TomLRbCo8DUVWZkKQWHdYroa01MDliPtDimzhEepE2M1T5EJmLzY3S\r\n"
"qVGe7UMsJjJfWgJAezyXteANQK+2YSt+CjPIqIHch1KexUnvdN9++1oEx6AbuZ8T\r\n"
"4avhFYZQP15tZNGsk2LfQLYS/NfbowkCsd0/TVubJBmDGUML/E5MbxjxLz1aNB2M\r\n"
"V59cBNngsgA35CODAUf4xOyoSfZGqG1Rb9qQrv1E6Jz56dG8SsKF3HqnDjxiPOVBN\r\n"
"FBnVJ+ECgYEA29MhAsKMM4XqBUKp6pIMFTgm/s1E5vxiG70vqiL+guvBhhQ7zs1\r\n"
"8UMTNXxZOMELNoB/ev9fN0Cjc1Vr46b/x/yDw7wMb96i+vzEN0zu4RHWi3OWpCPbp\r\n"
"qBKEi3hzN8M+BulPX8CDQx3aLRrfxw51J5EuA0NeybngbItgxTi0u6kCgYEA1zr0\r\n"
"6P5YdOhYHTSWD1ked49MApcVuzaHnsHZVAhUqu3Rwiy9LRaJLZfr7fQDb9DYJbZp\r\n"
"sxTRLG6LSAcSR7mw+m+GvNqGt/9pSqbtW+L/VwVWSyF+YYklxZUD3UAAyrDVcDEC\r\n"
"a5S+jad4Csi/1Vht5ulWickWL1fJvadnSubKNDCcGYA+71xVGPP+1sFgTiytfrC8\r\n"
```

```
"5n2rl4MxinJ9+w0I+EbZCKNMYGvTgiU4dJasSMEdiBKslFMGo7dF8F0BLHF1IsIa\r\n"
"5Aht2XItXn9l54o90iTQXMmK6qmRaneM6fhOoeaCwYAhpGxYIpx/Xr4TOhiag46\r\n"
"jMMaphAeOvw4t1K2RDziOQKBgQCYPCCU0gxuw/o1jda2CxbZy9EmU/erEX09+0n+\r\n"
"TOFQpSEPq/z9WaxAFY9LfsdZ0ZktoeHmalbNdL3i6A3DWAM3YSQzQMRPmzOWnqXx\r\n"
"cgoCBmlvzkzaeLjO5phMoLQHJmmafvuCG6uxov3F8Hi3LyHUF2c8k0nL6ucmJ3vj\r\n"
"uzu4AQKBgBSASMAJS63M9UJB1Eazy2v2Nww04CmzNxUfWrHuKpd/C2ik4QKu0sRO\r\n"
"r9KnkDgxxEhjDm7lXhlW12PU42yORst5I3EaalCfi4KPFn/ozt+iNBYrzd8Tyvnb\r\n"
"qkdEC10+G2Fo/ER4NRCv7a24WNEsOMGzGRqw5cnSJrjbZLYMaIyK\r\n"
"-----END RSA PRIVATE KEY-----\r\n"
```

Definition at line 105 of file certs.c.

Referenced by main(), and ssl_test().

13.119.1.8 char xyssl_ca_cert[]

Initial value:

```
"-----BEGIN CERTIFICATE-----\r\n"
"MIID4DCCAsigAwIBAgIJAOLw9BMV1jxMMA0GCSqGSIb3DQEBBQUAMFMCzAJBgNV\r\n"
"BAYTAKZSMQ4wDAYDVQQIEwVQYXJpczEOMAwGA1UEChMFWH1TU0wxJDAiBgNVBAMT\r\n"
"G1h5U1NMIENlcnRpZmljYXRlIEF1dGhvcml0eTAeFw0wNjEwMzEyMjU5MjRaFw0x\r\n"
"NjEwMzEyMjU5MjRaMFMCzAJBgNVBAYTAKZSMQ4wDAYDVQQIEwVQYXJpczEOMAwG\r\n"
"A1UEChMFWH1TU0wxJDAiBgNVBAMTG1h5U1NMIENlcnRpZmljYXRlIEF1dGhvcml0\r\n"
"eTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKnprf1RQ7IYPI3FmI/h\r\n"
"f2EJGfaIP+Jt551VZWFrS3A56Nn4KS57zTAKZUA7YGBLwLAznfaphJ7SvENALeZR\r\n"
"/J0c/n9jwMpfXReQL7RVpgg/zlR+t2DUI3DAwigPZiHHCSJSBC73vpMc6uH0eV2d\r\n"
"itqjJUnJG5F8Zg9/gX4UMRADlwGWqxvs+jc9i0XFKEZga8+rOn6WvKyBM4e20I\r\n"
"HcO4BPF92d6sm4qLgyR4oXUkBz6NfDWX8ZdTvXuRaK9qMy1327cCT48sis9F6/eK\r\n"
"QAx0VHlqGTtxDYjHJPsmLNejuyUDvsaC8TMCSpFTXpMTUvJdmsF1LZWmIo5lY5\r\n"
"zh0CAwEAAaOBtjCBsZAdBgNVHQ4EFgQUBlVzoNjrkkgPkj8xjuFvVP2E4GswgYMG\r\n"
"AlUdIwR8MHqAFaZVc6DSa5JID5CFMY7hb1T9hOBroVekVTBTMQswCQYDVQQGEwJG\r\n"
"UjEOMAwGA1UECBMFUGFyaXMxDjAMBGNVBAoTBVh5U1NMMSQwIgYDVQQDEXTYeVNT\r\n"
"TCBDZXJ0aWZpY2F0ZSBBDXR0b3JpdHmCCQDi8PQTFdY8TDAMBGNVHRMEBTADAQH\r\n"
"MA0GCSqGSIb3DQEBBQUAA4IBAQCgD65b2l5BASFSpVlrcRnLzu/99eWTVAJwJbbD\r\n"
"VhPAQiet0W4U/85EDK7uoFo/SEjyMB/m4T20A8FIDaK7jBPo/1gtbuQjGMR17h+z\r\n"
"F2iGuNh26Td26Uzqclt3oiFtSvDRoZ/9kqkEy7Lrs7FBzOmvfTvrqvAdf7cLma2D\r\n"
"ri/otDpPr4XoDnwd4C+4bQC/Gr3Uder4VAeTOJtKdGqfYlvPwPSPVBDuVLUybKi\r\n"
"8cMAT6p9IG1e12u6vFqcBT/I67Q0bGU6gzVVz9ZVULXOYzMjjLafVXC1gesUH2WT\r\n"
"gTEAnEBkSRrkfAi+RezoEFAbmEl3fPt09dwSPku3x7cB3zaJ\r\n"
"-----END CERTIFICATE-----\r\n"
```

Definition at line 185 of file certs.c.

Referenced by main().

13.120 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/debug.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/debug.h"
#include <stdarg.h>
#include <stdlib.h>
```

Functions

- char * [debug_fmt](#) (const char *format,...)
- void [debug_print_msg](#) (ssl_context *ssl, int level, char *file, int line, char *text)
- void [debug_print_ret](#) (ssl_context *ssl, int level, char *file, int line, char *text, int ret)
- void [debug_print_buf](#) (ssl_context *ssl, int level, char *file, int line, char *text, unsigned char *buf, int len)
- void [debug_print_mpi](#) (ssl_context *ssl, int level, char *file, int line, char *text, mpi *X)
- void [debug_print_cert](#) (ssl_context *ssl, int level, char *file, int line, char *text, x509_cert *cert)

13.120.1 Function Documentation

13.120.1.1 char* debug_fmt (const char *format, ...)

Definition at line 38 of file debug.c.

13.120.1.2 void debug_print_buf (ssl_context *ssl, int level, char *file, int line, char *text, unsigned char *buf, int len)

Definition at line 82 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

13.120.1.3 void debug_print_cert (ssl_context *ssl, int level, char *file, int line, char *text, x509_cert *cert)

Definition at line 169 of file debug.c.

References `debug_print_mpi()`, `rsa_context::E`, `_ssl_context::f_dbg`, `rsa_context::N`, `_x509_cert::next`, `_ssl_context::p_dbg`, `_x509_cert::rsa`, and `x509parse_cert_info()`.

13.120.1.4 void debug_print_mpi (ssl_context *ssl, int level, char *file, int line, char *text, mpi *X)

Definition at line 124 of file debug.c.

References `_ssl_context::f_dbg`, `mpi::n`, `mpi::p`, and `_ssl_context::p_dbg`.

Referenced by `debug_print_cert()`.

13.120.1.5 void debug_print_msg (ssl_context * *ssl*, int *level*, char * *file*, int *line*, char * *text*)

Definition at line 52 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

13.120.1.6 void debug_print_ret (ssl_context * *ssl*, int *level*, char * *file*, int *line*, char * *text*, int *ret*)

Definition at line 66 of file debug.c.

References `_ssl_context::f_dbg`, and `_ssl_context::p_dbg`.

13.121 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/des.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/des.h"
#include <string.h>
#include <stdio.h>
```

Defines

- #define [GET_ULONG_BE](#)(n, b, i)
- #define [PUT_ULONG_BE](#)(n, b, i)
- #define [DES_IP](#)(X, Y)
- #define [DES_FP](#)(X, Y)
- #define [DES_ROUND](#)(X, Y)
- #define [SWAP](#)(a, b) { unsigned long t = a; a = b; b = t; t = 0; }

Functions

- static void [des_setkey](#) (unsigned long SK[32], unsigned char key[8])
- void [des_setkey_enc](#) ([des_context](#) *ctx, unsigned char key[8])
DES key schedule (56-bit, encryption).
- void [des_setkey_dec](#) ([des_context](#) *ctx, unsigned char key[8])
DES key schedule (56-bit, decryption).
- static void [des3_set2key](#) (unsigned long esk[96], unsigned long dsk[96], unsigned char key[16])
- void [des3_set2key_enc](#) ([des3_context](#) *ctx, unsigned char key[16])
Triple-DES key schedule (112-bit, encryption).
- void [des3_set2key_dec](#) ([des3_context](#) *ctx, unsigned char key[16])
Triple-DES key schedule (112-bit, decryption).
- static void [des3_set3key](#) (unsigned long esk[96], unsigned long dsk[96], unsigned char key[24])
- void [des3_set3key_enc](#) ([des3_context](#) *ctx, unsigned char key[24])
Triple-DES key schedule (168-bit, encryption).
- void [des3_set3key_dec](#) ([des3_context](#) *ctx, unsigned char key[24])
Triple-DES key schedule (168-bit, decryption).
- void [des_crypt_ecb](#) ([des_context](#) *ctx, unsigned char input[8], unsigned char output[8])
DES-ECB block encryption/decryption.
- void [des_crypt_cbc](#) ([des_context](#) *ctx, int mode, int length, unsigned char iv[8], unsigned char *input, unsigned char *output)
DES-CBC buffer encryption/decryption.

- void [des3_crypt_ecb](#) ([des3_context](#) *ctx, unsigned char input[8], unsigned char output[8])
3DES-ECB block encryption/decryption
- void [des3_crypt_cbc](#) ([des3_context](#) *ctx, int mode, int length, unsigned char iv[8], unsigned char *input, unsigned char *output)
3DES-CBC buffer encryption/decryption
- int [des_self_test](#) (int verbose)

Variables

- static const unsigned long [SB1](#) [64]
- static const unsigned long [SB2](#) [64]
- static const unsigned long [SB3](#) [64]
- static const unsigned long [SB4](#) [64]
- static const unsigned long [SB5](#) [64]
- static const unsigned long [SB6](#) [64]
- static const unsigned long [SB7](#) [64]
- static const unsigned long [SB8](#) [64]
- static const unsigned long [LHs](#) [16]
- static const unsigned long [RHs](#) [16]
- static const unsigned char [des3_test_keys](#) [24]
- static const unsigned char [des3_test_iv](#) [8]
- static const unsigned char [des3_test_buf](#) [8]
- static const unsigned char [des3_test_ecb_dec](#) [3][8]
- static const unsigned char [des3_test_ecb_enc](#) [3][8]
- static const unsigned char [des3_test_cbc_dec](#) [3][8]
- static const unsigned char [des3_test_cbc_enc](#) [3][8]

13.121.1 Define Documentation

13.121.1.1 #define DES_FP(X, Y)

Value:

```
{
    X = ((X << 31) | (X >> 1)) & 0xFFFFFFFF; \
    T = (X ^ Y) & 0xAAAAAAAA; X ^= T; Y ^= T; \
    Y = ((Y << 31) | (Y >> 1)) & 0xFFFFFFFF; \
    T = ((Y >> 8) ^ X) & 0x00FF00FF; X ^= T; Y ^= (T << 8); \
    T = ((Y >> 2) ^ X) & 0x33333333; X ^= T; Y ^= (T << 2); \
    T = ((X >> 16) ^ Y) & 0x0000FFFF; Y ^= T; X ^= (T << 16); \
    T = ((X >> 4) ^ Y) & 0x0F0F0F0F; Y ^= T; X ^= (T << 4); \
}
```

Definition at line 257 of file des.c.

Referenced by [des3_crypt_ecb\(\)](#), and [des_crypt_ecb\(\)](#).

13.121.1.2 #define DES_IP(X, Y)**Value:**

```

{
    T = ((X >> 4) ^ Y) & 0x0F0F0F0F; Y ^= T; X ^= (T << 4); \
    T = ((X >> 16) ^ Y) & 0x0000FFFF; Y ^= T; X ^= (T << 16); \
    T = ((Y >> 2) ^ X) & 0x33333333; X ^= T; Y ^= (T << 2); \
    T = ((Y >> 8) ^ X) & 0x00FF00FF; X ^= T; Y ^= (T << 8); \
    Y = ((Y << 1) | (Y >> 31)) & 0xFFFFFFFF; \
    T = (X ^ Y) & 0xAAAAAAAA; Y ^= T; X ^= T; \
    X = ((X << 1) | (X >> 31)) & 0xFFFFFFFF; \
}

```

Definition at line 243 of file des.c.

Referenced by des3_crypt_ecb(), and des_crypt_ecb().

13.121.1.3 #define DES_ROUND(X, Y)**Value:**

```

{
    T = *SK++ ^ X; \
    Y ^= SB8[ (T >> 8) & 0x3F ] ^ \
        SB6[ (T >> 16) & 0x3F ] ^ \
        SB4[ (T >> 24) & 0x3F ] ^ \
        SB2[ (T >> 32) & 0x3F ]; \
    T = *SK++ ^ ((X << 28) | (X >> 4)); \
    Y ^= SB7[ (T >> 8) & 0x3F ] ^ \
        SB5[ (T >> 16) & 0x3F ] ^ \
        SB3[ (T >> 24) & 0x3F ] ^ \
        SB1[ (T >> 32) & 0x3F ]; \
}

```

Definition at line 271 of file des.c.

Referenced by des3_crypt_ecb(), and des_crypt_ecb().

13.121.1.4 #define GET_ULONG_BE(n, b, i)**Value:**

```

{
    (n) = ( (unsigned long) (b)[(i) << 24] << 24 ) \
        | ( (unsigned long) (b)[(i) << 16] << 16 ) \
        | ( (unsigned long) (b)[(i) << 8] << 8 ) \
        | ( (unsigned long) (b)[(i)] << 0 ); \
}

```

Definition at line 39 of file des.c.

Referenced by des3_crypt_ecb(), des_crypt_ecb(), des_setkey(), sha1_process(), and sha2_process().

13.121.1.5 #define PUT_ULONG_BE(n, b, i)**Value:**

```

{
    (b) [(i)      ] = (unsigned char) ( (n) >> 24 ); \
    (b) [(i) + 1] = (unsigned char) ( (n) >> 16 ); \
    (b) [(i) + 2] = (unsigned char) ( (n) >>  8 ); \
    (b) [(i) + 3] = (unsigned char) ( (n)          ); \
}

```

Definition at line 49 of file des.c.

Referenced by des3_crypt_ecb(), des_crypt_ecb(), sha1_finish(), and sha2_finish().

13.121.1.6 #define SWAP(a, b) { unsigned long t = a; a = b; b = t; t = 0; }

Definition at line 286 of file des.c.

Referenced by des_setkey_dec().

13.121.2 Function Documentation

13.121.2.1 void des3_crypt_cbc (des3_context * ctx, int mode, int length, unsigned char iv[8], unsigned char * input, unsigned char * output)

3DES-CBC buffer encryption/decryption

Parameters:

ctx 3DES context
mode DES_ENCRYPT or DES_DECRYPT
length length of the input data
iv initialization vector (updated after use)
input buffer holding the input data
output buffer holding the output data

Definition at line 593 of file des.c.

References des3_crypt_ecb(), and DES_ENCRYPT.

Referenced by des_self_test(), main(), ssl_decrypt_buf(), ssl_encrypt_buf(), and x509_des3_decrypt().

13.121.2.2 void des3_crypt_ecb (des3_context * ctx, unsigned char input[8], unsigned char output[8])

3DES-ECB block encryption/decryption

Parameters:

ctx 3DES context
input 64-bit input block
output 64-bit output block

Definition at line 552 of file des.c.

References DES_FP, DES_IP, DES_ROUND, GET_ULONG_BE, PUT_ULONG_BE, and des3_context::sk.

Referenced by des3_crypt_cbc(), and des_self_test().

13.121.2.3 static void des3_set2key (unsigned long *esk*[96], unsigned long *dsk*[96], unsigned char *key*[16]) [static]

Definition at line 381 of file des.c.

References des_setkey().

Referenced by des3_set2key_dec(), and des3_set2key_enc().

13.121.2.4 void des3_set2key_dec (des3_context * *ctx*, unsigned char *key*[16])

Triple-DES key schedule (112-bit, decryption).

Parameters:

ctx 3DES context to be initialized

key 16-byte secret key

Definition at line 420 of file des.c.

References des3_set2key(), and des3_context::sk.

Referenced by des_self_test().

13.121.2.5 void des3_set2key_enc (des3_context * *ctx*, unsigned char *key*[16])

Triple-DES key schedule (112-bit, encryption).

Parameters:

ctx 3DES context to be initialized

key 16-byte secret key

Definition at line 409 of file des.c.

References des3_set2key(), and des3_context::sk.

Referenced by des_self_test().

13.121.2.6 static void des3_set3key (unsigned long *esk*[96], unsigned long *dsk*[96], unsigned char *key*[24]) [static]

Definition at line 428 of file des.c.

References des_setkey().

Referenced by des3_set3key_dec(), and des3_set3key_enc().

13.121.2.7 void des3_set3key_dec (des3_context * *ctx*, unsigned char *key*[24])

Triple-DES key schedule (168-bit, decryption).

Parameters:

ctx 3DES context to be initialized

key 24-byte secret key

Definition at line 465 of file des.c.

References `des3_set3key()`, and `des3_context::sk`.

Referenced by `des_self_test()`, `ssl_derive_keys()`, and `x509_des3_decrypt()`.

13.121.2.8 `void des3_set3key_enc (des3_context * ctx, unsigned char key[24])`

Triple-DES key schedule (168-bit, encryption).

Parameters:

ctx 3DES context to be initialized

key 24-byte secret key

Definition at line 454 of file des.c.

References `des3_set3key()`, and `des3_context::sk`.

Referenced by `des_self_test()`, `main()`, and `ssl_derive_keys()`.

13.121.2.9 `void des_crypt_cbc (des_context * ctx, int mode, int length, unsigned char iv[8], unsigned char * input, unsigned char * output)`

DES-CBC buffer encryption/decryption.

Parameters:

ctx DES context

mode DES_ENCRYPT or DES_DECRYPT

length length of the input data

iv initialization vector (updated after use)

input buffer holding the input data

output buffer holding the output data

Definition at line 505 of file des.c.

References `des_crypt_ecb()`, and `DES_ENCRYPT`.

Referenced by `des_self_test()`, and `main()`.

13.121.2.10 `void des_crypt_ecb (des_context * ctx, unsigned char input[8], unsigned char output[8])`

DES-ECB block encryption/decryption.

Parameters:

ctx DES context

input 64-bit input block

output 64-bit output block

Definition at line 476 of file des.c.

References `DES_FP`, `DES_IP`, `DES_ROUND`, `GET_ULONG_BE`, `PUT_ULONG_BE`, and `des_context::sk`.

Referenced by `des_crypt_cbc()`, and `des_self_test()`.

13.121.2.11 int des_self_test (int *verbose*)

Definition at line 694 of file des.c.

References `buf`, `des3_crypt_cbc()`, `des3_crypt_ecb()`, `des3_set2key_dec()`, `des3_set2key_enc()`, `des3_set3key_dec()`, `des3_set3key_enc()`, `des3_test_buf`, `des3_test_cbc_dec`, `des3_test_cbc_enc`, `des3_test_ecb_dec`, `des3_test_ecb_enc`, `des3_test_iv`, `des3_test_keys`, `des_crypt_cbc()`, `des_crypt_ecb()`, `DES_DECRYPT`, `des_setkey_dec()`, `des_setkey_enc()`, and `prv`.

Referenced by `main()`.

13.121.2.12 static void des_setkey (unsigned long *SK*[32], unsigned char *key*[8]) [static]

Definition at line 288 of file des.c.

References `GET_ULONG_BE`, `LHs`, and `RHs`.

Referenced by `des3_set2key()`, `des3_set3key()`, `des_setkey_dec()`, and `des_setkey_enc()`.

13.121.2.13 void des_setkey_dec (des_context * *ctx*, unsigned char *key*[8])

DES key schedule (56-bit, decryption).

Parameters:

ctx DES context to be initialized

key 8-byte secret key

Definition at line 368 of file des.c.

References `des_setkey()`, `des_context::sk`, and `SWAP`.

Referenced by `des_self_test()`.

13.121.2.14 void des_setkey_enc (des_context * *ctx*, unsigned char *key*[8])

DES key schedule (56-bit, encryption).

Parameters:

ctx DES context to be initialized

key 8-byte secret key

Definition at line 360 of file des.c.

References `des_setkey()`, and `des_context::sk`.

Referenced by `des_self_test()`, and `main()`.

13.121.3 Variable Documentation**13.121.3.1 const unsigned char des3_test_buf[8] [static]****Initial value:**

```
{
    0x4E, 0x6F, 0x77, 0x20, 0x69, 0x73, 0x20, 0x74
}
```

Definition at line 658 of file des.c.

Referenced by des_self_test().

13.121.3.2 `const unsigned char des3_test_cbc_dec[3][8] [static]`

Initial value:

```
{
    { 0x12, 0x9F, 0x40, 0xB9, 0xD2, 0x00, 0x56, 0xB3 },
    { 0x47, 0x0E, 0xFC, 0x9A, 0x6B, 0x8E, 0xE3, 0x93 },
    { 0xC5, 0xCE, 0xCF, 0x63, 0xEC, 0xEC, 0x51, 0x4C }
}
```

Definition at line 677 of file des.c.

Referenced by des_self_test().

13.121.3.3 `const unsigned char des3_test_cbc_enc[3][8] [static]`

Initial value:

```
{
    { 0x54, 0xF1, 0x5A, 0xF6, 0xEB, 0xE3, 0xA4, 0xB4 },
    { 0x35, 0x76, 0x11, 0x56, 0x5F, 0xA1, 0x8E, 0x4D },
    { 0xCB, 0x19, 0x1F, 0x85, 0xD1, 0xED, 0x84, 0x39 }
}
```

Definition at line 684 of file des.c.

Referenced by des_self_test().

13.121.3.4 `const unsigned char des3_test_ecb_dec[3][8] [static]`

Initial value:

```
{
    { 0xCD, 0xD6, 0x4F, 0x2F, 0x94, 0x27, 0xC1, 0x5D },
    { 0x69, 0x96, 0xC8, 0xFA, 0x47, 0xA2, 0xAB, 0xEB },
    { 0x83, 0x25, 0x39, 0x76, 0x44, 0x09, 0x1A, 0x0A }
}
```

Definition at line 663 of file des.c.

Referenced by des_self_test().

13.121.3.5 `const unsigned char des3_test_ecb_enc[3][8] [static]`

Initial value:

```
{
    { 0x6A, 0x2A, 0x19, 0xF4, 0x1E, 0xCA, 0x85, 0x4B },
    { 0x03, 0xE6, 0x9F, 0x5B, 0xFA, 0x58, 0xEB, 0x42 },
    { 0xDD, 0x17, 0xE8, 0xB8, 0xB4, 0x37, 0xD2, 0x32 }
}
```

Definition at line 670 of file des.c.

Referenced by des_self_test().

13.121.3.6 const unsigned char des3_test_iv[8] [static]

Initial value:

```
{
    0x12, 0x34, 0x56, 0x78, 0x90, 0xAB, 0xCD, 0xEF,
}
```

Definition at line 653 of file des.c.

Referenced by des_self_test().

13.121.3.7 const unsigned char des3_test_keys[24] [static]

Initial value:

```
{
    0x01, 0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF,
    0x23, 0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF, 0x01,
    0x45, 0x67, 0x89, 0xAB, 0xCD, 0xEF, 0x01, 0x23
}
```

Definition at line 646 of file des.c.

Referenced by des_self_test().

13.121.3.8 const unsigned long LHs[16] [static]

Initial value:

```
{
    0x00000000, 0x00000001, 0x00000100, 0x00000101,
    0x00010000, 0x00010001, 0x00010100, 0x00010101,
    0x01000000, 0x01000001, 0x01000100, 0x01000101,
    0x01010000, 0x01010001, 0x01010100, 0x01010101
}
```

Definition at line 224 of file des.c.

Referenced by des_setkey().

13.121.3.9 const unsigned long RHs[16] [static]

Initial value:

```
{
    0x00000000, 0x01000000, 0x00010000, 0x01010000,
    0x00000100, 0x01000100, 0x00010100, 0x01010100,
    0x00000001, 0x01000001, 0x00010001, 0x01010001,
    0x00000101, 0x01000101, 0x00010101, 0x01010101,
}
```

Definition at line 232 of file des.c.

Referenced by des_setkey().

13.121.3.10 const unsigned long SB1[64] [static]

Initial value:

```
{
    0x01010400, 0x00000000, 0x00010000, 0x01010404,
    0x01010004, 0x00010404, 0x00000004, 0x00010000,
    0x00000400, 0x01010400, 0x01010404, 0x00000400,
    0x01000404, 0x01010004, 0x01000000, 0x00000004,
    0x00000404, 0x01000400, 0x01000400, 0x00010400,
    0x00010400, 0x01010000, 0x01010000, 0x01000404,
    0x00010004, 0x01000004, 0x01000004, 0x00010004,
    0x00000000, 0x00000404, 0x00010404, 0x01000000,
    0x00010000, 0x01010404, 0x00000004, 0x01010000,
    0x01010400, 0x01000000, 0x01000000, 0x00000400,
    0x01010004, 0x00010000, 0x00010400, 0x01000004,
    0x00000400, 0x00000004, 0x01000404, 0x00010404,
    0x01010404, 0x00010004, 0x01010000, 0x01000404,
    0x01000004, 0x00000404, 0x00010404, 0x01010400,
    0x00000404, 0x01000400, 0x01000400, 0x00000000,
    0x00010004, 0x00010400, 0x00000000, 0x01010004
}
```

Definition at line 61 of file des.c.

13.121.3.11 const unsigned long SB2[64] [static]

Initial value:

```
{
    0x80108020, 0x80008000, 0x00008000, 0x00108020,
    0x00100000, 0x00000020, 0x80100020, 0x80008020,
    0x80000020, 0x80108020, 0x80108000, 0x80000000,
    0x80008000, 0x00100000, 0x00000020, 0x80100020,
    0x00108000, 0x00100020, 0x80008020, 0x00000000,
    0x80000000, 0x00008000, 0x00108020, 0x80100000,
    0x00100020, 0x80000020, 0x00000000, 0x00108000,
    0x00008020, 0x80108000, 0x80100000, 0x00008020,
    0x00000000, 0x00108020, 0x80100020, 0x00100000,
    0x80008020, 0x80100000, 0x80108000, 0x00008000,
    0x80100000, 0x80008000, 0x00000020, 0x80108020,
    0x00108020, 0x00000020, 0x00008000, 0x80000000,
    0x00008020, 0x80108000, 0x00100000, 0x80000020,
    0x00100020, 0x80008020, 0x80000020, 0x00100020,
    0x00108000, 0x00000000, 0x80008000, 0x00008020,
    0x80000000, 0x80100020, 0x80108020, 0x00108000
}
```

Definition at line 81 of file des.c.

13.121.3.12 const unsigned long SB3[64] [static]**Initial value:**

```
{
    0x00000208, 0x08020200, 0x00000000, 0x08020008,
    0x08000200, 0x00000000, 0x00020208, 0x08000200,
    0x00020008, 0x08000008, 0x08000008, 0x00020000,
    0x08020208, 0x00020008, 0x08020000, 0x00000208,
    0x08000000, 0x00000008, 0x08020200, 0x00000200,
    0x00020200, 0x08020000, 0x08020008, 0x00020208,
    0x08000208, 0x00020200, 0x00020000, 0x08000208,
    0x00000008, 0x08020208, 0x00000200, 0x08000000,
    0x08020200, 0x08000000, 0x00020008, 0x00000208,
    0x00020000, 0x08020200, 0x08000200, 0x00000000,
    0x00000200, 0x00020008, 0x08020208, 0x08000200,
    0x08000008, 0x00000200, 0x00000000, 0x08020008,
    0x08000208, 0x00020000, 0x08000000, 0x08020208,
    0x00000008, 0x00020208, 0x00020200, 0x08000008,
    0x08020000, 0x08000208, 0x00000208, 0x08020000,
    0x00020208, 0x00000008, 0x08020008, 0x00020200
}
```

Definition at line 101 of file des.c.

13.121.3.13 const unsigned long SB4[64] [static]**Initial value:**

```
{
    0x00802001, 0x00002081, 0x00002081, 0x00000080,
    0x00802080, 0x00800081, 0x00800001, 0x00002001,
    0x00000000, 0x00802000, 0x00802000, 0x00802081,
    0x00000081, 0x00000000, 0x00800080, 0x00800001,
    0x00000001, 0x00002000, 0x00800000, 0x00802001,
    0x00000080, 0x00800000, 0x00002001, 0x00002080,
    0x00800081, 0x00000001, 0x00002080, 0x00800080,
    0x00002000, 0x00802080, 0x00802081, 0x00000081,
    0x00800080, 0x00800001, 0x00802000, 0x00802081,
    0x00000081, 0x00000000, 0x00000000, 0x00802000,
    0x00002080, 0x00800080, 0x00800081, 0x00000001,
    0x00802001, 0x00002081, 0x00002081, 0x00000080,
    0x00802081, 0x00000081, 0x00000001, 0x00002000,
    0x00800001, 0x00002001, 0x00802080, 0x00800081,
    0x00002001, 0x00002080, 0x00800000, 0x00802001,
    0x00000080, 0x00800000, 0x00002000, 0x00802080
}
```

Definition at line 121 of file des.c.

13.121.3.14 const unsigned long SB5[64] [static]**Initial value:**

```
{
    0x00000100, 0x02080100, 0x02080000, 0x42000100,
    0x00080000, 0x00000100, 0x40000000, 0x02080000,
    0x40080100, 0x00080000, 0x02000100, 0x40080100,
    0x42000100, 0x42080000, 0x00080100, 0x40000000,
}
```

```

0x02000000, 0x40080000, 0x40080000, 0x00000000,
0x40000100, 0x42080100, 0x42080100, 0x02000100,
0x42080000, 0x40000100, 0x00000000, 0x42000000,
0x02080100, 0x02000000, 0x42000000, 0x00080100,
0x00080000, 0x42000100, 0x00000100, 0x02000000,
0x40000000, 0x02080000, 0x42000100, 0x40080100,
0x02000100, 0x40000000, 0x42080000, 0x02080100,
0x40080100, 0x00000100, 0x02000000, 0x42080000,
0x42080100, 0x00080100, 0x42000000, 0x42080100,
0x02080000, 0x00000000, 0x40080000, 0x42000000,
0x00080100, 0x02000100, 0x40000100, 0x00080000,
0x00000000, 0x40080000, 0x02080100, 0x40000100
}

```

Definition at line 141 of file des.c.

13.121.3.15 const unsigned long SB6[64] [static]

Initial value:

```

{
    0x20000010, 0x20400000, 0x00004000, 0x20404010,
    0x20400000, 0x00000010, 0x20404010, 0x00400000,
    0x20004000, 0x00404010, 0x00400000, 0x20000010,
    0x00400010, 0x20004000, 0x20000000, 0x00004010,
    0x00000000, 0x00400010, 0x20004010, 0x00004000,
    0x00404000, 0x20004010, 0x00000010, 0x20400010,
    0x20400010, 0x00000000, 0x00404010, 0x20404000,
    0x00004010, 0x00404000, 0x20404000, 0x20000000,
    0x20004000, 0x00000010, 0x20400010, 0x00404000,
    0x20404010, 0x00400000, 0x00000010, 0x20000010,
    0x00400000, 0x20004000, 0x20000000, 0x00004010,
    0x20000010, 0x20404010, 0x00404000, 0x20400000,
    0x00404010, 0x20404000, 0x00000000, 0x20400010,
    0x00000010, 0x00004000, 0x20400000, 0x00404010,
    0x00004000, 0x00400010, 0x20004010, 0x00000000,
    0x20404000, 0x20000000, 0x00400010, 0x20004010
}

```

Definition at line 161 of file des.c.

13.121.3.16 const unsigned long SB7[64] [static]

Initial value:

```

{
    0x00200000, 0x04200002, 0x04000802, 0x00000000,
    0x00000800, 0x04000802, 0x00200802, 0x04200800,
    0x04200802, 0x00200000, 0x00000000, 0x04000002,
    0x00000002, 0x04000000, 0x04200002, 0x00000802,
    0x04000800, 0x00200802, 0x00200002, 0x04000800,
    0x04000002, 0x04200000, 0x04200800, 0x00200002,
    0x04200000, 0x00000800, 0x00000802, 0x04200802,
    0x00200800, 0x00000002, 0x04000000, 0x00200800,
    0x04000000, 0x00200800, 0x00200000, 0x04000802,
    0x04000802, 0x04200002, 0x04200002, 0x00000002,
    0x00200002, 0x04000000, 0x04000800, 0x00200000,
    0x04200800, 0x00000802, 0x00200802, 0x04200800,
    0x00000802, 0x04000002, 0x04200802, 0x04200000,
    0x00200800, 0x00000000, 0x00000002, 0x04200802,
}

```

```
    0x00000000, 0x00200802, 0x04200000, 0x00000800,  
    0x04000002, 0x04000800, 0x00000800, 0x00200002  
}
```

Definition at line 181 of file des.c.

13.121.3.17 const unsigned long SB8[64] [static]

Initial value:

```
{  
    0x10001040, 0x00001000, 0x00040000, 0x10041040,  
    0x10000000, 0x10001040, 0x00000040, 0x10000000,  
    0x00040040, 0x10040000, 0x10041040, 0x00041000,  
    0x10041000, 0x00041040, 0x00001000, 0x00000040,  
    0x10040000, 0x10000040, 0x10001000, 0x00001040,  
    0x00041000, 0x00040040, 0x10040040, 0x10041000,  
    0x00001040, 0x00000000, 0x00000000, 0x10040040,  
    0x10000040, 0x10001000, 0x00041040, 0x00040000,  
    0x00041040, 0x00040000, 0x10041000, 0x00001000,  
    0x00000040, 0x10040040, 0x00001000, 0x00041040,  
    0x10001000, 0x00000040, 0x10000040, 0x10040000,  
    0x10040040, 0x10000000, 0x00040000, 0x10001040,  
    0x00000000, 0x10041040, 0x00040040, 0x10000040,  
    0x10040000, 0x10001000, 0x10001040, 0x00000000,  
    0x10041040, 0x00041000, 0x00041000, 0x00001040,  
    0x00001040, 0x00040040, 0x10000000, 0x10041000  
}
```

Definition at line 201 of file des.c.

13.122 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/dhm.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/dhm.h"
#include <string.h>
```

Defines

- #define [DHM_MPI_EXPORT](#)(X, n)

Functions

- static [int](#) [dhm_read_bignum](#) ([mpi](#) *X, unsigned char **p, unsigned char *end)
- [int](#) [dhm_read_params](#) ([dhm_context](#) *ctx, unsigned char **p, unsigned char *end)
Parse the ServerKeyExchange parameters.
- [int](#) [dhm_make_params](#) ([dhm_context](#) *ctx, [int](#) x_size, unsigned char *output, [int](#) *olen, [int](#)(*f_rng)(void *), void *p_rng)
Setup and write the ServerKeyExchange parameters.
- [int](#) [dhm_read_public](#) ([dhm_context](#) *ctx, unsigned char *input, [int](#) ilen)
Import the peer's public value G^Y .
- [int](#) [dhm_make_public](#) ([dhm_context](#) *ctx, [int](#) x_size, unsigned char *output, [int](#) olen, [int](#)(*f_rng)(void *), void *p_rng)
Create own private value X and export G^X .
- [int](#) [dhm_calc_secret](#) ([dhm_context](#) *ctx, unsigned char *output, [int](#) *olen)
Derive and export the shared secret $(G^Y)^X \bmod P$.
- void [dhm_free](#) ([dhm_context](#) *ctx)
- [int](#) [dhm_self_test](#) ([int](#) verbose)
Checkup routine.

13.122.1 Define Documentation

13.122.1.1 #define DHM_MPI_EXPORT(X, n)

Value:

```
MPI_CHK( mpi_write_binary( X, p + 2, n ) ); \
    *p++ = (unsigned char)( n >> 8 ); \
    *p++ = (unsigned char)( n ); p += n;
```

Referenced by [dhm_make_params\(\)](#).

13.122.2 Function Documentation

13.122.2.1 `int dhm_calc_secret (dhm_context * ctx, unsigned char * output, int * olen)`

Derive and export the shared secret $(G^Y)^X \bmod P$.

Parameters:

ctx DHM context
output destination buffer
olen number of chars written

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 208 of file dhm.c.

References `dhm_context::GY`, `dhm_context::K`, `MPI_CHK`, `mpi_exp_mod()`, `mpi_size()`, `mpi_write_binary()`, `dhm_context::P`, `dhm_context::RP`, `dhm_context::X`, `XYSSL_ERR_DHM_BAD_INPUT_DATA`, and `XYSSL_ERR_DHM_CALC_SECRET_FAILED`.

Referenced by `main()`, `ssl_parse_client_key_exchange()`, and `ssl_write_client_key_exchange()`.

13.122.2.2 `void dhm_free (dhm_context * ctx)`

Definition at line 234 of file dhm.c.

References `dhm_context::G`, `dhm_context::GX`, `dhm_context::GY`, `dhm_context::K`, `mpi_free()`, `dhm_context::P`, `dhm_context::RP`, and `dhm_context::X`.

Referenced by `main()`, and `ssl_free()`.

13.122.2.3 `int dhm_make_params (dhm_context * ctx, int s_size, unsigned char * output, int * olen, int(*) (void *) f_rng, void * p_rng)`

Setup and write the ServerKeyExchange parameters.

Parameters:

ctx DHM context
x_size private value size in bits
output destination buffer
olen number of chars written
f_rng RNG function
p_rng RNG parameter

Note:

This function assumes that `ctx->P` and `ctx->G` have already been properly set (for example using `mpi_read_string` or `mpi_read_binary`).

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 93 of file dhm.c.

References DHM_MPI_EXPORT, dhm_context::G, dhm_context::GX, dhm_context::len, MPI_CHK, mpi_cmp_mpi(), mpi_exp_mod(), mpi_grow(), mpi_lset(), mpi_shift_r(), mpi_size(), dhm_context::P, mpi::p, dhm_context::RP, dhm_context::X, and XYSSL_ERR_DHM_MAKE_PARAMS_FAILED.

Referenced by main(), and ssl_write_server_key_exchange().

13.122.2.4 `int dhm_make_public(dhm_context * ctx, int s_size, unsigned char * output, int olen, int(*) (void *) f_rng, void * p_rng)`

Create own private value X and export G^X .

Parameters:

ctx DHM context
s_size private value size in bits
output destination buffer
olen must be equal to `ctx->P.len`
f_rng RNG function
p_rng RNG parameter

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 167 of file dhm.c.

References dhm_context::G, dhm_context::GX, dhm_context::len, MPI_CHK, mpi_cmp_mpi(), mpi_exp_mod(), mpi_grow(), mpi_lset(), mpi_shift_r(), mpi_write_binary(), dhm_context::P, mpi::p, dhm_context::RP, dhm_context::X, XYSSL_ERR_DHM_BAD_INPUT_DATA, and XYSSL_ERR_DHM_MAKE_PUBLIC_FAILED.

Referenced by main(), and ssl_write_client_key_exchange().

13.122.2.5 `static int dhm_read_bignum(mpi * X, unsigned char ** p, unsigned char * end) [static]`

Definition at line 37 of file dhm.c.

References mpi_read_binary(), XYSSL_ERR_DHM_BAD_INPUT_DATA, and XYSSL_ERR_DHM_READ_PARAMS_FAILED.

Referenced by dhm_read_params().

13.122.2.6 `int dhm_read_params(dhm_context * ctx, unsigned char ** p, unsigned char * end)`

Parse the ServerKeyExchange parameters.

Parameters:

ctx DHM context
p &(start of input buffer)
end end of buffer

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 63 of file dhm.c.

References dhm_read_bignum(), dhm_context::G, dhm_context::GY, dhm_context::len, mpi_size(), dhm_context::P, and XYSSL_ERR_DHM_BAD_INPUT_DATA.

Referenced by main(), and ssl_parse_server_key_exchange().

13.122.2.7 int dhm_read_public (dhm_context * ctx, unsigned char * input, int ilen)

Import the peer's public value G^Y .

Parameters:

ctx DHM context

input input buffer

ilen size of buffer

Returns:

0 if successful, or an XYSSL_ERR_DHM_XXX error code

Definition at line 150 of file dhm.c.

References dhm_context::GY, dhm_context::len, mpi_read_binary(), XYSSL_ERR_DHM_BAD_INPUT_DATA, and XYSSL_ERR_DHM_READ_PUBLIC_FAILED.

Referenced by main(), and ssl_parse_client_key_exchange().

13.122.2.8 int dhm_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 246 of file dhm.c.

13.123 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/havege.c File Reference

```
#include <string.h>
#include <time.h>
#include "xyssl/config.h"
#include "xyssl/havege.h"
#include "xyssl/timing.h"
```

Defines

- #define [SWAP\(X, Y\)](#) { `int *T = X; X = Y; Y = T;` }
- #define [TST1_ENTER](#) if(PTEST & 1) { PTEST ^= 3; PTEST >>= 1; }
- #define [TST2_ENTER](#) if(PTEST & 1) { PTEST ^= 3; PTEST >>= 1; }
- #define [TST1_LEAVE](#) U1++; }
- #define [TST2_LEAVE](#) U2++; }
- #define [ONE_ITERATION](#)

Functions

- static void [havege_fill](#) (havege_state *hs)
- void [havege_init](#) (havege_state *hs)
HAVEGE initialization.
- `int` [havege_rand](#) (void *p_rng)
HAVEGE rand function.

13.123.1 Define Documentation

13.123.1.1 #define ONE_ITERATION

Definition at line 60 of file havege.c.

Referenced by [havege_fill\(\)](#).

13.123.1.2 #define SWAP(X, Y) { int *T = X; X = Y; Y = T; }

Definition at line 52 of file havege.c.

13.123.1.3 #define TST1_ENTER if(PTEST & 1) { PTEST ^= 3; PTEST >>= 1; }

Definition at line 54 of file havege.c.

13.123.1.4 #define TST1_LEAVE U1++; }

Definition at line 57 of file havege.c.

13.123.1.5 `#define TST2_ENTER if(PTEST & 1) { PTEST ^= 3; PTEST >>= 1;`

Definition at line 55 of file havege.c.

13.123.1.6 `#define TST2_LEAVE U2++; }`

Definition at line 58 of file havege.c.

13.123.2 Function Documentation

13.123.2.1 `static void havege_fill (havege_state * hs) [static]`

Definition at line 154 of file havege.c.

References COLLECT_SIZE, havege_state::offset, ONE_ITERATION, havege_state::PT1, havege_state::PT2, and havege_state::WALK.

Referenced by havege_init(), and havege_rand().

13.123.2.2 `void havege_init (havege_state * hs)`

HAVEGE initialization.

Parameters:

hs HAVEGE state to be initialized

Definition at line 188 of file havege.c.

References havege_fill().

Referenced by generate_AES_key(), generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), initiate_migration_process(), main(), and ssl_test().

13.123.2.3 `int havege_rand (void * p_rng)`

HAVEGE rand function.

Parameters:

rng_st points to an HAVEGE state

Returns:

A random int

Definition at line 198 of file havege.c.

References COLLECT_SIZE, havege_fill(), havege_state::offset, and havege_state::pool.

Referenced by generate_AES_key(), generate_RSA_keys_ciphertext(), generate_RSA_keys_plaintext(), initiate_migration_process(), main(), and ssl_test().

13.124 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/md2.c File Reference

```
#include "xyssl/config.h"
```

13.125 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/md4.c File Reference

```
#include "xyssl/config.h"
```

13.126 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/md5.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/md5.h"
#include <string.h>
#include <stdio.h>
```

Defines

- `#define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))`
- `#define P(a, b, c, d, k, s, t)`
- `#define F(x, y, z) (z ^ (x & (y ^ z)))`
- `#define F(x, y, z) (y ^ (z & (x ^ y)))`
- `#define F(x, y, z) (x ^ y ^ z)`
- `#define F(x, y, z) (y ^ (x | ~z))`

Functions

- void `md5_starts` (`md5_context` *ctx)
MD5 context setup.
- static void `md5_process` (`md5_context` *ctx, unsigned char data[64])
- void `md5_update` (`md5_context` *ctx, unsigned char *input, `int` ilen)
MD5 process buffer.
- void `md5_finish` (`md5_context` *ctx, unsigned char output[16])
MD5 final digest.
- void `md5` (unsigned char *input, `int` ilen, unsigned char output[16])
Output = MD5(input buffer).
- `int` `md5_file` (char *path, unsigned char output[16])
Output = MD5(file contents).
- void `md5_hmac_starts` (`md5_context` *ctx, unsigned char *key, `int` keylen)
MD5 HMAC context setup.
- void `md5_hmac_update` (`md5_context` *ctx, unsigned char *input, `int` ilen)
MD5 HMAC process buffer.
- void `md5_hmac_finish` (`md5_context` *ctx, unsigned char output[16])
MD5 HMAC final digest.
- void `md5_hmac` (unsigned char *key, `int` keylen, unsigned char *input, `int` ilen, unsigned char output[16])
Output = HMAC-MD5(hmac key, input buffer).

- `int md5_self_test (int verbose)`

Checkup routine.

Variables

- static const unsigned char `md5_padding` [64]
- static unsigned char `md5_test_buf` [7][81]
- static const `int md5_test_buflen` [7]
- static const unsigned char `md5_test_sum` [7][16]
- static unsigned char `md5_hmac_test_key` [7][26]
- static const `int md5_hmac_test_keylen` [7]
- static unsigned char `md5_hmac_test_buf` [7][74]
- static const `int md5_hmac_test_buflen` [7]
- static const unsigned char `md5_hmac_test_sum` [7][16]

13.126.1 Define Documentation

13.126.1.1 `#define F(x, y, z) (y ^ (x | ~z))`

13.126.1.2 `#define F(x, y, z) (x ^ y ^ z)`

13.126.1.3 `#define F(x, y, z) (y ^ (z & (x ^ y)))`

13.126.1.4 `#define F(x, y, z) (z ^ (x & (y ^ z)))`

Referenced by `sha2_process()`, and `sha4_process()`.

13.126.1.5 `#define P(a, b, c, d, k, s, t)`

Value:

```
{
    a += F(b, c, d) + X[k] + t; a = S(a, s) + b;
}
```

Referenced by `main()`, `md5_process()`, `rsa_decryption()`, `sha1_process()`, `sha2_process()`, and `sha4_process()`.

13.126.1.6 `#define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))`

13.126.2 Function Documentation

13.126.2.1 `void md5 (unsigned char *input, int ilen, unsigned char output[16])`

Output = MD5(input buffer).

Parameters:

input buffer holding the data

ilen length of the input data

output MD5 checksum result

Definition at line 278 of file md5.c.

References md5_finish(), md5_starts(), and md5_update().

Referenced by initiate_migration_process(), main(), md5_hmac_starts(), md5_self_test(), reply_migration_process(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_finished(), ssl_parse_server_key_exchange(), ssl_write_finished(), ssl_write_server_key_exchange(), and x509_hash().

13.126.2.2 int md5_file (char * *path*, unsigned char *output*[16])

Output = MD5(file contents).

Parameters:

path input file name

output MD5 checksum result

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 292 of file md5.c.

References buf, f, md5_finish(), md5_starts(), and md5_update().

Referenced by md5_wrapper().

13.126.2.3 void md5_finish (md5_context * *ctx*, unsigned char *output*[16])

MD5 final digest.

Parameters:

ctx MD5 context

output MD5 checksum result

Definition at line 250 of file md5.c.

References md5_padding, md5_update(), PUT_ULONG_LE, md5_context::state, and md5_context::total.

Referenced by md5(), md5_file(), md5_hmac_finish(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

13.126.2.4 void md5_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char *output*[16])

Output = HMAC-MD5(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key
input buffer holding the data
ilen length of the input data
output HMAC-MD5 result

Definition at line 378 of file md5.c.

References md5_hmac_finish(), md5_hmac_starts(), and md5_hmac_update().

Referenced by ssl_decrypt_buf(), ssl_encrypt_buf(), and tls1_prf().

13.126.2.5 void md5_hmac_finish (md5_context * ctx, unsigned char output[16])

MD5 HMAC final digest.

Parameters:

ctx HMAC context
output MD5 HMAC checksum result

Definition at line 362 of file md5.c.

References md5_finish(), md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac(), and md5_self_test().

13.126.2.6 void md5_hmac_starts (md5_context * ctx, unsigned char * key, int keylen)

MD5 HMAC context setup.

Parameters:

ctx HMAC context to be initialized
key HMAC secret key
keylen length of the HMAC key

Definition at line 324 of file md5.c.

References md5_context::ipad, md5(), md5_starts(), md5_update(), and md5_context::opad.

Referenced by md5_hmac(), and md5_self_test().

13.126.2.7 void md5_hmac_update (md5_context * ctx, unsigned char * input, int ilen)

MD5 HMAC process buffer.

Parameters:

ctx HMAC context
input buffer holding the data
ilen length of the input data

Definition at line 354 of file md5.c.

References md5_update().

Referenced by md5_hmac(), and md5_self_test().

13.126.2.8 static void md5_process (md5_context * *ctx*, unsigned char *data*[64]) [static]

Definition at line 72 of file md5.c.

References GET_ULONG_LE, P, and md5_context::state.

Referenced by md5_update().

13.126.2.9 int md5_self_test (int *verbose*)

Checksum routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 495 of file md5.c.

References buf, md5(), md5_hmac_finish(), md5_hmac_starts(), md5_hmac_test_buf, md5_hmac_test_buflen, md5_hmac_test_key, md5_hmac_test_keylen, md5_hmac_test_sum, md5_hmac_update(), md5_test_buf, md5_test_buflen, and md5_test_sum.

Referenced by main().

13.126.2.10 void md5_starts (md5_context * *ctx*)

MD5 context setup.

Parameters:

ctx context to be initialized

Definition at line 61 of file md5.c.

References md5_context::state, and md5_context::total.

Referenced by md5(), md5_file(), md5_hmac_finish(), md5_hmac_starts(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_init(), ssl_mac_md5(), ssl_parse_server_key_exchange(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

13.126.2.11 void md5_update (md5_context * *ctx*, unsigned char * *input*, int *ilen*)

MD5 process buffer.

Parameters:

ctx MD5 context

input buffer holding the data

ilen length of the input data

Definition at line 198 of file md5.c.

References md5_context::buffer, md5_process(), and md5_context::total.

Referenced by md5(), md5_file(), md5_finish(), md5_hmac_finish(), md5_hmac_starts(), md5_hmac_update(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_md5(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_read_record(), ssl_write_record(), ssl_write_server_key_exchange(), and x509_des3_decrypt().

13.126.3.4 const int md5_hmac_test_keylen[7] [static]**Initial value:**

```
{
    16, 4, 16, 25, 16, 80, 80
}
```

Definition at line 444 of file md5.c.

Referenced by md5_self_test().

13.126.3.5 const unsigned char md5_hmac_test_sum[7][16] [static]**Initial value:**

```
{
    { 0x92, 0x94, 0x72, 0x7A, 0x36, 0x38, 0xBB, 0x1C,
      0x13, 0xF4, 0x8E, 0xF8, 0x15, 0x8B, 0xFC, 0x9D },
    { 0x75, 0x0C, 0x78, 0x3E, 0x6A, 0xB0, 0xB5, 0x03,
      0xEA, 0xA8, 0x6E, 0x31, 0x0A, 0x5D, 0xB7, 0x38 },
    { 0x56, 0xBE, 0x34, 0x52, 0x1D, 0x14, 0x4C, 0x88,
      0xDB, 0xB8, 0xC7, 0x33, 0xF0, 0xE8, 0xB3, 0xF6 },
    { 0x69, 0x7E, 0xAF, 0x0A, 0xCA, 0x3A, 0x3A, 0xEA,
      0x3A, 0x75, 0x16, 0x47, 0x46, 0xFF, 0xAA, 0x79 },
    { 0x56, 0x46, 0x1E, 0xF2, 0x34, 0x2E, 0xDC, 0x00,
      0xF9, 0xBA, 0xB9, 0x95 },
    { 0x6B, 0x1A, 0xB7, 0xFE, 0x4B, 0xD7, 0xBF, 0x8F,
      0x0B, 0x62, 0xE6, 0xCE, 0x61, 0xB9, 0xD0, 0xCD },
    { 0x6F, 0x63, 0x0F, 0xAD, 0x67, 0xCD, 0xA0, 0xEE,
      0x1F, 0xB1, 0xF5, 0x62, 0xDB, 0x3A, 0xA5, 0x3E }
}
```

Definition at line 474 of file md5.c.

Referenced by md5_self_test().

13.126.3.6 const unsigned char md5_padding[64] [static]**Initial value:**

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 239 of file md5.c.

Referenced by md5_finish().

13.126.3.7 unsigned char md5_test_buf[7][81] [static]**Initial value:**

```

{
    { "" },
    { "a" },
    { "abc" },
    { "message digest" },
    { "abcdefghijklmnopqrstuvwxyz" },
    { "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789" },
    { "1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890" }
}

```

Definition at line 394 of file md5.c.

Referenced by md5_self_test().

13.126.3.8 const int md5_test_buflen[7] [static]

Initial value:

```

{
    0, 1, 3, 14, 26, 62, 80
}

```

Definition at line 406 of file md5.c.

Referenced by md5_self_test().

13.126.3.9 const unsigned char md5_test_sum[7][16] [static]

Initial value:

```

{
    { 0xD4, 0x1D, 0x8C, 0xD9, 0x8F, 0x00, 0xB2, 0x04,
      0xE9, 0x80, 0x09, 0x98, 0xEC, 0xF8, 0x42, 0x7E },
    { 0x0C, 0xC1, 0x75, 0xB9, 0xC0, 0xF1, 0xB6, 0xA8,
      0x31, 0xC3, 0x99, 0xE2, 0x69, 0x77, 0x26, 0x61 },
    { 0x90, 0x01, 0x50, 0x98, 0x3C, 0xD2, 0x4F, 0xB0,
      0xD6, 0x96, 0x3F, 0x7D, 0x28, 0xE1, 0x7F, 0x72 },
    { 0xF9, 0x6B, 0x69, 0x7D, 0x7C, 0xB7, 0x93, 0x8D,
      0x52, 0x5A, 0x2F, 0x31, 0xAA, 0xF1, 0x61, 0xD0 },
    { 0xC3, 0xFC, 0xD3, 0xD7, 0x61, 0x92, 0xE4, 0x00,
      0x7D, 0xFB, 0x49, 0x6C, 0xCA, 0x67, 0xE1, 0x3B },
    { 0xD1, 0x74, 0xAB, 0x98, 0xD2, 0x77, 0xD9, 0xF5,
      0xA5, 0x61, 0x1C, 0x2C, 0x9F, 0x41, 0x9D, 0x9F },
    { 0x57, 0xED, 0xF4, 0xA2, 0x2B, 0xE3, 0xC9, 0x55,
      0xAC, 0x49, 0xDA, 0x2E, 0x21, 0x07, 0xB6, 0x7A }
}

```

Definition at line 411 of file md5.c.

Referenced by md5_self_test().

13.127 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/net.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/net.h"
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <sys/time.h>
#include <unistd.h>
#include <signal.h>
#include <fcntl.h>
#include <netdb.h>
#include <errno.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
```

Functions

- static unsigned short [net_htons](#) (int port)
- [int net_connect](#) (int *fd, char *host, int port)
Initiate a TCP connection with host:port.
- [int net_bind](#) (int *fd, char *bind_ip, int port)
Create a listening socket on bind_ip:port. If bind_ip == NULL, all interfaces are binded.
- static [int net_is_blocking](#) (void)
- [int net_accept](#) (int bind_fd, int *client_fd, void *client_ip)
Accept a connection from a remote client.
- [int net_set_block](#) (int fd)
Set the socket blocking.
- [int net_set_nonblock](#) (int fd)
Set the socket non-blocking.
- void [net_usleep](#) (unsigned long usec)
Portable usleep helper.
- [int net_recv](#) (void *ctx, unsigned char *buf, int len)

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

- `int net_send` (void *ctx, unsigned char *buf, int len)

Write at most 'len' characters. len is updated to reflect the number of characters _not_ written.

- `void net_close` (int fd)

Gracefully shutdown the connection.

13.127.1 Function Documentation

13.127.1.1 `int net_accept` (int bind_fd, int * client_fd, void * client_ip)

Accept a connection from a remote client.

Returns:

0 if successful, XYSSL_ERR_NET_ACCEPT_FAILED, or XYSSL_ERR_NET_WOULD_BLOCK if bind_fd was set to non-blocking and accept() is blocking.

Definition at line 212 of file net.c.

References `int`, `net_is_blocking()`, `XYSSL_ERR_NET_ACCEPT_FAILED`, and `XYSSL_ERR_NET_TRY_AGAIN`.

Referenced by `main()`, and `ssl_test()`.

13.127.1.2 `int net_bind` (int * fd, char * bind_ip, int port)

Create a listening socket on bind_ip:port. If bind_ip == NULL, all interfaces are binded.

Returns:

0 if successful, or one of: XYSSL_ERR_NET_SOCKET_FAILED, XYSSL_ERR_NET_BIND_FAILED, XYSSL_ERR_NET_LISTEN_FAILED

Definition at line 126 of file net.c.

References `net_htons()`, `SOCKET_ERROR`, `XYSSL_ERR_NET_BIND_FAILED`, `XYSSL_ERR_NET_LISTEN_FAILED`, and `XYSSL_ERR_NET_SOCKET_FAILED`.

Referenced by `main()`, and `ssl_test()`.

13.127.1.3 `void net_close` (int fd)

Gracefully shutdown the connection.

Definition at line 338 of file net.c.

Referenced by `main()`, and `ssl_test()`.

13.127.1.4 `int net_connect` (int * fd, char * host, int port)

Initiate a TCP connection with host:port.

Returns:

0 if successful, or one of: XYSSL_ERR_NET_SOCKET_FAILED, XYSSL_ERR_NET_UNKNOWN_HOST, XYSSL_ERR_NET_CONNECT_FAILED

Definition at line 81 of file net.c.

References net_htons(), SOCKET_ERROR, XYSSL_ERR_NET_CONNECT_FAILED, XYSSL_ERR_NET_SOCKET_FAILED, and XYSSL_ERR_NET_UNKNOWN_HOST.

Referenced by main(), and ssl_test().

13.127.1.5 static unsigned short net_htons (int *port*) [static]

Definition at line 67 of file net.c.

References buf.

Referenced by net_bind(), and net_connect().

13.127.1.6 static int net_is_blocking (void) [static]

Definition at line 190 of file net.c.

Referenced by net_accept(), net_recv(), and net_send().

13.127.1.7 int net_recv (void * *ctx*, unsigned char * *buf*, int *len*)

Read at most 'len' characters. len is updated to reflect the actual number of characters read.

Returns:

This function returns the number of bytes received, or a negative error code; XYSSL_ERR_NET_TRY_AGAIN indicates read() is blocking.

Definition at line 277 of file net.c.

References net_is_blocking(), XYSSL_ERR_NET_CONN_RESET, XYSSL_ERR_NET_RECV_FAILED, and XYSSL_ERR_NET_TRY_AGAIN.

Referenced by main(), and ssl_test().

13.127.1.8 int net_send (void * *ctx*, unsigned char * *buf*, int *len*)

Write at most 'len' characters. len is updated to reflect the number of characters *_not_* written.

Returns:

This function returns the number of bytes sent, or a negative error code; XYSSL_ERR_NET_TRY_AGAIN indicates write() is blocking.

Definition at line 309 of file net.c.

References net_is_blocking(), XYSSL_ERR_NET_CONN_RESET, XYSSL_ERR_NET_SEND_FAILED, and XYSSL_ERR_NET_TRY_AGAIN.

Referenced by main(), and ssl_test().

13.127.1.9 int net_set_block (int *fd*)

Set the socket blocking.

Returns:

0 if successful, or a non-zero error code

Definition at line 243 of file net.c.

13.127.1.10 int net_set_nonblock (int *fd*)

Set the socket non-blocking.

Returns:

0 if successful, or a non-zero error code

Definition at line 253 of file net.c.

Referenced by ssl_test().

13.127.1.11 void net_usleep (unsigned long *usec*)

Portable usleep helper.

Note:

Real amount of time slept will not be less than select()'s timeout granularity (typically, 10ms).

Definition at line 266 of file net.c.

13.128 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/padlock.c File Reference

```
#include "xyssl/config.h"  
#include "xyssl/aes.h"  
#include "xyssl/padlock.h"
```

13.129 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/rsa.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/rsa.h"
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
#include "xyssl/sha1.h"
```

Defines

- #define [KEY_LEN](#) 128
- #define [RSA_N](#)
- #define [RSA_E](#) "10001"
- #define [RSA_D](#)
- #define [RSA_P](#)
- #define [RSA_Q](#)
- #define [RSA_DP](#)
- #define [RSA_DQ](#)
- #define [RSA_QP](#)
- #define [PT_LEN](#) 24
- #define [RSA_PT](#)

Functions

- void [rsa_init](#) ([rsa_context](#) *ctx, int padding, int hash_id, int(*f_rng)(void *), void *p_rng)
Initialize an RSA context.
- int [rsa_gen_key](#) ([rsa_context](#) *ctx, int nbits, int exponent)
Generate an RSA keypair.
- int [rsa_check_pubkey](#) ([rsa_context](#) *ctx)
Check a public RSA key.
- int [rsa_check_privkey](#) ([rsa_context](#) *ctx)
Check a private RSA key.
- int [rsa_public](#) ([rsa_context](#) *ctx, unsigned char *input, unsigned char *output)
Do an RSA public key operation.
- int [rsa_private](#) ([rsa_context](#) *ctx, unsigned char *input, unsigned char *output)
Do an RSA private key operation.
- int [rsa_pkcs1_encrypt](#) ([rsa_context](#) *ctx, int mode, int ilen, unsigned char *input, unsigned char *output)
Add the message padding, then do an RSA operation.

- `int rsa_pkcs1_decrypt (rsa_context *ctx, int mode, int *olen, unsigned char *input, unsigned char *output)`

Do an RSA operation, then remove the message padding.

- `int rsa_pkcs1_sign (rsa_context *ctx, int mode, int hash_id, int hashlen, unsigned char *hash, unsigned char *sig)`

Do a private RSA to sign a message digest.

- `int rsa_pkcs1_verify (rsa_context *ctx, int mode, int hash_id, int hashlen, unsigned char *hash, unsigned char *sig)`

Do a public RSA and check the message digest.

- `void rsa_free (rsa_context *ctx)`

Free the components of an RSA key.

- `int rsa_self_test (int verbose)`

Checkup routine.

13.129.1 Define Documentation

13.129.1.1 #define KEY_LEN 128

Definition at line 573 of file rsa.c.

Referenced by `rsa_self_test()`.

13.129.1.2 #define PT_LEN 24

Definition at line 620 of file rsa.c.

Referenced by `rsa_self_test()`.

13.129.1.3 #define RSA_D

Value:

```
"24BF6185468786FDD303083D25E64EFC" \
    "66CA472BC44D253102F8B4A9D3BFA750" \
    "91386C0077937FE33FA325D28855837" \
    "AE1B484A8A9A45F7EE8C0C634F99E8CD" \
    "DF79C5CE07EE72C7F123142198164234" \
    "CABB724CF78B8173B9F880FC86322407" \
    "AF1FEDFDDE2BEB674CA15F3E81A1521E" \
    "071513A1E85B5DFA031F21ECAE91A34D"
```

Definition at line 586 of file rsa.c.

Referenced by `rsa_self_test()`.

13.129.1.4 #define RSA_DP

Value:

```
"C1ACF567564274FB07A0BBAD5D26E298" \
    "3C94D22288ACD763FD8E5600ED4A702D" \
    "F84198A5F06C2E72236AE490C93F07F8" \
    "3CC559CD27BC2D1CA488811730BB5725"
```

Definition at line 605 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.5 #define RSA_DQ

Value:

```
"4959CBF6F8FEF750AEE6977C155579C7" \
    "D8AAEA56749EA28623272E4F7D0592AF" \
    "7C1F1313CAC9471B5C523BFE592F517B" \
    "407A1BD76C164B93DA2D32A383E58357"
```

Definition at line 610 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.6 #define RSA_E "10001"

Definition at line 584 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.7 #define RSA_N

Value:

```
"9292758453063D803DD603D5E777D788" \
    "8ED1D5BF35786190FA2F23EBC0848AEA" \
    "DDA92CA6C3D80B32C4D109BE0F36D6AE" \
    "7130B9CED7ACDF54CFC7555AC14EEBAA" \
    "93A89813FBF3C4F8066D2D800F7C38A8" \
    "1AE31942917403FF4946B0A83D3D3E05" \
    "EE57C6F5F5606FB5D4BC6CD34EE0801A" \
    "5E94BB77B07507233A0BC7BAC8F90F79"
```

Definition at line 575 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.8 #define RSA_P

Value:

```
"C36D0EB7FCD285223CFB5AABA5BDA3D8" \
    "2C01CAD19EA484A87EA4377637E75500" \
    "FCB2005C5C7DD6EC4AC023CDA285D796" \
    "C3D9E75E1EFC42488BB4F1D13AC30A57"
```

Definition at line 595 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.9 #define RSA_PT

Value:

```
"\xAA\xBB\xCC\x03\x02\x01\x00\xff\xff\xff\xff" \
"\x11\x22\x33\x0A\x0B\x0C\xCC\xDD\xDD\xDD\xDD"
```

Definition at line 621 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.10 #define RSA_Q

Value:

```
"C000DF51A7C77AE8D7C7370C1FF55B69" \
"E211C2B9E5DB1ED0BF61D0D9899620F4" \
"910E4168387E3C30AA1E00C339A79508" \
"8452DD96A9A5EA5D9DCA68DA636032AF"
```

Definition at line 600 of file rsa.c.

Referenced by rsa_self_test().

13.129.1.11 #define RSA_QP

Value:

```
"9AE7FBC99546432DF71896FC239EADAE" \
"F38D18D2B2F0E2DD275AA977E2BF4411" \
"F5A3B2A5D33605AEBBCCBA7FEB9F2D2F" \
"A74206CEC169D74BF5A8C50D6F48EA08"
```

Definition at line 615 of file rsa.c.

Referenced by rsa_self_test().

13.129.2 Function Documentation

13.129.2.1 int rsa_check_privkey (rsa_context * ctx)

Check a private RSA key.

Parameters:

ctx RSA context to be checked

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Definition at line 152 of file rsa.c.

References `rsa_context::D`, `rsa_context::E`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_init()`, `mpi_mod_mpi()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_check_pubkey()`, and `XYSSL_ERR_RSA_KEY_CHECK_FAILED`.

Referenced by `rsa_decryption()`, `rsa_self_test()`, and `x509parse_key()`.

13.129.2.2 `int rsa_check_pubkey (rsa_context * ctx)`

Check a public RSA key.

Parameters:

ctx RSA context to be checked

Returns:

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Definition at line 132 of file rsa.c.

References `rsa_context::E`, `mpi_msb()`, `rsa_context::N`, `mpi::p`, and `XYSSL_ERR_RSA_KEY_CHECK_FAILED`.

Referenced by `rsa_check_privkey()`, `rsa_encryption()`, `rsa_self_test()`, and `x509parse_crt()`.

13.129.2.3 `void rsa_free (rsa_context * ctx)`

Free the components of an RSA key.

Definition at line 558 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `mpi_free()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_context::RN`, `rsa_context::RP`, and `rsa_context::RQ`.

Referenced by `main()`, `rsa_gen_key()`, `rsa_self_test()`, `ssl_test()`, `x509_free()`, `x509_self_test()`, and `x509parse_key()`.

13.129.2.4 `int rsa_gen_key (rsa_context * ctx, int nbits, int exponent)`

Generate an RSA keypair.

Parameters:

ctx RSA context that will hold the key

nbits size of the public key in bits

exponent public exponent (e.g., 65537)

Note:

[rsa_init\(\)](#) must be called beforehand to setup the RSA context (especially `f_rng` and `p_rng`).

Returns:

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Definition at line 60 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `rsa_context::f_rng`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_int()`, `mpi_cmp_mpi()`, `mpi_free()`, `mpi_gcd()`, `mpi_gen_prime()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_lset()`, `mpi_mod_mpi()`, `mpi_msb()`, `mpi_mul_mpi()`, `mpi_sub_int()`, `mpi_swap()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::p_rng`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_KEY_GEN_FAILED`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, and `main()`.

13.129.2.5 `void rsa_init (rsa_context * ctx, int padding, int hash_id, int(*) (void *) f_rng, void * p_rng)`

Initialize an RSA context.

Parameters:

ctx RSA context to be initialized
padding `RSA_PKCS_V15` or `RSA_PKCS_V21`
hash_id `RSA_PKCS_V21` hash identifier
f_rng RNG function
p_rng RNG parameter

Note:

The *hash_id* parameter is actually ignored when using `RSA_PKCS_V15` padding.
 Currently (xyssl-0.8), `RSA_PKCS_V21` padding is not supported.

Definition at line 40 of file rsa.c.

References `rsa_context::f_rng`, `rsa_context::hash_id`, `rsa_context::p_rng`, and `rsa_context::padding`.

Referenced by `generate_RSA_keys_ciphertext()`, `generate_RSA_keys_plaintext()`, `main()`, `rsa_decryption()`, and `rsa_encryption()`.

13.129.2.6 `int rsa_pkcs1_decrypt (rsa_context * ctx, int mode, int * olen, unsigned char * input, unsigned char * output)`

Do an RSA operation, then remove the message padding.

Parameters:

ctx RSA context
mode `RSA_PUBLIC` or `RSA_PRIVATE`
input buffer holding the encrypted data
output buffer that will hold the plaintext
olen will contain the plaintext length

Returns:

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Note:

The output buffer must be as large as the size of `ctx->N` (eg. 128 bytes if RSA-1024 is used).

Definition at line 326 of file rsa.c.

References `buf`, `int`, `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_decryption()`, `RSA_private_decrypt()`, `RSA_public_decrypt()`, `rsa_self_test()`, and `ssl_parse_client_key_exchange()`.

13.129.2.7 `int rsa_pkcs1_encrypt (rsa_context * ctx, int mode, int ilen, unsigned char * input, unsigned char * output)`

Add the message padding, then do an RSA operation.

Parameters:

ctx RSA context
mode `RSA_PUBLIC` or `RSA_PRIVATE`
ilen contains the the plaintext length
input buffer holding the data to be encrypted
output buffer that will hold the ciphertext

Returns:

0 if successful, or an `XYSSL_ERR_RSA_XXX` error code

Note:

The output buffer must be as large as the size of `ctx->N` (eg. 128 bytes if RSA-1024 is used).

Definition at line 280 of file rsa.c.

References `rsa_context::len`, `rsa_context::padding`, `RSA_CRYPT`, `RSA_PKCS_V15`, `rsa_private()`, `rsa_public()`, `RSA_PUBLIC`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_INVALID_PADDING`.

Referenced by `rsa_encryption()`, `RSA_private_encrypt()`, `RSA_public_encrypt()`, `rsa_self_test()`, and `ssl_write_client_key_exchange()`.

13.129.2.8 `int rsa_pkcs1_sign (rsa_context * ctx, int mode, int hash_id, int hashlen, unsigned char * hash, unsigned char * sig)`

Do a private RSA to sign a message digest.

Parameters:

ctx RSA context
mode `RSA_PUBLIC` or `RSA_PRIVATE`
hash_id `RSA_RAW`, `RSA_MD{2,4,5}` or `RSA_SHA{1,256}`
hashlen message digest length (for `RSA_RAW` only)
hash buffer holding the message digest
sig buffer that will hold the ciphertext

Returns:

0 if the signing operation was successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 379 of file rsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, rsa_context::len, rsa_context::padding, RSA_MD2, RSA_MD4, RSA_MD5, RSA_PKCS_V15, rsa_private(), rsa_public(), RSA_PUBLIC, RSA_RAW, RSA_SHA1, RSA_SIGN, XYSSL_ERR_RSA_BAD_INPUT_DATA, and XYSSL_ERR_RSA_INVALID_PADDING.

Referenced by main(), rsa_self_test(), ssl_write_certificate_verify(), and ssl_write_server_key_exchange().

13.129.2.9 **int rsa_pkcs1_verify (rsa_context * ctx, int mode, int hash_id, int hashlen, unsigned char * hash, unsigned char * sig)**

Do a public RSA and check the message digest.

Parameters:

ctx points to an RSA public key
mode RSA_PUBLIC or RSA_PRIVATE
hash_id RSA_RAW, RSA_MD{2,4,5} or RSA_SHA{1,256}
hashlen message digest length (for RSA_RAW only)
hash buffer holding the message digest
sig buffer holding the ciphertext

Returns:

0 if the verify operation was successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The "sig" buffer must be as large as the size of ctx->N (eg. 128 bytes if RSA-1024 is used).

Definition at line 468 of file rsa.c.

References ASN1_HASH_MDX, ASN1_HASH_SHA1, buf, int, rsa_context::len, rsa_context::padding, RSA_MD2, RSA_MD4, RSA_MD5, RSA_PKCS_V15, rsa_private(), rsa_public(), RSA_PUBLIC, RSA_RAW, RSA_SHA1, RSA_SIGN, XYSSL_ERR_RSA_BAD_INPUT_DATA, XYSSL_ERR_RSA_INVALID_PADDING, and XYSSL_ERR_RSA_VERIFY_FAILED.

Referenced by main(), rsa_self_test(), ssl_parse_certificate_verify(), ssl_parse_server_key_exchange(), and x509parse_verify().

13.129.2.10 **int rsa_private (rsa_context * ctx, unsigned char * input, unsigned char * output)**

Do an RSA private key operation.

Parameters:

ctx RSA context

input input buffer
output output buffer

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Note:

The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 221 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::len`, `mpi_add_mpi()`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_mod_mpi()`, `mpi_mul_mpi()`, `mpi_read_binary()`, `mpi_sub_mpi()`, `mpi_write_binary()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_context::RN`, `rsa_context::RP`, `rsa_context::RQ`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_PRIVATE_FAILED`.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

13.129.2.11 int rsa_public (rsa_context * ctx, unsigned char * input, unsigned char * output)

Do an RSA public key operation.

Parameters:

ctx RSA context
input input buffer
output output buffer

Returns:

0 if successful, or an XYSSL_ERR_RSA_XXX error code

Note:

This function does NOT take care of message padding. Also, be sure to set `input[0] = 0`.
 The input and output buffers must be large enough (eg. 128 bytes if RSA-1024 is used).

Definition at line 187 of file rsa.c.

References `rsa_context::E`, `rsa_context::len`, `MPI_CHK`, `mpi_cmp_mpi()`, `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_read_binary()`, `mpi_write_binary()`, `rsa_context::N`, `rsa_context::RN`, `XYSSL_ERR_RSA_BAD_INPUT_DATA`, and `XYSSL_ERR_RSA_PUBLIC_FAILED`.

Referenced by `main()`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, and `rsa_pkcs1_verify()`.

13.129.2.12 int rsa_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 627 of file rsa.c.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `KEY_LEN`, `rsa_context::len`, `mpi_read_string()`, `rsa_context::N`, `rsa_context::P`, `PT_LEN`, `rsa_context::Q`, `rsa_context::QP`, `rsa_check_privkey()`, `rsa_check_pubkey()`, `RSA_D`, `RSA_DP`, `RSA_DQ`, `RSA_E`, `rsa_free()`, `RSA_N`, `RSA_P`, `rsa_pkcs1_decrypt()`, `rsa_pkcs1_encrypt()`, `rsa_pkcs1_sign()`, `rsa_pkcs1_verify()`, `RSA_PRIVATE`, `RSA_PT`, `RSA_PUBLIC`, `RSA_Q`, `RSA_QP`, `RSA_SHA1`, and `sha1()`.

Referenced by `main()`.

13.130 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/sha1.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/sha1.h"
#include <string.h>
#include <stdio.h>
```

Defines

- `#define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))`
- `#define R(t)`
- `#define P(a, b, c, d, e, x)`
- `#define F(x, y, z) (z ^ (x & (y ^ z)))`
- `#define K 0x5A827999`
- `#define F(x, y, z) (x ^ y ^ z)`
- `#define K 0x6ED9EBA1`
- `#define F(x, y, z) ((x & y) | (z & (x | y)))`
- `#define K 0x8F1BBCDC`
- `#define F(x, y, z) (x ^ y ^ z)`
- `#define K 0xCA62C1D6`

Functions

- void `sha1_starts` (`sha1_context` *ctx)
SHA-1 context setup.
- static void `sha1_process` (`sha1_context` *ctx, unsigned char data[64])
- void `sha1_update` (`sha1_context` *ctx, unsigned char *input, int ilen)
SHA-1 process buffer.
- void `sha1_finish` (`sha1_context` *ctx, unsigned char output[20])
SHA-1 final digest.
- void `sha1` (unsigned char *input, int ilen, unsigned char output[20])
Output = SHA-1(input buffer).
- int `sha1_file` (char *path, unsigned char output[20])
Output = SHA-1(file contents).
- void `sha1_hmac_starts` (`sha1_context` *ctx, unsigned char *key, int keylen)
SHA-1 HMAC context setup.
- void `sha1_hmac_update` (`sha1_context` *ctx, unsigned char *input, int ilen)
SHA-1 HMAC process buffer.
- void `sha1_hmac_finish` (`sha1_context` *ctx, unsigned char output[20])

SHA-1 HMAC final digest.

- void `sha1_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char output[20])

Output = HMAC-SHA-1(hmac key, input buffer).

- int `sha1_self_test` (int verbose)

Checkup routine.

Variables

- static const unsigned char `sha1_padding` [64]
- static unsigned char `sha1_test_buf` [3][57]
- static const int `sha1_test_buflen` [3]
- static const unsigned char `sha1_test_sum` [3][20]
- static unsigned char `sha1_hmac_test_key` [7][26]
- static const int `sha1_hmac_test_keylen` [7]
- static unsigned char `sha1_hmac_test_buf` [7][74]
- static const int `sha1_hmac_test_buflen` [7]
- static const unsigned char `sha1_hmac_test_sum` [7][20]

13.130.1 Define Documentation

13.130.1.1 `#define F(x, y, z) (x ^ y ^ z)`

13.130.1.2 `#define F(x, y, z) ((x & y) | (z & (x | y)))`

13.130.1.3 `#define F(x, y, z) (x ^ y ^ z)`

13.130.1.4 `#define F(x, y, z) (z ^ (x & (y ^ z)))`

13.130.1.5 `#define K 0xCA62C1D6`

13.130.1.6 `#define K 0x8F1BBCDC`

13.130.1.7 `#define K 0x6ED9EBA1`

13.130.1.8 `#define K 0x5A827999`

Referenced by `sha4_process()`.

13.130.1.9 `#define P(a, b, c, d, e, x)`

Value:

```
{
    e += S(a, 5) + F(b, c, d) + K + x; b = S(b, 30);
}
```

13.130.1.10 #define R(t)**Value:**

```
(
    temp = W[(t - 3) & 0x0F] ^ W[(t - 8) & 0x0F] ^
           W[(t - 14) & 0x0F] ^ W[t & 0x0F],
    ( W[t & 0x0F] = S(temp, 1) )
)
```

Referenced by `mpi_is_prime()`, `sha1_process()`, and `sha2_process()`.

13.130.1.11 #define S(x, n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))**13.130.2 Function Documentation****13.130.2.1 void sha1 (unsigned char * *input*, int *ilen*, unsigned char *output*[20])**

Output = SHA-1(*input* buffer).

Parameters:

input buffer holding the data
ilen length of the input data
output SHA-1 checksum result

Definition at line 313 of file `sha1.c`.

References `sha1_finish()`, `sha1_starts()`, and `sha1_update()`.

Referenced by `main()`, `rsa_self_test()`, `sha1_hmac_starts()`, `ssl_calc_verify()`, `ssl_derive_keys()`, `ssl_mac_sha1()`, `ssl_parse_finished()`, `ssl_parse_server_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_key_exchange()`, and `x509_hash()`.

13.130.2.2 int sha1_file (char * *path*, unsigned char *output*[20])

Output = SHA-1(*file* contents).

Parameters:

path input file name
output SHA-1 checksum result

Returns:

0 if successful, 1 if `fopen` failed, or 2 if `fread` failed

Definition at line 327 of file `sha1.c`.

References `buf`, `f`, `sha1_finish()`, `sha1_starts()`, and `sha1_update()`.

Referenced by `main()`, and `sha1_wrapper()`.

13.130.2.3 void sha1_finish (sha1_context * ctx, unsigned char output[20])

SHA-1 final digest.

Parameters:

ctx SHA-1 context

output SHA-1 checksum result

Definition at line 284 of file sha1.c.

References PUT_ULONG_BE, sha1_padding, sha1_update(), sha1_context::state, and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_hmac_finish(), sha1_self_test(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

13.130.2.4 void sha1_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char output[20])

Output = HMAC-SHA-1(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key

input buffer holding the data

ilen length of the input data

output HMAC-SHA-1 result

Definition at line 413 of file sha1.c.

References sha1_hmac_finish(), sha1_hmac_starts(), and sha1_hmac_update().

Referenced by ssl_decrypt_buf(), ssl_encrypt_buf(), and tls1_prf().

13.130.2.5 void sha1_hmac_finish (sha1_context * ctx, unsigned char output[20])

SHA-1 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-1 HMAC checksum result

Definition at line 397 of file sha1.c.

References sha1_context::opad, sha1_finish(), sha1_starts(), and sha1_update().

Referenced by sha1_hmac(), and sha1_self_test().

13.130.2.6 void sha1_hmac_starts (sha1_context * ctx, unsigned char * key, int keylen)

SHA-1 HMAC context setup.

Parameters:

ctx HMAC context to be initialized

key HMAC secret key

keylen length of the HMAC key

Definition at line 359 of file sha1.c.

References sha1_context::ipad, sha1_context::opad, sha1(), sha1_starts(), and sha1_update().

Referenced by sha1_hmac(), and sha1_self_test().

13.130.2.7 void sha1_hmac_update (sha1_context * ctx, unsigned char * input, int ilen)

SHA-1 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

Definition at line 389 of file sha1.c.

References sha1_update().

Referenced by sha1_hmac(), and sha1_self_test().

13.130.2.8 static void sha1_process (sha1_context * ctx, unsigned char data[64]) [static]

Definition at line 73 of file sha1.c.

References GET_ULONG_BE, P, R, and sha1_context::state.

Referenced by sha1_update().

13.130.2.9 int sha1_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 521 of file sha1.c.

References buf, sha1_finish(), sha1_hmac_finish(), sha1_hmac_starts(), sha1_hmac_test_buf, sha1_hmac_test_buflen, sha1_hmac_test_key, sha1_hmac_test_keylen, sha1_hmac_test_sum, sha1_hmac_update(), sha1_starts(), sha1_test_buf, sha1_test_buflen, sha1_test_sum, and sha1_update().

Referenced by main().

13.130.2.10 void sha1_starts (sha1_context * ctx)

SHA-1 context setup.

Parameters:

ctx context to be initialized

Definition at line 61 of file sha1.c.

References sha1_context::state, and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_hmac_finish(), sha1_hmac_starts(), sha1_self_test(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_init(), ssl_mac_sha1(), ssl_parse_server_key_exchange(), and ssl_write_server_key_exchange().

13.130.2.11 void sha1_update (sha1_context * ctx, unsigned char * input, int ilen)

SHA-1 process buffer.

Parameters:

ctx SHA-1 context

input buffer holding the data

ilen length of the input data

Definition at line 232 of file sha1.c.

References sha1_context::buffer, sha1_process(), and sha1_context::total.

Referenced by sha1(), sha1_file(), sha1_finish(), sha1_hmac_finish(), sha1_hmac_starts(), sha1_hmac_update(), sha1_self_test(), ssl_calc_finished(), ssl_calc_verify(), ssl_derive_keys(), ssl_mac_sha1(), ssl_parse_client_hello(), ssl_parse_server_key_exchange(), ssl_read_record(), ssl_write_record(), and ssl_write_server_key_exchange().

13.130.3 Variable Documentation**13.130.3.1 unsigned char sha1_hmac_test_buf[7][74] [static]****Initial value:**

```
{
    { "Hi There" },
    { "what do ya want for nothing?" },
    { "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD"
      "\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD\xDD" },
    { "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD"
      "\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD\xCD" },
    { "Test With Truncation" },
    { "Test Using Larger Than Block-Size Key - Hash Key First" },
    { "Test Using Larger Than Block-Size Key and Larger"
      " Than One Block-Size Data" }
}
```

Definition at line 475 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.2 const int sha1_hmac_test_buflen[7] [static]**Initial value:**

```
{
    8, 28, 50, 50, 20, 54, 73
}
```

Definition at line 495 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.3 unsigned char sha1_hmac_test_key[7][26] [static]**Initial value:**

```
{
    { "\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B"
      "\x0B\x0B\x0B\x0B" },
    { "Jefe" },
    { "\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA"
      "\xAA\xAA\xAA\xAA" },
    { "\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0A\x0B\x0C\x0D\x0E\x0F\x10"
      "\x11\x12\x13\x14\x15\x16\x17\x18\x19" },
    { "\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C"
      "\x0C\x0C\x0C\x0C" },
    { "" },
    { "" }
}
```

Definition at line 455 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.4 const int sha1_hmac_test_keylen[7] [static]**Initial value:**

```
{
    20, 4, 20, 25, 20, 80, 80
}
```

Definition at line 470 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.5 const unsigned char sha1_hmac_test_sum[7][20] [static]**Initial value:**

```
{
    { 0xB6, 0x17, 0x31, 0x86, 0x55, 0x05, 0x72, 0x64, 0xE2, 0x8B,
      0xC0, 0xB6, 0xFB, 0x37, 0x8C, 0x8E, 0xF1, 0x46, 0xBE, 0x00 },
    { 0xEF, 0xFC, 0xDF, 0x6A, 0xE5, 0xEB, 0x2F, 0xA2, 0xD2, 0x74,
      0x16, 0xD5, 0xF1, 0x84, 0xDF, 0x9C, 0x25, 0x9A, 0x7C, 0x79 },
}
```

```

    { 0x12, 0x5D, 0x73, 0x42, 0xB9, 0xAC, 0x11, 0xCD, 0x91, 0xA3,
      0x9A, 0xF4, 0x8A, 0xA1, 0x7B, 0x4F, 0x63, 0xF1, 0x75, 0xD3 },
    { 0x4C, 0x90, 0x07, 0xF4, 0x02, 0x62, 0x50, 0xC6, 0xBC, 0x84,
      0x14, 0xF9, 0xBF, 0x50, 0xC8, 0x6C, 0x2D, 0x72, 0x35, 0xDA },
    { 0x4C, 0x1A, 0x03, 0x42, 0x4B, 0x55, 0xE0, 0x7F, 0xE7, 0xF2,
      0x7B, 0xE1 },
    { 0xAA, 0x4A, 0xE5, 0xE1, 0x52, 0x72, 0xD0, 0x0E, 0x95, 0x70,
      0x56, 0x37, 0xCE, 0x8A, 0x3B, 0x55, 0xED, 0x40, 0x21, 0x12 },
    { 0xE8, 0xE9, 0x9D, 0x0F, 0x45, 0x23, 0x7D, 0x78, 0x6D, 0x6B,
      0xBA, 0xA7, 0x96, 0x5C, 0x78, 0x08, 0xBB, 0xFF, 0x1A, 0x91 }
}

```

Definition at line 500 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.6 const unsigned char sha1_padding[64] [static]

Initial value:

```

{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}

```

Definition at line 273 of file sha1.c.

Referenced by sha1_finish().

13.130.3.7 unsigned char sha1_test_buf[3][57] [static]

Initial value:

```

{
    { "abc" },
    { "abcdcbdecdefdefgefghfghighijhijkijklklmklmnlmnomnopnopq" },
    { "" }
}

```

Definition at line 430 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.8 const int sha1_test_buflen[3] [static]

Initial value:

```

{
    3, 56, 1000
}

```

Definition at line 437 of file sha1.c.

Referenced by sha1_self_test().

13.130.3.9 const unsigned char sha1_test_sum[3][20] [static]**Initial value:**

```
{
    { 0xA9, 0x99, 0x3E, 0x36, 0x47, 0x06, 0x81, 0x6A, 0xBA, 0x3E,
      0x25, 0x71, 0x78, 0x50, 0xC2, 0x6C, 0x9C, 0xD0, 0xD8, 0x9D },
    { 0x84, 0x98, 0x3E, 0x44, 0x1C, 0x3B, 0xD2, 0x6E, 0xBA, 0xAE,
      0x4A, 0xA1, 0xF9, 0x51, 0x29, 0xE5, 0xE5, 0x46, 0x70, 0xF1 },
    { 0x34, 0xAA, 0x97, 0x3C, 0xD4, 0xC4, 0xDA, 0xA4, 0xF6, 0x1E,
      0xEB, 0x2B, 0xDB, 0xAD, 0x27, 0x31, 0x65, 0x34, 0x01, 0x6F }
}
```

Definition at line 442 of file sha1.c.

Referenced by sha1_self_test().

13.131 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/sha2.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/sha2.h"
#include <string.h>
#include <stdio.h>
```

Defines

- #define [SHR](#)(x, n) ((x & 0xFFFFFFFF) >> n)
- #define [ROTR](#)(x, n) (SHR(x,n) | (x << (32 - n)))
- #define [S0](#)(x) (ROTR(x, 7) ^ ROTR(x,18) ^ SHR(x, 3))
- #define [S1](#)(x) (ROTR(x,17) ^ ROTR(x,19) ^ SHR(x,10))
- #define [S2](#)(x) (ROTR(x, 2) ^ ROTR(x,13) ^ ROTR(x,22))
- #define [S3](#)(x) (ROTR(x, 6) ^ ROTR(x,11) ^ ROTR(x,25))
- #define [F0](#)(x, y, z) ((x & y) | (z & (x | y)))
- #define [F1](#)(x, y, z) (z ^ (x & (y ^ z)))
- #define [R](#)(t)
- #define [P](#)(a, b, c, d, e, [f](#), g, h, x, [K](#))

Functions

- void [sha2_starts](#) ([sha2_context](#) *ctx, [int](#) is224)
SHA-256 context setup.
- static void [sha2_process](#) ([sha2_context](#) *ctx, unsigned char data[64])
- void [sha2_update](#) ([sha2_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-256 process buffer.
- void [sha2_finish](#) ([sha2_context](#) *ctx, unsigned char output[32])
SHA-256 final digest.
- void [sha2](#) (unsigned char *input, [int](#) ilen, unsigned char output[32], [int](#) is224)
Output = SHA-256(input buffer).
- [int](#) [sha2_file](#) (char *path, unsigned char output[32], [int](#) is224)
Output = SHA-256(file contents).
- void [sha2_hmac_starts](#) ([sha2_context](#) *ctx, unsigned char *key, [int](#) keylen, [int](#) is224)
SHA-256 HMAC context setup.
- void [sha2_hmac_update](#) ([sha2_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-256 HMAC process buffer.
- void [sha2_hmac_finish](#) ([sha2_context](#) *ctx, unsigned char output[32])
SHA-256 HMAC final digest.

- void `sha2_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char output[32], int is224)

Output = HMAC-SHA-256(hmac key, input buffer).

- int `sha2_self_test` (int verbose)

Checkup routine.

Variables

- static const unsigned char `sha2_padding` [64]
- static unsigned char `sha2_test_buf` [3][57]
- static const int `sha2_test_buflen` [3]
- static const unsigned char `sha2_test_sum` [6][32]
- static unsigned char `sha2_hmac_test_key` [7][26]
- static const int `sha2_hmac_test_keylen` [7]
- static unsigned char `sha2_hmac_test_buf` [7][153]
- static const int `sha2_hmac_test_buflen` [7]
- static const unsigned char `sha2_hmac_test_sum` [14][32]

13.131.1 Define Documentation

13.131.1.1 `#define F0(x, y, z) ((x & y) | (z & (x | y)))`

13.131.1.2 `#define F1(x, y, z) (z ^ (x & (y ^ z)))`

13.131.1.3 `#define P(a, b, c, d, e, f, g, h, x, K)`

Value:

```
{
    temp1 = h + S3(e) + F1(e, f, g) + K + x;      \
    temp2 = S2(a) + F0(a, b, c);                  \
    d += temp1; h = temp1 + temp2;                 \
}
```

13.131.1.4 `#define R(t)`

Value:

```
(
    W[t] = S1(W[t - 2]) + W[t - 7] +              \
    S0(W[t - 15]) + W[t - 16]                      \
)
```

13.131.1.5 `#define ROTR(x, n) (SHR(x,n) | (x << (32 - n)))`

13.131.1.6 `#define S0(x) (ROTR(x, 7) ^ ROTR(x,18) ^ SHR(x, 3))`

Referenced by `sha4_process()`.

13.131.1.7 `#define S1(x) (ROTR(x,17) ^ ROTR(x,19) ^ SHR(x,10))`

Referenced by sha4_process(), and tls1_prf().

13.131.1.8 `#define S2(x) (ROTR(x, 2) ^ ROTR(x,13) ^ ROTR(x,22))`

Referenced by tls1_prf().

13.131.1.9 `#define S3(x) (ROTR(x, 6) ^ ROTR(x,11) ^ ROTR(x,25))`

13.131.1.10 `#define SHR(x, n) ((x & 0xFFFFFFFF) >> n)`

13.131.2 Function Documentation

13.131.2.1 `void sha2 (unsigned char *input, int ilen, unsigned char output[32], int is224)`

Output = SHA-256(input buffer).

Parameters:

input buffer holding the data

ilen length of the input data

output SHA-224/256 checksum result

is224 0 = use SHA256, 1 = use SHA224

Definition at line 314 of file sha2.c.

References sha2_finish(), sha2_starts(), and sha2_update().

Referenced by main(), and sha2_hmac_starts().

13.131.2.2 `int sha2_file (char *path, unsigned char output[32], int is224)`

Output = SHA-256(file contents).

Parameters:

path input file name

output SHA-224/256 checksum result

is224 0 = use SHA256, 1 = use SHA224

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 329 of file sha2.c.

References buf, f, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by sha2_wrapper().

13.131.2.3 void sha2_finish (sha2_context * ctx, unsigned char output[32])

SHA-256 final digest.

Parameters:

ctx SHA-256 context

output SHA-224/256 checksum result

Definition at line 280 of file sha2.c.

References sha2_context::is224, PUT_ULONG_BE, sha2_padding, sha2_update(), sha2_context::state, and sha2_context::total.

Referenced by aes_en_de(), main(), sha2(), sha2_file(), sha2_hmac_finish(), and sha2_self_test().

13.131.2.4 void sha2_hmac (unsigned char * key, int keylen, unsigned char * input, int ilen, unsigned char output[32], int is224)

Output = HMAC-SHA-256(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key

input buffer holding the data

ilen length of the input data

output HMAC-SHA-224/256 result

is224 0 = use SHA256, 1 = use SHA224

Definition at line 420 of file sha2.c.

References sha2_hmac_finish(), sha2_hmac_starts(), and sha2_hmac_update().

13.131.2.5 void sha2_hmac_finish (sha2_context * ctx, unsigned char output[32])

SHA-256 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-224/256 HMAC checksum result

Definition at line 400 of file sha2.c.

References sha2_context::is224, sha2_context::opad, sha2_finish(), sha2_starts(), and sha2_update().

Referenced by aes_en_de(), main(), sha2_hmac(), and sha2_self_test().

13.131.2.6 void sha2_hmac_starts (sha2_context * ctx, unsigned char * key, int keylen, int is224)

SHA-256 HMAC context setup.

Parameters:

ctx HMAC context to be initialized
key HMAC secret key
keylen length of the HMAC key
is224 0 = use SHA256, 1 = use SHA224

Definition at line 361 of file sha2.c.

References sha2_context::ipad, sha2_context::opad, sha2(), sha2_starts(), and sha2_update().

Referenced by aes_en_de(), main(), sha2_hmac(), and sha2_self_test().

13.131.2.7 void sha2_hmac_update (sha2_context * ctx, unsigned char * input, int ilen)

SHA-256 HMAC process buffer.

Parameters:

ctx HMAC context
input buffer holding the data
ilen length of the input data

Definition at line 392 of file sha2.c.

References sha2_update().

Referenced by aes_en_de(), main(), sha2_hmac(), and sha2_self_test().

13.131.2.8 static void sha2_process (sha2_context * ctx, unsigned char data[64]) [static]

Definition at line 94 of file sha2.c.

References F, GET_ULONG_BE, P, R, and sha2_context::state.

Referenced by sha2_update().

13.131.2.9 int sha2_self_test (int verbose)

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 599 of file sha2.c.

References buf, sha2_finish(), sha2_hmac_finish(), sha2_hmac_starts(), sha2_hmac_test_buf, sha2_hmac_test_buflen, sha2_hmac_test_key, sha2_hmac_test_keylen, sha2_hmac_test_sum, sha2_hmac_update(), sha2_starts(), sha2_test_buf, sha2_test_buflen, sha2_test_sum, and sha2_update().

Referenced by main().

Definition at line 507 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.2 `const int sha2_hmac_test_buflen[7] [static]`

Initial value:

```
{
    8, 28, 50, 50, 20, 54, 152
}
```

Definition at line 528 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.3 `unsigned char sha2_hmac_test_key[7][26] [static]`

Initial value:

```
{
    { "\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B\x0B"
      "\x0B\x0B\x0B\x0B" },
    { "Jefe" },
    { "\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA\xAA"
      "\xAA\xAA\xAA\xAA" },
    { "\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0A\x0B\x0C\x0D\x0E\x0F\x10"
      "\x11\x12\x13\x14\x15\x16\x17\x18\x19" },
    { "\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C\x0C"
      "\x0C\x0C\x0C\x0C" },
    { "" },
    { "" }
}
```

Definition at line 487 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.4 `const int sha2_hmac_test_keylen[7] [static]`

Initial value:

```
{
    20, 4, 20, 25, 20, 131, 131
}
```

Definition at line 502 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.5 `const unsigned char sha2_hmac_test_sum[14][32] [static]`

Definition at line 533 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.6 const unsigned char sha2_padding[64] [static]**Initial value:**

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 269 of file sha2.c.

Referenced by sha2_finish().

13.131.3.7 unsigned char sha2_test_buf[3][57] [static]**Initial value:**

```
{
    { "abc" },
    { "abcdbcdecdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq" },
    { "" }
}
```

Definition at line 437 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.8 const int sha2_test_buflen[3] [static]**Initial value:**

```
{
    3, 56, 1000
}
```

Definition at line 444 of file sha2.c.

Referenced by sha2_self_test().

13.131.3.9 const unsigned char sha2_test_sum[6][32] [static]**Initial value:**

```
{
    { 0x23, 0x09, 0x7D, 0x22, 0x34, 0x05, 0xD8, 0x22,
      0x86, 0x42, 0xA4, 0x77, 0xBD, 0xA2, 0x55, 0xB3,
      0x2A, 0xAD, 0xBC, 0xE4, 0xBD, 0xA0, 0xB3, 0xF7,
      0xE3, 0x6C, 0x9D, 0xA7 },
    { 0x75, 0x38, 0x8B, 0x16, 0x51, 0x27, 0x76, 0xCC,
      0x5D, 0xBA, 0x5D, 0xA1, 0xFD, 0x89, 0x01, 0x50,
      0xB0, 0xC6, 0x45, 0x5C, 0xB4, 0xF5, 0x8B, 0x19,
      0x52, 0x52, 0x25, 0x25 }
}
```

```
{ 0x20, 0x79, 0x46, 0x55, 0x98, 0x0C, 0x91, 0xD8,
  0xBB, 0xB4, 0xC1, 0xEA, 0x97, 0x61, 0x8A, 0x4B,
  0xF0, 0x3F, 0x42, 0x58, 0x19, 0x48, 0xB2, 0xEE,
  0x4E, 0xE7, 0xAD, 0x67 },

{ 0xBA, 0x78, 0x16, 0xBF, 0x8F, 0x01, 0xCF, 0xEA,
  0x41, 0x41, 0x40, 0xDE, 0x5D, 0xAE, 0x22, 0x23,
  0xB0, 0x03, 0x61, 0xA3, 0x96, 0x17, 0x7A, 0x9C,
  0xB4, 0x10, 0xFF, 0x61, 0xF2, 0x00, 0x15, 0xAD },
{ 0x24, 0x8D, 0x6A, 0x61, 0xD2, 0x06, 0x38, 0xB8,
  0xE5, 0xC0, 0x26, 0x93, 0x0C, 0x3E, 0x60, 0x39,
  0xA3, 0x3C, 0xE4, 0x59, 0x64, 0xFF, 0x21, 0x67,
  0xF6, 0xEC, 0xED, 0xD4, 0x19, 0xDB, 0x06, 0xC1 },
{ 0xCD, 0xC7, 0x6E, 0x5C, 0x99, 0x14, 0xFB, 0x92,
  0x81, 0xA1, 0xC7, 0xE2, 0x84, 0xD7, 0x3E, 0x67,
  0xF1, 0x80, 0x9A, 0x48, 0xA4, 0x97, 0x20, 0x0E,
  0x04, 0x6D, 0x39, 0xCC, 0xC7, 0x11, 0x2C, 0xD0 }
}
```

Definition at line 449 of file sha2.c.

Referenced by sha2_self_test().

13.132 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/sha4.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/sha4.h"
#include <string.h>
#include <stdio.h>
```

Defines

- #define [GET_UINT64_BE](#)(n, b, i)
- #define [PUT_UINT64_BE](#)(n, b, i)
- #define [SHR](#)(x, n) (x >> n)
- #define [ROTR](#)(x, n) (SHR(x, n) | (x << (64 - n)))
- #define [S0](#)(x) (ROTR(x, 1) ^ ROTR(x, 8) ^ SHR(x, 7))
- #define [S1](#)(x) (ROTR(x, 19) ^ ROTR(x, 61) ^ SHR(x, 6))
- #define [S2](#)(x) (ROTR(x, 28) ^ ROTR(x, 34) ^ ROTR(x, 39))
- #define [S3](#)(x) (ROTR(x, 14) ^ ROTR(x, 18) ^ ROTR(x, 41))
- #define [F0](#)(x, y, z) ((x & y) | (z & (x | y)))
- #define [F1](#)(x, y, z) (z ^ (x & (y ^ z)))
- #define [P](#)(a, b, c, d, e, f, g, h, x, [K](#))

Functions

- void [sha4_starts](#) ([sha4_context](#) *ctx, [int](#) is384)
SHA-512 context setup.
- static void [sha4_process](#) ([sha4_context](#) *ctx, unsigned char data[128])
- void [sha4_update](#) ([sha4_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-512 process buffer.
- void [sha4_finish](#) ([sha4_context](#) *ctx, unsigned char output[64])
SHA-512 final digest.
- void [sha4](#) (unsigned char *input, [int](#) ilen, unsigned char output[64], [int](#) is384)
Output = SHA-512(input buffer).
- [int](#) [sha4_file](#) (char *path, unsigned char output[64], [int](#) is384)
Output = SHA-512(file contents).
- void [sha4_hmac_starts](#) ([sha4_context](#) *ctx, unsigned char *key, [int](#) keylen, [int](#) is384)
SHA-512 HMAC context setup.
- void [sha4_hmac_update](#) ([sha4_context](#) *ctx, unsigned char *input, [int](#) ilen)
SHA-512 HMAC process buffer.
- void [sha4_hmac_finish](#) ([sha4_context](#) *ctx, unsigned char output[64])

SHA-512 HMAC final digest.

- void `sha4_hmac` (unsigned char *key, int keylen, unsigned char *input, int ilen, unsigned char output[64], int is384)

Output = HMAC-SHA-512(hmac key, input buffer).

- int `sha4_self_test` (int verbose)

Checkup routine.

Variables

- static const unsigned int64 `K` [80]
- static const unsigned char `sha4_padding` [128]
- static unsigned char `sha4_test_buf` [3][113]
- static const int `sha4_test_buflen` [3]
- static const unsigned char `sha4_test_sum` [6][64]
- static unsigned char `sha4_hmac_test_key` [7][26]
- static const int `sha4_hmac_test_keylen` [7]
- static unsigned char `sha4_hmac_test_buf` [7][153]
- static const int `sha4_hmac_test_buflen` [7]
- static const unsigned char `sha4_hmac_test_sum` [14][64]

13.132.1 Define Documentation

13.132.1.1 `#define F0(x, y, z) ((x & y) | (z & (x | y)))`

13.132.1.2 `#define F1(x, y, z) (z ^ (x & (y ^ z)))`

13.132.1.3 `#define GET_UINT64_BE(n, b, i)`

Value:

```
{
    (n) = ( (unsigned int64) (b)[(i)      ] << 56 ) \
    | ( (unsigned int64) (b)[(i) + 1] << 48 ) \
    | ( (unsigned int64) (b)[(i) + 2] << 40 ) \
    | ( (unsigned int64) (b)[(i) + 3] << 32 ) \
    | ( (unsigned int64) (b)[(i) + 4] << 24 ) \
    | ( (unsigned int64) (b)[(i) + 5] << 16 ) \
    | ( (unsigned int64) (b)[(i) + 6] <<  8 ) \
    | ( (unsigned int64) (b)[(i) + 7]      ); \
}
```

Definition at line 39 of file sha4.c.

Referenced by `sha4_process()`.

13.132.1.4 `#define P(a, b, c, d, e, f, g, h, x, K)`

Value:

```

{
    temp1 = h + S3(e) + F1(e, f, g) + K + x;
    temp2 = S2(a) + F0(a, b, c);
    d += temp1; h = temp1 + temp2;
}

```

13.132.1.5 #define PUT_UINT64_BE(n, b, i)

Value:

```

{
    (b)[(i)    ] = (unsigned char) ( (n) >> 56 );
    (b)[(i) + 1] = (unsigned char) ( (n) >> 48 );
    (b)[(i) + 2] = (unsigned char) ( (n) >> 40 );
    (b)[(i) + 3] = (unsigned char) ( (n) >> 32 );
    (b)[(i) + 4] = (unsigned char) ( (n) >> 24 );
    (b)[(i) + 5] = (unsigned char) ( (n) >> 16 );
    (b)[(i) + 6] = (unsigned char) ( (n) >> 8 );
    (b)[(i) + 7] = (unsigned char) ( (n) );
}

```

Definition at line 53 of file sha4.c.

Referenced by sha4_finish().

13.132.1.6 #define ROTR(x, n) (SHR(x,n) | (x << (64 - n)))

13.132.1.7 #define S0(x) (ROTR(x, 1) ^ ROTR(x, 8) ^ SHR(x, 7))

13.132.1.8 #define S1(x) (ROTR(x,19) ^ ROTR(x,61) ^ SHR(x, 6))

13.132.1.9 #define S2(x) (ROTR(x,28) ^ ROTR(x,34) ^ ROTR(x,39))

13.132.1.10 #define S3(x) (ROTR(x,14) ^ ROTR(x,18) ^ ROTR(x,41))

13.132.1.11 #define SHR(x, n) (x >> n)

13.132.2 Function Documentation

13.132.2.1 void sha4 (unsigned char *input, int ilen, unsigned char output[64], int is384)

Output = SHA-512(input buffer).

Parameters:

input buffer holding the data

ilen length of the input data

output SHA-384/512 checksum result

is384 0 = use SHA512, 1 = use SHA384

Definition at line 312 of file sha4.c.

References sha4_finish(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac_starts().

13.132.2.2 int sha4_file (char * *path*, unsigned char *output*[64], int *is384*)

Output = SHA-512(file contents).

Parameters:

path input file name

output SHA-384/512 checksum result

is384 0 = use SHA512, 1 = use SHA384

Returns:

0 if successful, 1 if fopen failed, or 2 if fread failed

Definition at line 327 of file sha4.c.

References buf, f, sha4_finish(), sha4_starts(), and sha4_update().

13.132.2.3 void sha4_finish (sha4_context * *ctx*, unsigned char *output*[64])

SHA-512 final digest.

Parameters:

ctx SHA-512 context

output SHA-384/512 checksum result

Definition at line 276 of file sha4.c.

References int, int64, sha4_context::is384, PUT_UINT64_BE, sha4_padding, sha4_update(), sha4_context::state, and sha4_context::total.

Referenced by sha4(), sha4_file(), sha4_hmac_finish(), and sha4_self_test().

13.132.2.4 void sha4_hmac (unsigned char * *key*, int *keylen*, unsigned char * *input*, int *ilen*, unsigned char *output*[64], int *is384*)

Output = HMAC-SHA-512(hmac key, input buffer).

Parameters:

key HMAC secret key

keylen length of the HMAC key

input buffer holding the data

ilen length of the input data

output HMAC-SHA-384/512 result

is384 0 = use SHA512, 1 = use SHA384

Definition at line 419 of file sha4.c.

References sha4_hmac_finish(), sha4_hmac_starts(), and sha4_hmac_update().

13.132.2.5 void sha4_hmac_finish (sha4_context * *ctx*, unsigned char *output*[64])

SHA-512 HMAC final digest.

Parameters:

ctx HMAC context

output SHA-384/512 HMAC checksum result

Definition at line 399 of file sha4.c.

References sha4_context::is384, sha4_context::opad, sha4_finish(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac(), and sha4_self_test().

13.132.2.6 void sha4_hmac_starts (sha4_context * *ctx*, unsigned char * *key*, int *keylen*, int *is384*)

SHA-512 HMAC context setup.

Parameters:

ctx HMAC context to be initialized

is384 0 = use SHA512, 1 = use SHA384

key HMAC secret key

keylen length of the HMAC key

Definition at line 359 of file sha4.c.

References sha4_context::ipad, sha4_context::opad, sha4(), sha4_starts(), and sha4_update().

Referenced by sha4_hmac(), and sha4_self_test().

13.132.2.7 void sha4_hmac_update (sha4_context * *ctx*, unsigned char * *input*, int *ilen*)

SHA-512 HMAC process buffer.

Parameters:

ctx HMAC context

input buffer holding the data

ilen length of the input data

Definition at line 390 of file sha4.c.

References sha4_update().

Referenced by sha4_hmac(), and sha4_self_test().

13.132.2.8 static void sha4_process (sha4_context * *ctx*, unsigned char *data*[128]) [static]

Definition at line 149 of file sha4.c.

References F, GET_UINT64_BE, int64, K, P, S0, S1, and sha4_context::state.

Referenced by sha4_update().

13.132.2.9 `int sha4_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 654 of file sha4.c.

References `buf`, `sha4_finish()`, `sha4_hmac_finish()`, `sha4_hmac_starts()`, `sha4_hmac_test_buf`, `sha4_hmac_test_buflen`, `sha4_hmac_test_key`, `sha4_hmac_test_keylen`, `sha4_hmac_test_sum`, `sha4_hmac_update()`, `sha4_starts()`, `sha4_test_buf`, `sha4_test_buflen`, `sha4_test_sum`, and `sha4_update()`.

Referenced by `main()`.

13.132.2.10 `void sha4_starts (sha4_context * ctx, int is384)`

SHA-512 context setup.

Parameters:

ctx context to be initialized

is384 0 = use SHA512, 1 = use SHA384

Definition at line 116 of file sha4.c.

References `sha4_context::is384`, `sha4_context::state`, `sha4_context::total`, and `UL64`.

Referenced by `sha4()`, `sha4_file()`, `sha4_hmac_finish()`, `sha4_hmac_starts()`, and `sha4_self_test()`.

13.132.2.11 `void sha4_update (sha4_context * ctx, unsigned char * input, int ilen)`

SHA-512 process buffer.

Parameters:

ctx SHA-512 context

input buffer holding the data

ilen length of the input data

Definition at line 221 of file sha4.c.

References `sha4_context::buffer`, `int`, `int64`, `sha4_process()`, and `sha4_context::total`.

Referenced by `sha4()`, `sha4_file()`, `sha4_finish()`, `sha4_hmac_finish()`, `sha4_hmac_starts()`, `sha4_hmac_update()`, and `sha4_self_test()`.

13.132.3 Variable Documentation

13.132.3.1 `const unsigned int64 K[80] [static]`

Definition at line 69 of file sha4.c.

Definition at line 506 of file sha4.c.

Referenced by sha4_self_test().

13.132.3.5 `const int sha4_hmac_test_keylen[7] [static]`

Initial value:

```
{
    20, 4, 20, 25, 20, 131, 131
}
```

Definition at line 521 of file sha4.c.

Referenced by sha4_self_test().

13.132.3.6 `const unsigned char sha4_hmac_test_sum[14][64] [static]`

Definition at line 552 of file sha4.c.

Referenced by sha4_self_test().

13.132.3.7 `const unsigned char sha4_padding[128] [static]`

Initial value:

```
{
    0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
}
```

Definition at line 261 of file sha4.c.

Referenced by sha4_finish().

13.132.3.8 `unsigned char sha4_test_buf[3][113] [static]`

Initial value:

```
{
    { "abc" },
    { "abcdefghbcdefghicdefghijdefghijkefghijklfghijklmghijklmn"
      "hijklmnoijklmnopjklmnopqklmnopqrlmnopqrsmnopqrstnopqrstu" },
    { "" }
}
```

Definition at line 437 of file sha4.c.

Referenced by sha4_self_test().

13.132.3.9 `const int sha4_test_buflen[3] [static]`

Initial value:

```
{  
    3, 112, 1000  
}
```

Definition at line 445 of file sha4.c.

Referenced by sha4_self_test().

13.132.3.10 `const unsigned char sha4_test_sum[6][64] [static]`

Definition at line 450 of file sha4.c.

Referenced by sha4_self_test().

13.133 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/ssl_cli.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/debug.h"
#include "xyssl/ssl.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
```

Functions

- static [int ssl_write_client_hello](#) (ssl_context *ssl)
- static [int ssl_parse_server_hello](#) (ssl_context *ssl)
- static [int ssl_parse_server_key_exchange](#) (ssl_context *ssl)
- static [int ssl_parse_certificate_request](#) (ssl_context *ssl)
- static [int ssl_parse_server_hello_done](#) (ssl_context *ssl)
- static [int ssl_write_client_key_exchange](#) (ssl_context *ssl)
- static [int ssl_write_certificate_verify](#) (ssl_context *ssl)
- [int ssl_handshake_client](#) (ssl_context *ssl)

13.133.1 Function Documentation

13.133.1.1 [int ssl_handshake_client](#) (ssl_context *ssl)

Definition at line 660 of file `ssl_cli.c`.

References `SSL_CERTIFICATE_REQUEST`, `SSL_CERTIFICATE_VERIFY`, `SSL_CLIENT_CERTIFICATE`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_CLIENT_FINISHED`, `SSL_CLIENT_HELLO`, `SSL_CLIENT_KEY_EXCHANGE`, `SSL_DEBUG_MSG`, `SSL_FLUSH_BUFFERS`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_HELLO_REQUEST`, `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, `SSL_SERVER_CERTIFICATE`, `SSL_SERVER_CHANGE_CIPHER_SPEC`, `SSL_SERVER_FINISHED`, `SSL_SERVER_HELLO`, `SSL_SERVER_HELLO_DONE`, `SSL_SERVER_KEY_EXCHANGE`, `ssl_write_certificate()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `ssl_handshake()`.

13.133.1.2 [static int ssl_parse_certificate_request](#) (ssl_context *ssl) **[static]**

Definition at line 428 of file `ssl_cli.c`.

References `_ssl_context::client_auth`, `_ssl_context::in_msg`, `_ssl_context::in_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE_REQUEST`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`.

13.133.1.3 static int ssl_parse_server_hello (ssl_context * ssl) [static]

Definition at line 168 of file ssl_cli.c.

References buf, _ssl_session::cipher, _ssl_context::ciphers, _ssl_session::id, _ssl_context::in_hslen, _ssl_context::in_msg, _ssl_context::in_msgtype, _ssl_session::length, _ssl_context::minor_ver, _ssl_context::randbytes, _ssl_context::resume, _ssl_context::session, SSL_COMPRESS_NULL, SSL_DEBUG_BUF, SSL_DEBUG_MSG, SSL_DEBUG_RET, ssl_derive_keys(), SSL_HS_SERVER_HELLO, SSL_MAJOR_VERSION_3, SSL_MINOR_VERSION_0, SSL_MINOR_VERSION_1, SSL_MSG_HANDSHAKE, ssl_read_record(), SSL_SERVER_CHANGE_CIPHER_SPEC, _ssl_session::start, _ssl_context::state, XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO, and XYSSL_ERR_SSL_UNEXPECTED_MESSAGE.

Referenced by ssl_handshake_client().

13.133.1.4 static int ssl_parse_server_hello_done (ssl_context * ssl) [static]

Definition at line 471 of file ssl_cli.c.

References _ssl_context::client_auth, _ssl_context::in_hslen, _ssl_context::in_msg, _ssl_context::in_msgtype, SSL_DEBUG_MSG, SSL_DEBUG_RET, SSL_HS_SERVER_HELLO_DONE, SSL_MSG_HANDSHAKE, ssl_read_record(), _ssl_context::state, XYSSL_ERR_SSL_BAD_HS_SERVER_HELLO_DONE, and XYSSL_ERR_SSL_UNEXPECTED_MESSAGE.

Referenced by ssl_handshake_client().

13.133.1.5 static int ssl_parse_server_key_exchange (ssl_context * ssl) [static]

Definition at line 312 of file ssl_cli.c.

References _ssl_session::cipher, _ssl_context::dhm_ctx, dhm_read_params(), dhm_context::G, dhm_context::GY, _ssl_context::in_hslen, _ssl_context::in_msg, _ssl_context::in_msgtype, dhm_context::len, rsa_context::len, md5(), md5_finish(), md5_starts(), md5_update(), dhm_context::P, _ssl_context::peer_cert, _ssl_context::randbytes, x509_cert::rsa, rsa_pkcs1_verify(), RSA_PUBLIC, RSA_RAW, _ssl_context::session, sha1(), sha1_finish(), sha1_starts(), sha1_update(), SSL_DEBUG_BUF, SSL_DEBUG_MPI, SSL_DEBUG_MSG, SSL_DEBUG_RET, SSL_EDH_RSA_AES_256_SHA, SSL_EDH_RSA_DES_168_SHA, SSL_HS_SERVER_KEY_EXCHANGE, SSL_MSG_HANDSHAKE, ssl_read_record(), _ssl_context::state, XYSSL_ERR_SSL_BAD_HS_SERVER_KEY_EXCHANGE, XYSSL_ERR_SSL_FEATURE_UNAVAILABLE, and XYSSL_ERR_SSL_UNEXPECTED_MESSAGE.

Referenced by ssl_handshake_client().

13.133.1.6 static int ssl_write_certificate_verify (ssl_context * ssl) [static]

Definition at line 604 of file ssl_cli.c.

References _ssl_context::client_auth, rsa_context::len, _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, _ssl_context::rsa_key, rsa_pkcs1_sign(), RSA_PRIVATE, RSA_RAW, ssl_calc_verify(), SSL_DEBUG_MSG, SSL_DEBUG_RET, SSL_HS_CERTIFICATE_VERIFY, SSL_MSG_HANDSHAKE, ssl_write_record(), _ssl_context::state, and XYSSL_ERR_SSL_PRIVATE_KEY_REQUIRED.

Referenced by ssl_handshake_client().

13.133.1.7 static int ssl_write_client_hello (ssl_context *ssl) [static]

Definition at line 33 of file ssl_cli.c.

References buf, _ssl_context::ciphers, _ssl_context::f_rng, _ssl_context::hostname, _ssl_context::hostname_len, _ssl_session::id, _ssl_session::length, _ssl_context::major_ver, _ssl_context::max_major_ver, _ssl_context::max_minor_ver, _ssl_context::minor_ver, _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, _ssl_context::p_rng, _ssl_context::randbytes, _ssl_context::resume, _ssl_context::session, SSL_COMPRESS_NULL, SSL_DEBUG_BUF, SSL_DEBUG_MSG, SSL_DEBUG_RET, SSL_HS_CLIENT_HELLO, SSL_MAJOR_VERSION_3, SSL_MINOR_VERSION_0, SSL_MINOR_VERSION_1, SSL_MSG_HANDSHAKE, ssl_write_record(), _ssl_session::start, _ssl_context::state, _ssl_context::timeout, TLS_EXT_SERVERNAME, and TLS_EXT_SERVERNAME_HOSTNAME.

Referenced by ssl_handshake_client().

13.133.1.8 static int ssl_write_client_key_exchange (ssl_context *ssl) [static]

Definition at line 506 of file ssl_cli.c.

References _ssl_session::cipher, dhm_calc_secret(), _ssl_context::dhm_ctx, dhm_make_public(), _ssl_context::f_rng, dhm_context::GX, dhm_context::K, rsa_context::len, dhm_context::len, _ssl_context::max_major_ver, _ssl_context::max_minor_ver, _ssl_context::minor_ver, _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, _ssl_context::p_rng, _ssl_context::peer_cert, _ssl_context::pmslen, _ssl_context::premaster, x509_cert::rsa, rsa_pkcs1_encrypt(), RSA_PUBLIC, _ssl_context::session, SSL_DEBUG_MPI, SSL_DEBUG_MSG, SSL_DEBUG_RET, ssl_derive_keys(), SSL_EDH_RSA_AES_256_SHA, SSL_EDH_RSA_DES_168_SHA, SSL_HS_CLIENT_KEY_EXCHANGE, SSL_MINOR_VERSION_0, SSL_MSG_HANDSHAKE, ssl_write_record(), _ssl_context::state, dhm_context::X, and XYSSL_ERR_SSL_FEATURE_UNAVAILABLE.

Referenced by ssl_handshake_client().

13.134 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/ssl_srv.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/debug.h"
#include "xyssl/ssl.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
```

Functions

- static [int ssl_parse_client_hello](#) (ssl_context *ssl)
- static [int ssl_write_server_hello](#) (ssl_context *ssl)
- static [int ssl_write_certificate_request](#) (ssl_context *ssl)
- static [int ssl_write_server_key_exchange](#) (ssl_context *ssl)
- static [int ssl_write_server_hello_done](#) (ssl_context *ssl)
- static [int ssl_parse_client_key_exchange](#) (ssl_context *ssl)
- static [int ssl_parse_certificate_verify](#) (ssl_context *ssl)
- [int ssl_handshake_server](#) (ssl_context *ssl)

13.134.1 Function Documentation

13.134.1.1 [int ssl_handshake_server](#) (ssl_context * ssl)

Definition at line 819 of file ssl_srv.c.

References [SSL_CERTIFICATE_REQUEST](#), [SSL_CERTIFICATE_VERIFY](#), [SSL_CLIENT_CERTIFICATE](#), [SSL_CLIENT_CHANGE_CIPHER_SPEC](#), [SSL_CLIENT_FINISHED](#), [SSL_CLIENT_HELLO](#), [SSL_CLIENT_KEY_EXCHANGE](#), [SSL_DEBUG_MSG](#), [SSL_FLUSH_BUFFERS](#), [ssl_flush_output\(\)](#), [SSL_HANDSHAKE_OVER](#), [SSL_HELLO_REQUEST](#), [ssl_parse_certificate\(\)](#), [ssl_parse_certificate_verify\(\)](#), [ssl_parse_change_cipher_spec\(\)](#), [ssl_parse_client_hello\(\)](#), [ssl_parse_client_key_exchange\(\)](#), [ssl_parse_finished\(\)](#), [SSL_SERVER_CERTIFICATE](#), [SSL_SERVER_CHANGE_CIPHER_SPEC](#), [SSL_SERVER_FINISHED](#), [SSL_SERVER_HELLO](#), [SSL_SERVER_HELLO_DONE](#), [SSL_SERVER_KEY_EXCHANGE](#), [ssl_write_certificate\(\)](#), [ssl_write_certificate_request\(\)](#), [ssl_write_change_cipher_spec\(\)](#), [ssl_write_finished\(\)](#), [ssl_write_server_hello\(\)](#), [ssl_write_server_hello_done\(\)](#), [ssl_write_server_key_exchange\(\)](#), [_ssl_context::state](#), and [XYSSL_ERR_SSL_BAD_INPUT_DATA](#).

Referenced by [ssl_handshake\(\)](#).

13.134.1.2 [static int ssl_parse_certificate_verify](#) (ssl_context * ssl) **[static]**

Definition at line 758 of file ssl_srv.c.

References [_ssl_context::in_hrlen](#), [_ssl_context::in_msg](#), [_ssl_context::in_msgtype](#), [rsa_context::len](#), [_ssl_context::peer_cert](#), [_x509_cert::rsa](#), [rsa_pkcs1_verify\(\)](#), [RSA_PUBLIC](#), [RSA_RAW](#), [ssl_calc_verify\(\)](#), [SSL_DEBUG_MSG](#), [SSL_DEBUG_RET](#), [SSL_HS_CERTIFICATE_VERIFY](#), [SSL_](#)

MSG_HANDSHAKE, ssl_read_record(), _ssl_context::state, and XYSSL_ERR_SSL_BAD_HS_CERTIFICATE_VERIFY.

Referenced by ssl_handshake_server().

13.134.1.3 static int ssl_parse_client_hello (ssl_context * ssl) [static]

Definition at line 33 of file ssl_srv.c.

References buf, _ssl_session::cipher, _ssl_context::ciphers, _ssl_context::fin_md5, _ssl_context::fin_sha1, _ssl_session::id, _ssl_context::in_hdr, _ssl_context::in_left, _ssl_context::in_msg, _ssl_session::length, _ssl_context::major_ver, _ssl_context::max_major_ver, _ssl_context::max_minor_ver, md5_update(), _ssl_context::minor_ver, _ssl_context::randbytes, _ssl_context::session, sha1_update(), SSL_DEBUG_BUF, SSL_DEBUG_MSG, SSL_DEBUG_RET, ssl_fetch_input(), SSL_HS_CLIENT_HELLO, SSL_MAJOR_VERSION_3, SSL_MINOR_VERSION_1, SSL_MSG_HANDSHAKE, _ssl_context::state, XYSSL_ERR_SSL_BAD_HS_CLIENT_HELLO, and XYSSL_ERR_SSL_NO_CIPHER_CHOSEN.

Referenced by ssl_handshake_server().

13.134.1.4 static int ssl_parse_client_key_exchange (ssl_context * ssl) [static]

Definition at line 634 of file ssl_srv.c.

References _ssl_session::cipher, dhm_calc_secret(), _ssl_context::dhm_ctx, dhm_read_public(), _ssl_context::f_rng, dhm_context::GY, _ssl_context::in_hrlen, _ssl_context::in_msg, _ssl_context::in_msgtype, dhm_context::K, rsa_context::len, dhm_context::len, _ssl_context::max_major_ver, _ssl_context::max_minor_ver, _ssl_context::minor_ver, _ssl_context::p_rng, _ssl_context::pmslen, _ssl_context::premaster, _ssl_context::rsa_key, rsa_pkcs1_decrypt(), RSA_PRIVATE, _ssl_context::s_set, _ssl_context::session, SSL_DEBUG_MPI, SSL_DEBUG_MSG, SSL_DEBUG_RET, ssl_derive_keys(), SSL_EDH_RSA_AES_256_SHA, SSL_EDH_RSA_DES_168_SHA, SSL_HS_CLIENT_KEY_EXCHANGE, SSL_MINOR_VERSION_0, SSL_MSG_HANDSHAKE, ssl_read_record(), _ssl_context::state, XYSSL_ERR_SSL_BAD_HS_CLIENT_KEY_EXCHANGE, and XYSSL_ERR_SSL_FEATURE_UNAVAILABLE.

Referenced by ssl_handshake_server().

13.134.1.5 static int ssl_write_certificate_request (ssl_context * ssl) [static]

Definition at line 447 of file ssl_srv.c.

References _ssl_context::authmode, buf, _ssl_context::ca_chain, _x509_buf::len, _x509_cert::next, _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, _x509_buf::p, SSL_DEBUG_BUF, SSL_DEBUG_MSG, SSL_HS_CERTIFICATE_REQUEST, SSL_MSG_HANDSHAKE, SSL_VERIFY_NONE, ssl_write_record(), _ssl_context::state, and _x509_cert::subject_raw.

Referenced by ssl_handshake_server().

13.134.1.6 static int ssl_write_server_hello (ssl_context * ssl) [static]

Definition at line 349 of file ssl_srv.c.

References buf, _ssl_session::cipher, _ssl_context::f_rng, _ssl_session::id, _ssl_session::length, _ssl_context::major_ver, _ssl_context::minor_ver, _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, _ssl_context::p_rng, _ssl_context::randbytes, _ssl_context::resume, _ssl_context::s_get, _ssl_context::session, SSL_COMPRESS_NULL, SSL_DEBUG_BUF, SSL_DEBUG_

MSG, ssl_derive_keys(), SSL_HS_SERVER_HELLO, SSL_MSG_HANDSHAKE, SSL_SERVER_CHANGE_CIPHER_SPEC, ssl_write_record(), and _ssl_context::state.

Referenced by ssl_handshake_server().

13.134.1.7 static int ssl_write_server_hello_done(ssl_context *ssl) [static]

Definition at line 611 of file ssl_srv.c.

References _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, SSL_DEBUG_MSG, SSL_DEBUG_RET, SSL_HS_SERVER_HELLO_DONE, SSL_MSG_HANDSHAKE, ssl_write_record(), and _ssl_context::state.

Referenced by ssl_handshake_server().

13.134.1.8 static int ssl_write_server_key_exchange(ssl_context *ssl) [static]

Definition at line 512 of file ssl_srv.c.

References _ssl_session::cipher, _ssl_context::dhm_ctx, dhm_make_params(), _ssl_context::f_rng, dhm_context::G, dhm_context::GX, rsa_context::len, md5(), md5_finish(), md5_starts(), md5_update(), _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, dhm_context::P, _ssl_context::p_rng, _ssl_context::randbytes, _ssl_context::rsa_key, rsa_pkcs1_sign(), RSA_PRIVATE, RSA_RAW, _ssl_context::session, sha1(), sha1_finish(), sha1_starts(), sha1_update(), SSL_DEBUG_BUF, SSL_DEBUG_MPI, SSL_DEBUG_MSG, SSL_DEBUG_RET, SSL_EDH_RSA_AES_256_SHA, SSL_EDH_RSA_DES_168_SHA, SSL_HS_SERVER_KEY_EXCHANGE, SSL_MSG_HANDSHAKE, ssl_write_record(), _ssl_context::state, dhm_context::X, and XYSSL_ERR_SSL_FEATURE_UNAVAILABLE.

Referenced by ssl_handshake_server().

13.135 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/ssl_tls.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/aes.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
#include "xyssl/debug.h"
#include "xyssl/ssl.h"
#include <string.h>
#include <stdlib.h>
#include <time.h>
```

Functions

- static [int](#) [tls1_prf](#) (unsigned char *secret, [int](#) slen, char *label, unsigned char *random, [int](#) rlen, unsigned char *dstbuf, [int](#) dlen)
- [int](#) [ssl_derive_keys](#) ([ssl_context](#) *ssl)
- void [ssl_calc_verify](#) ([ssl_context](#) *ssl, unsigned char hash[36])
- static void [ssl_mac_md5](#) (unsigned char *secret, unsigned char *buf, [int](#) len, unsigned char *ctr, [int](#) type)
- static void [ssl_mac_sha1](#) (unsigned char *secret, unsigned char *buf, [int](#) len, unsigned char *ctr, [int](#) type)
- static [int](#) [ssl_encrypt_buf](#) ([ssl_context](#) *ssl)
- static [int](#) [ssl_decrypt_buf](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_fetch_input](#) ([ssl_context](#) *ssl, [int](#) nb_want)
- [int](#) [ssl_flush_output](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_write_record](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_read_record](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_write_certificate](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_parse_certificate](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_write_change_cipher_spec](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_parse_change_cipher_spec](#) ([ssl_context](#) *ssl)
- static void [ssl_calc_finished](#) ([ssl_context](#) *ssl, unsigned char *buf, [int](#) from, [md5_context](#) *md5, [sha1_context](#) *sha1)
- [int](#) [ssl_write_finished](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_parse_finished](#) ([ssl_context](#) *ssl)
- [int](#) [ssl_init](#) ([ssl_context](#) *ssl)

Initialize an SSL context.

- void [ssl_set_endpoint](#) ([ssl_context](#) *ssl, [int](#) endpoint)

Set the current endpoint type.

- void [ssl_set_authmode](#) ([ssl_context](#) *ssl, [int](#) authmode)

Set the certificate verification mode.

- void `ssl_set_rng` (`ssl_context` *ssl, `int`(*f_rng)(void *), void *p_rng)
Set the random number generator callback.
- void `ssl_set_dbg` (`ssl_context` *ssl, void(*f_dbg)(void *, `int`, char *), void *p_dbg)
Set the debug callback.
- void `ssl_set_bio` (`ssl_context` *ssl, `int`(*f_recv)(void *, unsigned char *, `int`), void *p_recv, `int`(*f_send)(void *, unsigned char *, `int`), void *p_send)
Set the underlying BIO read and write callbacks.
- void `ssl_set_scb` (`ssl_context` *ssl, `int`(*s_get)(`ssl_context` *), `int`(*s_set)(`ssl_context` *))
Set the session callbacks (server-side only).
- void `ssl_set_session` (`ssl_context` *ssl, `int` resume, `int` timeout, `ssl_session` *session)
Set the session resuming flag, timeout and data.
- void `ssl_set_ciphers` (`ssl_context` *ssl, `int` *ciphers)
Set the list of allowed ciphersuites.
- void `ssl_set_ca_chain` (`ssl_context` *ssl, `x509_cert` *ca_chain, char *peer_cn)
Set the data required to verify peer certificate.
- void `ssl_set_own_cert` (`ssl_context` *ssl, `x509_cert` *own_cert, `rsa_context` *rsa_key)
Set own certificate and private key.
- `int` `ssl_set_dh_param` (`ssl_context` *ssl, char *dhm_P, char *dhm_G)
Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).
- `int` `ssl_set_hostname` (`ssl_context` *ssl, char *hostname)
Set hostname for ServerName TLS Extension.
- `int` `ssl_get_bytes_avail` (`ssl_context` *ssl)
Return the number of data bytes available to read.
- `int` `ssl_get_verify_result` (`ssl_context` *ssl)
Return the result of the certificate verification.
- char * `ssl_get_cipher` (`ssl_context` *ssl)
Return the name of the current cipher.
- `int` `ssl_handshake` (`ssl_context` *ssl)
Perform the SSL handshake.
- `int` `ssl_read` (`ssl_context` *ssl, unsigned char *buf, `int` len)
Read at most 'len' application data bytes.
- `int` `ssl_write` (`ssl_context` *ssl, unsigned char *buf, `int` len)
Write exactly 'len' application data bytes.
- `int` `ssl_close_notify` (`ssl_context` *ssl)

Notify the peer that the connection is being closed.

- void `ssl_free` (`ssl_context` *`ssl`)

Free an SSL context.

Variables

- int `ssl_default_ciphers` []

13.135.1 Function Documentation

13.135.1.1 static void `ssl_calc_finished` (`ssl_context` *`ssl`, unsigned char *`buf`, int `from`, `md5_context` *`md5`, `sha1_context` *`sha1`) [static]

Definition at line 1344 of file `ssl_tls.c`.

References `_ssl_session::master`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::session`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `sha1_context::state`, `md5_context::state`, and `tls1_prf()`.

Referenced by `ssl_parse_finished()`, and `ssl_write_finished()`.

13.135.1.2 void `ssl_calc_verify` (`ssl_context` *`ssl`, unsigned char `hash`[36])

Definition at line 335 of file `ssl_tls.c`.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_session::master`, `md5()`, `md5_finish()`, `md5_starts()`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::session`, `sha1()`, `sha1_finish()`, `sha1_starts()`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, and `SSL_MINOR_VERSION_0`.

Referenced by `ssl_parse_certificate_verify()`, and `ssl_write_certificate_verify()`.

13.135.1.3 int `ssl_close_notify` (`ssl_context` *`ssl`)

Notify the peer that the connection is being closed.

Definition at line 1904 of file `ssl_tls.c`.

References `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_WARNING`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_ALERT`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.4 static int `ssl_decrypt_buf` (`ssl_context` *`ssl`) [static]

Definition at line 555 of file `ssl_tls.c`.

References `aes_crypt_cbc()`, `AES_DECRYPT`, `arc4_crypt()`, `_ssl_context::ctx_dec`, `des3_crypt_cbc()`, `DES_DECRYPT`, `_ssl_context::in_ctr`, `_ssl_context::in_hdr`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::iv_dec`, `_ssl_context::ivlen`, `_ssl_context::mac_dec`, `_ssl_context::maclen`, `md5_hmac()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `_ssl_context::nb_`

zero, sha1_hmac(), SSL_DEBUG_BUF, SSL_DEBUG_MSG, ssl_mac_md5(), ssl_mac_sha1(), SSL_MINOR_VERSION_0, XYSSL_ERR_SSL_FEATURE_UNAVAILABLE, and XYSSL_ERR_SSL_INVALID_MAC.

Referenced by ssl_read_record().

13.135.1.5 int ssl_derive_keys (ssl_context * ssl)

Definition at line 106 of file ssl_tls.c.

References aes_setkey_dec(), aes_setkey_enc(), arc4_setup(), _ssl_session::cipher, _ssl_context::ctx_dec, _ssl_context::ctx_enc, des3_set3key_dec(), des3_set3key_enc(), _ssl_context::endpoint, _ssl_context::iv_dec, _ssl_context::iv_enc, _ssl_context::ivlen, _ssl_context::keylen, _ssl_context::mac_dec, _ssl_context::mac_enc, _ssl_context::maclen, _ssl_session::master, md5(), md5_finish(), md5_starts(), md5_update(), _ssl_context::minlen, _ssl_context::minor_ver, _ssl_context::pmslen, _ssl_context::premaster, _ssl_context::randbytes, _ssl_context::resume, _ssl_context::session, sha1(), sha1_finish(), sha1_starts(), sha1_update(), SSL_DEBUG_BUF, SSL_DEBUG_MSG, SSL_EDH_RSA_AES_256_SHA, SSL_EDH_RSA_DES_168_SHA, ssl_get_cipher(), SSL_IS_CLIENT, SSL_MINOR_VERSION_0, SSL_RSA_AES_128_SHA, SSL_RSA_AES_256_SHA, SSL_RSA_DES_168_SHA, SSL_RSA_RC4_128_MD5, SSL_RSA_RC4_128_SHA, tls1_prf(), and XYSSL_ERR_SSL_FEATURE_UNAVAILABLE.

Referenced by ssl_parse_client_key_exchange(), ssl_parse_server_hello(), ssl_write_client_key_exchange(), and ssl_write_server_hello().

13.135.1.6 static int ssl_encrypt_buf (ssl_context * ssl) [static]

Definition at line 448 of file ssl_tls.c.

References aes_crypt_cbc(), AES_ENCRYPT, arc4_crypt(), _ssl_context::ctx_enc, des3_crypt_cbc(), DES_ENCRYPT, _ssl_context::iv_enc, _ssl_context::ivlen, _ssl_context::mac_enc, _ssl_context::maclen, md5_hmac(), _ssl_context::minor_ver, _ssl_context::out_ctr, _ssl_context::out_msg, _ssl_context::out_msglen, _ssl_context::out_msgtype, sha1_hmac(), SSL_DEBUG_BUF, SSL_DEBUG_MSG, ssl_mac_md5(), ssl_mac_sha1(), SSL_MINOR_VERSION_0, and XYSSL_ERR_SSL_FEATURE_UNAVAILABLE.

Referenced by ssl_write_record().

13.135.1.7 int ssl_fetch_input (ssl_context * ssl, int nb_want)

Definition at line 727 of file ssl_tls.c.

References _ssl_context::f_recv, _ssl_context::in_hdr, _ssl_context::in_left, _ssl_context::p_recv, SSL_DEBUG_MSG, and SSL_DEBUG_RET.

Referenced by ssl_parse_client_hello(), and ssl_read_record().

13.135.1.8 int ssl_flush_output (ssl_context * ssl)

Definition at line 756 of file ssl_tls.c.

References buf, _ssl_context::f_send, _ssl_context::out_hdr, _ssl_context::out_left, _ssl_context::out_msglen, _ssl_context::p_send, SSL_DEBUG_MSG, and SSL_DEBUG_RET.

Referenced by ssl_close_notify(), ssl_handshake_client(), ssl_handshake_server(), ssl_write(), and ssl_write_record().

13.135.1.9 void ssl_free (ssl_context * ssl)

Free an SSL context.

Definition at line 1938 of file ssl_tls.c.

References `_ssl_context::dhm_ctx`, `dhm_free()`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::out_ctr`, `_ssl_context::peer_cert`, `SSL_BUFFER_LEN`, `SSL_DEBUG_MSG`, and `x509_free()`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.10 int ssl_get_bytes_avail (ssl_context * ssl)

Return the number of data bytes available to read.

Parameters:

ssl SSL context

Returns:

how many bytes are available in the read buffer

Definition at line 1691 of file ssl_tls.c.

References `_ssl_context::in_msglen`, and `_ssl_context::in_offt`.

13.135.1.11 char* ssl_get_cipher (ssl_context * ssl)

Return the name of the current cipher.

Parameters:

ssl SSL context

Returns:

a string containing the cipher name

Definition at line 1701 of file ssl_tls.c.

References `_ssl_session::cipher`, `_ssl_context::session`, `SSL_EDH_RSA_AES_256_SHA`, `SSL_EDH_RSA_DES_168_SHA`, `SSL_RSA_AES_128_SHA`, `SSL_RSA_AES_256_SHA`, `SSL_RSA_DES_168_SHA`, `SSL_RSA_RC4_128_MD5`, and `SSL_RSA_RC4_128_SHA`.

Referenced by `main()`, and `ssl_derive_keys()`.

13.135.1.12 int ssl_get_verify_result (ssl_context * ssl)

Return the result of the certificate verification.

Parameters:

ssl SSL context

Returns:

0 if successful, or a combination of: BADCERT_EXPIRED BADCERT_REVOKED BADCERT_CN_MISMATCH BADCERT_NOT_TRUSTED

Definition at line 1696 of file ssl_tls.c.

References `_ssl_context::verify_result`.

Referenced by `main()`.

13.135.1.13 int ssl_handshake (ssl_context * ssl)

Perform the SSL handshake.

Parameters:

ssl SSL context

Returns:

0 if successful, XYSSL_ERR_NET_TRY_AGAIN, or a specific SSL error code.

Definition at line 1767 of file ssl_tls.c.

References `_ssl_context::endpoint`, `SSL_DEBUG_MSG`, `ssl_handshake_client()`, `ssl_handshake_server()`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, and `XYSSL_ERR_SSL_FEATURE_UNAVAILABLE`.

Referenced by `main()`, `ssl_read()`, and `ssl_write()`.

13.135.1.14 int ssl_init (ssl_context * ssl)

Initialize an SSL context.

Parameters:

ssl SSL context

Returns:

0 if successful, or 1 if memory allocation failed

Definition at line 1542 of file ssl_tls.c.

References `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::hostname`, `_ssl_context::hostname_len`, `_ssl_context::in_ctr`, `_ssl_context::in_hdr`, `_ssl_context::in_msg`, `md5_starts()`, `_ssl_context::out_ctr`, `_ssl_context::out_hdr`, `_ssl_context::out_msg`, `sha1_starts()`, `SSL_BUFFER_LEN`, and `SSL_DEBUG_MSG`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.15 static void ssl_mac_md5 (unsigned char * secret, unsigned char * buf, int len, unsigned char * ctr, int type) [static]

Definition at line 387 of file ssl_tls.c.

References `md5()`, `md5_finish()`, `md5_starts()`, and `md5_update()`.

Referenced by `ssl_decrypt_buf()`, and `ssl_encrypt_buf()`.

13.135.1.16 `static void ssl_mac_sha1 (unsigned char * secret, unsigned char * buf, int len, unsigned char * ctr, int type) [static]`

Definition at line 416 of file `ssl_tls.c`.

References `sha1()`, `sha1_finish()`, `sha1_starts()`, and `sha1_update()`.

Referenced by `ssl_decrypt_buf()`, and `ssl_encrypt_buf()`.

13.135.1.17 `int ssl_parse_certificate (ssl_context * ssl)`

Definition at line 1140 of file `ssl_tls.c`.

References `_ssl_context::authmode`, `_ssl_context::ca_chain`, `_ssl_context::endpoint`, `_ssl_context::in_hrlen`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `int`, `_ssl_context::minor_ver`, `_ssl_context::peer_cert`, `_ssl_context::peer_cn`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_SERVER`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `SSL_VERIFY_NONE`, `SSL_VERIFY_OPTIONAL`, `SSL_VERIFY_REQUIRED`, `_ssl_context::state`, `_ssl_context::verify_result`, `x509parse_cert()`, `x509parse_verify()`, `XYSSL_ERR_SSL_BAD_HS_CERTIFICATE`, `XYSSL_ERR_SSL_CA_CHAIN_REQUIRED`, `XYSSL_ERR_SSL_NO_CLIENT_CERTIFICATE`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.135.1.18 `int ssl_parse_change_cipher_spec (ssl_context * ssl)`

Definition at line 1311 of file `ssl_tls.c`.

References `_ssl_context::do_crypt`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_read_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_BAD_HS_CHANGE_CIPHER_SPEC`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.135.1.19 `int ssl_parse_finished (ssl_context * ssl)`

Definition at line 1480 of file `ssl_tls.c`.

References `buf`, `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hrlen`, `_ssl_context::in_msg`, `_ssl_context::in_msgtype`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_IS_SERVER`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_read_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_BAD_HS_FINISHED`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.135.1.20 `int ssl_read (ssl_context * ssl, unsigned char * buf, int len)`

Read at most 'len' application data bytes.

Parameters:

ssl SSL context
buf buffer that will hold the data
len how many bytes must be read

Returns:

This function returns the number of bytes read, or a negative error code.

Definition at line 1791 of file ssl_tls.c.

References `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::in_offt`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MSG_APPLICATION_DATA`, `ssl_read_record()`, `_ssl_context::state`, and `XYSSL_ERR_SSL_UNEXPECTED_MESSAGE`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.21 int ssl_read_record (ssl_context * ssl)

Definition at line 842 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::in_hdr`, `_ssl_context::in_hrlen`, `_ssl_context::in_left`, `_ssl_context::in_msg`, `_ssl_context::in_msglen`, `_ssl_context::in_msgtype`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minlen`, `_ssl_context::minor_ver`, `sha1_update()`, `SSL_ALERT_CLOSE_NOTIFY`, `SSL_ALERT_FATAL`, `SSL_ALERT_WARNING`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_decrypt_buf()`, `ssl_fetch_input()`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MINOR_VERSION_1`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `XYSSL_ERR_SSL_FATAL_ALERT_MESSAGE`, `XYSSL_ERR_SSL_INVALID_RECORD`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

Referenced by `ssl_parse_certificate()`, `ssl_parse_certificate_request()`, `ssl_parse_certificate_verify()`, `ssl_parse_change_cipher_spec()`, `ssl_parse_client_key_exchange()`, `ssl_parse_finished()`, `ssl_parse_server_hello()`, `ssl_parse_server_hello_done()`, `ssl_parse_server_key_exchange()`, and `ssl_read()`.

13.135.1.22 void ssl_set_authmode (ssl_context * ssl, int authmode)

Set the certificate verification mode.

Parameters:

ssl SSL context
mode can be:

`SSL_VERIFY_NONE`: peer certificate is not checked (default), this is insecure and SHOULD be avoided.

`SSL_VERIFY_OPTIONAL`: peer certificate is checked, however the handshake continues even if verification failed; [ssl_get_verify_result\(\)](#) can be called after the handshake is complete.

`SSL_VERIFY_REQUIRED`: peer *must* present a valid certificate, handshake is aborted if verification failed.

Definition at line 1589 of file ssl_tls.c.

References `_ssl_context::authmode`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.23 `void ssl_set_bio (ssl_context * ssl, int(*)(void *, unsigned char *, int) f_recv, void * p_recv, int(*)(void *, unsigned char *, int) f_send, void * p_send)`

Set the underlying BIO read and write callbacks.

Parameters:

ssl SSL context
f_recv read callback
p_recv read parameter
f_send write callback
p_send write parameter

Definition at line 1610 of file `ssl_tls.c`.

References `_ssl_context::f_recv`, `_ssl_context::f_send`, `_ssl_context::p_recv`, and `_ssl_context::p_send`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.24 `void ssl_set_ca_chain (ssl_context * ssl, x509_cert * ca_chain, char * peer_cn)`

Set the data required to verify peer certificate.

Parameters:

ssl SSL context
ca_chain trusted CA chain
peer_cn expected peer CommonName (or NULL)

Note:

TODO: add two more parameters: depth and `crl`

Definition at line 1641 of file `ssl_tls.c`.

References `_ssl_context::ca_chain`, and `_ssl_context::peer_cn`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.25 `void ssl_set_ciphers (ssl_context * ssl, int * ciphers)`

Set the list of allowed ciphersuites.

Parameters:

ssl SSL context
ciphers 0-terminated list of allowed ciphers

Definition at line 1636 of file `ssl_tls.c`.

References `_ssl_context::ciphers`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.26 void ssl_set_dbg (ssl_context * ssl, void(*) (void *, int, char *) f_dbg, void * p_dbg)

Set the debug callback.

Parameters:

ssl SSL context

f_dbg debug function

p_dbg debug parameter

Definition at line 1602 of file ssl_tls.c.

References _ssl_context::f_dbg, and _ssl_context::p_dbg.

Referenced by main(), and ssl_test().

13.135.1.27 int ssl_set_dh_param (ssl_context * ssl, char * dhm_P, char * dhm_G)

Set the Diffie-Hellman public P and G values, read as hexadecimal strings (server-side only).

Parameters:

ssl SSL context

dhm_P Diffie-Hellman-Merkle modulus

dhm_G Diffie-Hellman-Merkle generator

Returns:

0 if successful

Definition at line 1655 of file ssl_tls.c.

References _ssl_context::dhm_ctx, dhm_context::G, mpi_read_string(), dhm_context::P, and SSL_DEBUG_RET.

Referenced by main(), and ssl_test().

13.135.1.28 void ssl_set_endpoint (ssl_context * ssl, int endpoint)

Set the current endpoint type.

Parameters:

ssl SSL context

endpoint must be SSL_IS_CLIENT or SSL_IS_SERVER

Definition at line 1584 of file ssl_tls.c.

References _ssl_context::endpoint.

Referenced by main(), and ssl_test().

13.135.1.29 int ssl_set_hostname (ssl_context * ssl, char * hostname)

Set hostname for ServerName TLS Extension.

Parameters:

ssl SSL context
hostname the server hostname

Returns:

0 if successful

Definition at line 1674 of file ssl_tls.c.

References `_ssl_context::hostname`, `_ssl_context::hostname_len`, and `XYSSL_ERR_SSL_BAD_INPUT_DATA`.

Referenced by `main()`.

13.135.1.30 void ssl_set_own_cert (ssl_context * ssl, x509_cert * own_cert, rsa_context * rsa_key)

Set own certificate and private key.

Parameters:

ssl SSL context
own_cert own public certificate
rsa_key own private RSA key

Definition at line 1648 of file ssl_tls.c.

References `_ssl_context::own_cert`, and `_ssl_context::rsa_key`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.31 void ssl_set_rng (ssl_context * ssl, int(*) (void *) f_rng, void * p_rng)

Set the random number generator callback.

Parameters:

ssl SSL context
f_rng RNG function
p_rng RNG parameter

Definition at line 1594 of file ssl_tls.c.

References `_ssl_context::f_rng`, and `_ssl_context::p_rng`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.32 void ssl_set_scb (ssl_context * ssl, int(*) (ssl_context *) s_get, int(*) (ssl_context *) s_set)

Set the session callbacks (server-side only).

Parameters:

ssl SSL context

s_get session get callback

s_set session set callback

Definition at line 1620 of file ssl_tls.c.

References `_ssl_context::s_get`, and `_ssl_context::s_set`.

Referenced by `main()`.

13.135.1.33 void ssl_set_session (ssl_context * ssl, int resume, int timeout, ssl_session * session)

Set the session resuming flag, timeout and data.

Parameters:

ssl SSL context

resume if 0 (default), the session will not be resumed

timeout session timeout in seconds, or 0 (no timeout)

session session context

Definition at line 1628 of file ssl_tls.c.

References `_ssl_context::resume`, `_ssl_context::session`, and `_ssl_context::timeout`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.34 int ssl_write (ssl_context * ssl, unsigned char * buf, int len)

Write exactly 'len' application data bytes.

Parameters:

ssl SSL context

buf buffer holding the data

len how many bytes must be written

Returns:

This function returns the number of bytes written, or a negative error code.

Note:

When this function returns `XYSSL_ERR_NET_TRY_AGAIN`, it must be called later with the **same** arguments, until it returns a positive value.

Definition at line 1857 of file ssl_tls.c.

References `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_flush_output()`, `ssl_handshake()`, `SSL_HANDSHAKE_OVER`, `SSL_MAX_CONTENT_LEN`, `SSL_MSG_APPLICATION_DATA`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `main()`, and `ssl_test()`.

13.135.1.35 int ssl_write_certificate (ssl_context * ssl)

Definition at line 1044 of file ssl_tls.c.

References `_ssl_context::client_auth`, `_ssl_context::endpoint`, `_x509_buf::len`, `_ssl_context::minor_ver`, `_x509_cert::next`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::own_cert`, `_x509_buf::p`, `_x509_cert::raw`, `SSL_ALERT_NO_CERTIFICATE`, `SSL_ALERT_WARNING`, `SSL_DEBUG_CRT`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HS_CERTIFICATE`, `SSL_IS_CLIENT`, `SSL_MAX_CONTENT_LEN`, `SSL_MINOR_VERSION_0`, `SSL_MSG_ALERT`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, `_ssl_context::state`, `XYSSL_ERR_SSL_CERTIFICATE_REQUIRED`, and `XYSSL_ERR_SSL_CERTIFICATE_TOO_LARGE`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.135.1.36 int ssl_write_change_cipher_spec (ssl_context * ssl)

Definition at line 1287 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_MSG_CHANGE_CIPHER_SPEC`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.135.1.37 int ssl_write_finished (ssl_context * ssl)

Definition at line 1433 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::endpoint`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `md5()`, `_ssl_context::minor_ver`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `_ssl_context::resume`, `sha1()`, `ssl_calc_finished()`, `SSL_CLIENT_CHANGE_CIPHER_SPEC`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `SSL_HANDSHAKE_OVER`, `SSL_HS_FINISHED`, `SSL_IS_CLIENT`, `SSL_MINOR_VERSION_0`, `SSL_MSG_HANDSHAKE`, `ssl_write_record()`, and `_ssl_context::state`.

Referenced by `ssl_handshake_client()`, and `ssl_handshake_server()`.

13.135.1.38 int ssl_write_record (ssl_context * ssl)

Definition at line 786 of file ssl_tls.c.

References `_ssl_context::do_crypt`, `_ssl_context::fin_md5`, `_ssl_context::fin_sha1`, `_ssl_context::major_ver`, `md5_update()`, `_ssl_context::minor_ver`, `_ssl_context::out_hdr`, `_ssl_context::out_left`, `_ssl_context::out_msg`, `_ssl_context::out_msglen`, `_ssl_context::out_msgtype`, `sha1_update()`, `SSL_DEBUG_BUF`, `SSL_DEBUG_MSG`, `SSL_DEBUG_RET`, `ssl_encrypt_buf()`, `ssl_flush_output()`, and `SSL_MSG_HANDSHAKE`.

Referenced by `ssl_close_notify()`, `ssl_write()`, `ssl_write_certificate()`, `ssl_write_certificate_request()`, `ssl_write_certificate_verify()`, `ssl_write_change_cipher_spec()`, `ssl_write_client_hello()`, `ssl_write_client_key_exchange()`, `ssl_write_finished()`, `ssl_write_server_hello()`, `ssl_write_server_hello_done()`, and `ssl_write_server_key_exchange()`.

13.135.1.39 static int tls1_prf (unsigned char * *secret*, int *slen*, char * *label*, unsigned char * *random*, int *rlen*, unsigned char * *dstbuf*, int *dlen*) [static]

Definition at line 46 of file ssl_tls.c.

References md5_hmac(), S1, S2, sha1_hmac(), and XYSSL_ERR_SSL_BAD_INPUT_DATA.

Referenced by ssl_calc_finished(), and ssl_derive_keys().

13.135.2 Variable Documentation**13.135.2.1 int ssl_default_ciphers[]**

Initial value:

```
{  
  
    SSL_EDH_RSA_AES_256_SHA,  
  
    SSL_EDH_RSA_DES_168_SHA,  
  
    SSL_RSA_AES_128_SHA,  
    SSL_RSA_AES_256_SHA,  
  
    SSL_RSA_DES_168_SHA,  
  
    SSL_RSA_RC4_128_SHA,  
    SSL_RSA_RC4_128_MD5,  
  
    0  
}
```

Definition at line 1739 of file ssl_tls.c.

Referenced by main(), and ssl_test().

13.136 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/timing.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/timing.h"
#include <unistd.h>
#include <sys/types.h>
#include <sys/time.h>
#include <signal.h>
#include <time.h>
```

Data Structures

- struct [_hr_time](#)

Functions

- unsigned long [hardclock](#) (void)
Return the CPU cycle counter value.
- unsigned long [get_timer](#) (struct [hr_time](#) *val, int reset)
Return the elapsed time in milliseconds.
- static void [sighandler](#) (int signum)
- void [set_alarm](#) (int seconds)
Setup an alarm clock.
- void [m_sleep](#) (int milliseconds)
Sleep for a certain amount of time.

Variables

- static int [hardclock_init](#) = 0
- static struct timeval [tv_init](#)
- int [alarmed](#) = 0

13.136.1 Function Documentation

13.136.1.1 unsigned long get_timer (struct hr_time * val, int reset)

Return the elapsed time in milliseconds.

Parameters:

val points to a timer structure

reset if set to 1, the timer is restarted

Definition at line 205 of file timing.c.

References `_hr_time::start`.

Referenced by `ssl_test()`.

13.136.1.2 unsigned long hardclock (void)

Return the CPU cycle counter value.

Definition at line 136 of file timing.c.

References `tv_init`.

Referenced by `main()`.

13.136.1.3 void m_sleep (int milliseconds)

Sleep for a certain amount of time.

Definition at line 238 of file timing.c.

13.136.1.4 void set_alarm (int seconds)

Setup an alarm clock.

Parameters:

seconds delay before the "alarmed" flag is set

Definition at line 231 of file timing.c.

References `sighandler()`.

Referenced by `main()`.

13.136.1.5 static void sighandler (int signum) [static]

Definition at line 225 of file timing.c.

Referenced by `set_alarm()`.

13.136.2 Variable Documentation

13.136.2.1 int alarmed = 0

Definition at line 159 of file timing.c.

Referenced by `main()`.

13.136.2.2 int hardclock_init = 0 [static]

Definition at line 133 of file timing.c.

13.136.2.3 struct timeval tv_init [static]

Definition at line 134 of file timing.c.

Referenced by hardclock().

13.137 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/library/x509parse.c File Reference

```
#include "xyssl/config.h"
#include "xyssl/x509.h"
#include "xyssl/base64.h"
#include "xyssl/des.h"
#include "xyssl/md2.h"
#include "xyssl/md4.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include "xyssl/certs.h"
```

Functions

- static [int](#) [asn1_get_len](#) (unsigned char **p, unsigned char *end, [int](#) *len)
- static [int](#) [asn1_get_tag](#) (unsigned char **p, unsigned char *end, [int](#) *len, [int](#) tag)
- static [int](#) [asn1_get_bool](#) (unsigned char **p, unsigned char *end, [int](#) *val)
- static [int](#) [asn1_get_int](#) (unsigned char **p, unsigned char *end, [int](#) *val)
- static [int](#) [asn1_get_mpi](#) (unsigned char **p, unsigned char *end, [mpi](#) *X)
- static [int](#) [x509_get_version](#) (unsigned char **p, unsigned char *end, [int](#) *ver)
- static [int](#) [x509_get_serial](#) (unsigned char **p, unsigned char *end, [x509_buf](#) *serial)
- static [int](#) [x509_get_alg](#) (unsigned char **p, unsigned char *end, [x509_buf](#) *alg)
- static [int](#) [x509_get_name](#) (unsigned char **p, unsigned char *end, [x509_name](#) *cur)
- static [int](#) [x509_get_dates](#) (unsigned char **p, unsigned char *end, [x509_time](#) *from, [x509_time](#) *to)
- static [int](#) [x509_get_pubkey](#) (unsigned char **p, unsigned char *end, [x509_buf](#) *pk_alg_oid, [mpi](#) *N, [mpi](#) *E)
- static [int](#) [x509_get_sig](#) (unsigned char **p, unsigned char *end, [x509_buf](#) *sig)
- static [int](#) [x509_get_uid](#) (unsigned char **p, unsigned char *end, [x509_buf](#) *uid, [int](#) n)
- static [int](#) [x509_get_ext](#) (unsigned char **p, unsigned char *end, [x509_buf](#) *ext, [int](#) *ca_istrue, [int](#) *max_pathlen)
- [int](#) [x509parse_crt](#) ([x509_cert](#) *chain, unsigned char *buf, [int](#) buflen)

Parse one or more certificates and add them to the chained list.
- [int](#) [x509parse_crtfile](#) ([x509_cert](#) *chain, char *path)

Load one or more certificates and add them to the chained list.
- static [int](#) [x509_get_iv](#) (unsigned char *s, unsigned char iv[8])
- static void [x509_des3_decrypt](#) (unsigned char des3_iv[8], unsigned char *buf, [int](#) buflen, unsigned char *pwd, [int](#) pwrlen)

- `int x509parse_key (rsa_context *rsa, unsigned char *buf, int buflen, unsigned char *pwd, int pwrlen)`
Parse a private RSA key.
- `int x509parse_keyfile (rsa_context *rsa, char *path, char *pwd)`
Load and parse a private RSA key.
- `int x509parse_dn_gets (char *buf, char *end, x509_name *dn)`
Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.
- `char * x509parse_cert_info (char *prefix, x509_cert *crt)`
Returns an informational string about the certificate.
- `int x509parse_expired (x509_cert *crt)`
Return 0 if the certificate is still valid, or BADCERT_EXPIRED.
- `static void x509_hash (unsigned char *in, int len, int alg, unsigned char *out)`
- `int x509parse_verify (x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)`
Verify the certificate signature.
- `void x509_free (x509_cert *crt)`
Unallocate all certificate data.
- `int x509_self_test (int verbose)`
Checkup routine.

13.137.1 Function Documentation

13.137.1.1 `static int asn1_get_bool (unsigned char **p, unsigned char *end, int *val) [static]`

Definition at line 108 of file x509parse.c.

References ASN1_BOOLEAN, `asn1_get_tag()`, and XYSSL_ERR_ASN1_INVALID_LENGTH.

Referenced by `x509_get_ext()`.

13.137.1.2 `static int asn1_get_int (unsigned char **p, unsigned char *end, int *val) [static]`

Definition at line 126 of file x509parse.c.

References `asn1_get_tag()`, ASN1_INTEGER, and XYSSL_ERR_ASN1_INVALID_LENGTH.

Referenced by `x509_get_ext()`, `x509_get_version()`, and `x509parse_key()`.

13.137.1.3 `static int asn1_get_len (unsigned char **p, unsigned char *end, int *len) [static]`

Definition at line 52 of file x509parse.c.

References XYSSL_ERR_ASN1_INVALID_LENGTH, and XYSSL_ERR_ASN1_OUT_OF_DATA.

Referenced by `asn1_get_tag()`, `x509_get_name()`, and `x509_get_serial()`.

13.137.1.4 static int asn1_get_mpi (unsigned char ** *p*, unsigned char * *end*, mpi * *X*)
[static]

Definition at line 149 of file x509parse.c.

References `asn1_get_tag()`, `ASN1_INTEGER`, and `mpi_read_binary()`.

Referenced by `x509_get_pubkey()`, and `x509parse_key()`.

13.137.1.5 static int asn1_get_tag (unsigned char ** *p*, unsigned char * *end*, int * *len*, int *tag*)
[static]

Definition at line 93 of file x509parse.c.

References `asn1_get_len()`, `XYSSL_ERR_ASN1_OUT_OF_DATA`, and `XYSSL_ERR_ASN1_UNEXPECTED_TAG`.

Referenced by `asn1_get_bool()`, `asn1_get_int()`, `asn1_get_mpi()`, `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_sig()`, `x509_get_uid()`, `x509_get_version()`, `x509parse_crt()`, and `x509parse_key()`.

13.137.1.6 static void x509_des3_decrypt (unsigned char *des3_iv*[8], unsigned char * *buf*, int *buflen*, unsigned char * *pwd*, int *pwdlen*) **[static]**

Definition at line 1042 of file x509parse.c.

References `des3_crypt_cbc()`, `des3_set3key_dec()`, `DES_DECRYPT`, `md5_finish()`, `md5_starts()`, and `md5_update()`.

Referenced by `x509parse_key()`.

13.137.1.7 void x509_free (x509_cert * *crt*)

Unallocate all certificate data.

Definition at line 1613 of file x509parse.c.

References `_x509_cert::issuer`, `_x509_buf::len`, `_x509_cert::next`, `_x509_name::next`, `_x509_buf::p`, `_x509_cert::raw`, `_x509_cert::rsa`, `rsa_free()`, and `_x509_cert::subject`.

Referenced by `main()`, `ssl_free()`, `ssl_test()`, `x509_self_test()`, and `x509parse_crt()`.

13.137.1.8 static int x509_get_alg (unsigned char ** *p*, unsigned char * *end*, x509_buf * *alg*)
[static]

Definition at line 229 of file x509parse.c.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_NULL`, `ASN1_OID`, `ASN1_SEQUENCE`, `_x509_buf::len`, `_x509_buf::p`, `_x509_buf::tag`, `XYSSL_ERR_ASN1_LENGTH_MISMATCH`, and `XYSSL_ERR_X509_CERT_INVALID_ALG`.

Referenced by `x509_get_pubkey()`, and `x509parse_crt()`.

13.137.1.9 `static int x509_get_dates (unsigned char **p, unsigned char *end, x509_time *from, x509_time *to) [static]`

Definition at line 358 of file x509parse.c.

References ASN1_CONSTRUCTED, asn1_get_tag(), ASN1_SEQUENCE, ASN1_UTC_TIME, _x509_time::day, _x509_time::hour, _x509_time::min, _x509_time::mon, _x509_time::sec, XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_X509_CERT_INVALID_DATE, and _x509_time::year.

Referenced by x509parse_crt().

13.137.1.10 `static int x509_get_ext (unsigned char **p, unsigned char *end, x509_buf *ext, int *ca_istrue, int *max_pathlen) [static]`

Definition at line 530 of file x509parse.c.

References ASN1_CONSTRUCTED, ASN1_CONTEXT_SPECIFIC, asn1_get_bool(), asn1_get_int(), asn1_get_tag(), ASN1_OCTET_STRING, ASN1_SEQUENCE, _x509_buf::len, _x509_buf::p, _x509_buf::tag, XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_ASN1_UNEXPECTED_TAG, and XYSSL_ERR_X509_CERT_INVALID_EXTENSIONS.

Referenced by x509parse_crt().

13.137.1.11 `static int x509_get_iv (unsigned char *s, unsigned char iv[8]) [static]`

Definition at line 1018 of file x509parse.c.

References XYSSL_ERR_X509_KEY_INVALID_ENC_IV.

Referenced by x509parse_key().

13.137.1.12 `static int x509_get_name (unsigned char **p, unsigned char *end, x509_name *cur) [static]`

Definition at line 276 of file x509parse.c.

References ASN1_BMP_STRING, ASN1_CONSTRUCTED, asn1_get_len(), asn1_get_tag(), ASN1_IA5_STRING, ASN1_OID, ASN1_PRINTABLE_STRING, ASN1_SEQUENCE, ASN1_SET, ASN1_T61_STRING, ASN1_UNIVERSAL_STRING, ASN1_UTF8_STRING, _x509_buf::len, _x509_name::next, _x509_name::oid, _x509_buf::p, _x509_buf::tag, _x509_name::val, XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_ASN1_OUT_OF_DATA, XYSSL_ERR_ASN1_UNEXPECTED_TAG, and XYSSL_ERR_X509_CERT_INVALID_NAME.

Referenced by x509parse_crt().

13.137.1.13 `static int x509_get_pubkey (unsigned char **p, unsigned char *end, x509_buf *pk_alg_oid, mpi *N, mpi *E) [static]`

Definition at line 421 of file x509parse.c.

References ASN1_BIT_STRING, ASN1_CONSTRUCTED, asn1_get_mpi(), asn1_get_tag(), ASN1_SEQUENCE, _x509_buf::len, OID_PKCS1_RSA, _x509_buf::p, x509_get_alg(), XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_ASN1_OUT_OF_DATA, XYSSL_ERR_X509_CERT_INVALID_PUBKEY, and XYSSL_ERR_X509_CERT_UNKNOWN_PK_ALG.

Referenced by x509parse_crt().

13.137.1.14 `static int x509_get_serial (unsigned char **p, unsigned char *end, x509_buf *serial)`
`[static]`

Definition at line 198 of file x509parse.c.

References ASN1_CONTEXT_SPECIFIC, asn1_get_len(), ASN1_INTEGER, ASN1_PRIMITIVE, _x509_buf::len, _x509_buf::p, _x509_buf::tag, XYSSL_ERR_ASN1_OUT_OF_DATA, XYSSL_ERR_ASN1_UNEXPECTED_TAG, and XYSSL_ERR_X509_CERT_INVALID_SERIAL.

Referenced by x509parse_crt().

13.137.1.15 `static int x509_get_sig (unsigned char **p, unsigned char *end, x509_buf *sig)`
`[static]`

Definition at line 476 of file x509parse.c.

References ASN1_BIT_STRING, asn1_get_tag(), _x509_buf::len, _x509_buf::p, _x509_buf::tag, and XYSSL_ERR_X509_CERT_INVALID_SIGNATURE.

Referenced by x509parse_crt().

13.137.1.16 `static int x509_get_uid (unsigned char **p, unsigned char *end, x509_buf *uid, int n)`
`[static]`

Definition at line 501 of file x509parse.c.

References ASN1_CONSTRUCTED, ASN1_CONTEXT_SPECIFIC, asn1_get_tag(), _x509_buf::len, _x509_buf::p, _x509_buf::tag, and XYSSL_ERR_ASN1_UNEXPECTED_TAG.

Referenced by x509parse_crt().

13.137.1.17 `static int x509_get_version (unsigned char **p, unsigned char *end, int *ver)`
`[static]`

Definition at line 168 of file x509parse.c.

References ASN1_CONSTRUCTED, ASN1_CONTEXT_SPECIFIC, asn1_get_int(), asn1_get_tag(), XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_ASN1_UNEXPECTED_TAG, and XYSSL_ERR_X509_CERT_INVALID_VERSION.

Referenced by x509parse_crt().

13.137.1.18 `static void x509_hash (unsigned char *in, int len, int alg, unsigned char *out)`
`[static]`

Definition at line 1481 of file x509parse.c.

References md2(), md4(), md5(), RSA_MD2, RSA_MD4, RSA_MD5, RSA_SHA1, and sha1().

Referenced by x509parse_verify().

13.137.1.19 `int x509_self_test (int verbose)`

Checkup routine.

Returns:

0 if successful, or 1 if the test failed

Definition at line 1675 of file x509parse.c.

References `rsa_free()`, `test_ca_cert`, `test_ca_key`, `test_ca_pwd`, `test_cli_cert`, `x509_free()`, `x509parse_cert()`, `x509parse_key()`, and `x509parse_verify()`.

Referenced by `main()`.

13.137.1.20 char* x509parse_cert_info (char * prefix, x509_cert * crt)

Returns an informational string about the certificate.

Definition at line 1399 of file x509parse.c.

References `buf`, `_x509_time::day`, `_x509_time::hour`, `_x509_cert::issuer`, `_x509_buf::len`, `_x509_time::min`, `_x509_time::mon`, `mpi::n`, `rsa_context::N`, `_x509_buf::p`, `_x509_cert::rsa`, `RSA_MD2`, `RSA_MD4`, `RSA_MD5`, `RSA_SHA1`, `_x509_time::sec`, `_x509_cert::serial`, `_x509_cert::sig_oid1`, `_x509_cert::subject`, `_x509_cert::valid_from`, `_x509_cert::valid_to`, `_x509_cert::version`, `x509parse_dn_gets()`, and `_x509_time::year`.

Referenced by `debug_print_cert()`, and `main()`.

13.137.1.21 int x509parse_cert (x509_cert * crt, unsigned char * buf, int buflen)

Parse one or more certificates and add them to the chained list.

Parameters:

chain points to the start of the chain
buf buffer holding the certificate data
buflen size of the buffer

Returns:

0 if successful, or a specific X509 error code

Definition at line 647 of file x509parse.c.

References `ASN1_CONSTRUCTED`, `asn1_get_tag()`, `ASN1_SEQUENCE`, `base64_decode()`, `_x509_cert::ca_istrue`, `rsa_context::E`, `_x509_cert::issuer`, `_x509_cert::issuer_id`, `_x509_cert::issuer_raw`, `rsa_context::len`, `_x509_buf::len`, `_x509_cert::max_pathlen`, `mpi_size()`, `rsa_context::N`, `_x509_cert::next`, `OID_PKCS1`, `_x509_buf::p`, `_x509_cert::pk_oid`, `_x509_cert::raw`, `_x509_cert::rsa`, `rsa_check_pubkey()`, `_x509_cert::serial`, `_x509_cert::sig`, `_x509_cert::sig_oid1`, `_x509_cert::sig_oid2`, `_x509_cert::subject`, `_x509_cert::subject_id`, `_x509_cert::subject_raw`, `_x509_cert::tbs`, `_x509_cert::v3_ext`, `_x509_cert::valid_from`, `_x509_cert::valid_to`, `_x509_cert::version`, `x509_free()`, `x509_get_alg()`, `x509_get_dates()`, `x509_get_ext()`, `x509_get_name()`, `x509_get_pubkey()`, `x509_get_serial()`, `x509_get_sig()`, `x509_get_uid()`, `x509_get_version()`, `x509parse_cert()`, `XYSSL_ERR_ASN1_LENGTH_MISMATCH`, `XYSSL_ERR_BASE64_INVALID_CHARACTER`, `XYSSL_ERR_X509_CERT_INVALID_FORMAT`, `XYSSL_ERR_X509_CERT_INVALID_PEM`, `XYSSL_ERR_X509_CERT_SIG_MISMATCH`, `XYSSL_ERR_X509_CERT_UNKNOWN_SIG_ALG`, and `XYSSL_ERR_X509_CERT_UNKNOWN_VERSION`.

Referenced by `main()`, `ssl_parse_certificate()`, `ssl_test()`, `x509_self_test()`, `x509parse_cert()`, and `x509parse_crtdfile()`.

13.137.1.22 `int x509parse_crtfile(x509_cert *crt, char *path)`

Load one or more certificates and add them to the chained list.

Parameters:

chain points to the start of the chain
path filename to read the certificates from

Returns:

0 if successful, or a specific X509 error code

Definition at line 979 of file x509parse.c.

References buf, f, and x509parse_cert().

13.137.1.23 `int x509parse_dn_gets(char *buf, char *end, x509_name *dn)`

Store the certificate DN in printable form into buf; no more than (end - buf) characters will be written.

Definition at line 1316 of file x509parse.c.

References _x509_buf::len, _x509_name::next, _x509_name::oid, OID_PKCS9, OID_X520, _x509_buf::p, PKCS9_EMAIL, _x509_name::val, X520_COMMON_NAME, X520_COUNTRY, X520_LOCALITY, X520_ORG_UNIT, X520_ORGANIZATION, and X520_STATE.

Referenced by x509parse_cert_info().

13.137.1.24 `int x509parse_expired(x509_cert *crt)`

Return 0 if the certificate is still valid, or BADCERT_EXPIRED.

Definition at line 1458 of file x509parse.c.

References BADCERT_EXPIRED, _x509_time::day, _x509_time::mon, _x509_cert::valid_to, and _x509_time::year.

Referenced by x509parse_verify().

13.137.1.25 `int x509parse_key(rsa_context *rsa, unsigned char *buf, int buflen, unsigned char *pwd, int pwrlen)`

Parse a private RSA key.

Parameters:

rsa RSA context to be initialized
buf input buffer
buflen size of the buffer
pwd password for decryption (optional)
pwrlen size of the password

Returns:

0 if successful, or a specific X509 error code

Definition at line 1082 of file x509parse.c.

References ASN1_CONSTRUCTED, asn1_get_int(), asn1_get_mpi(), asn1_get_tag(), ASN1_SEQUENCE, base64_decode(), rsa_context::D, rsa_context::DP, rsa_context::DQ, rsa_context::E, rsa_context::len, mpi_size(), rsa_context::N, rsa_context::P, rsa_context::Q, rsa_context::QP, rsa_check_privkey(), rsa_free(), rsa_context::ver, x509_des3_decrypt(), x509_get_iv(), XYSSL_ERR_ASN1_LENGTH_MISMATCH, XYSSL_ERR_BASE64_INVALID_CHARACTER, XYSSL_ERR_X509_FEATURE_UNAVAILABLE, XYSSL_ERR_X509_KEY_INVALID_ENC_IV, XYSSL_ERR_X509_KEY_INVALID_FORMAT, XYSSL_ERR_X509_KEY_INVALID_PEM, XYSSL_ERR_X509_KEY_INVALID_VERSION, XYSSL_ERR_X509_KEY_PASSWORD_MISMATCH, XYSSL_ERR_X509_KEY_PASSWORD_REQUIRED, and XYSSL_ERR_X509_KEY_UNKNOWN_ENC_ALG.

Referenced by main(), ssl_test(), x509_self_test(), and x509parse_keyfile().

13.137.1.26 int x509parse_keyfile (rsa_context *rsa, char *path, char *password)

Load and parse a private RSA key.

Parameters:

- rsa* RSA context to be initialized
- path* filename to read the private key from
- pwd* password to decrypt the file (can be NULL)

Returns:

- 0 if successful, or a specific X509 error code

Definition at line 1269 of file x509parse.c.

References buf, f, and x509parse_key().

13.137.1.27 int x509parse_verify (x509_cert *crt, x509_cert *trust_ca, char *cn, int *flags)

Verify the certificate signature.

Parameters:

- crt* a certificate to be verified
- trust_ca* the trusted CA chain
- cn* expected Common Name (can be set to NULL if the CN must not be verified)
- flags* result of the verification

Returns:

- 0 if successful or XYSSL_ERR_X509_SIG_VERIFY_FAILED, in which case *flags will have one or more of the following values set: BADCERT_EXPIRED -- BADCERT_REVOKED -- BADCERT_CN_MISMATCH -- BADCERT_NOT_TRUSTED

Note:

- TODO: add two arguments, depth and crl

Definition at line 1503 of file x509parse.c.

References BADCERT_CN_MISMATCH, BADCERT_NOT_TRUSTED, _x509_cert::ca_istrue, cur, _x509_cert::issuer_raw, _x509_buf::len, _x509_cert::max_pathlen, _x509_cert::next, _x509_name::next, _x509_name::oid, OID_CN, _x509_buf::p, _x509_cert::rsa, rsa_pkcs1_verify(), RSA_PUBLIC, _x509_cert::sig, _x509_cert::sig_oid1, _x509_cert::subject, _x509_cert::subject_raw, _x509_cert::tbs, _x509_name::val, _x509_cert::version, x509_hash(), x509parse_expired(), and XYSSL_ERR_X509_CERT_VERIFY_FAILED.

Referenced by ssl_parse_certificate(), and x509_self_test().

13.138 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/aes/aescrypt2.c File Reference

```
#include <sys/types.h>
#include <unistd.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include "xyssl/aes.h"
#include "xyssl/sha2.h"
```

Defines

- #define [MODE_ENCRYPT](#) 0
- #define [MODE_DECRYPT](#) 1
- #define [USAGE](#)

Functions

- [int main](#) ([int](#) argc, [char](#) *argv[])

13.138.1 Define Documentation

13.138.1.1 #define MODE_DECRYPT 1

Definition at line 42 of file aescrypt2.c.

13.138.1.2 #define MODE_ENCRYPT 0

Definition at line 41 of file aescrypt2.c.

13.138.1.3 #define USAGE

Value:

```
"\n aescrypt2 <mode> <input filename> <output filename> <key>\n" \
"\n <mode>: 0 = encrypt, 1 = decrypt\n" \
"\n example: aescrypt2 0 file file.aes hex:E76B2413958B00E193\n" \
"\n"
```

Definition at line 44 of file aescrypt2.c.

Referenced by [main\(\)](#).

13.138.2 Function Documentation

13.138.2.1 `int main (int argc, char * argv[])`

Definition at line 50 of file aescrypt2.c.

References `aes_crypt_ecb()`, `AES_DECRYPT`, `AES_ENCRYPT`, `aes_setkey_dec()`, `aes_setkey_enc()`, `int`, `MODE_DECRYPT`, `MODE_ENCRYPT`, `sha2_finish()`, `sha2_hmac_finish()`, `sha2_hmac_starts()`, `sha2_hmac_update()`, `sha2_starts()`, `sha2_update()`, and `USAGE`.

13.139 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/hello.c File Reference

```
#include <stdio.h>
#include "xyssl/md5.h"
```

Functions

- [int main](#) (void)

13.139.1 Function Documentation

13.139.1.1 int main (void)

Definition at line 29 of file hello.c.

References [md5\(\)](#).

13.140 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/md5sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/md5.h"
```

Functions

- static [int md5_wrapper](#) (char *filename, unsigned char *sum)
- static [int md5_print](#) (char *filename)
- static [int md5_check](#) (char *filename)
- [int main](#) (int argc, char *argv[])

13.140.1 Function Documentation

13.140.1.1 [int main](#) (int *argc*, char * *argv*[])

Definition at line 131 of file md5sum.c.

References [md5_check\(\)](#), and [md5_print\(\)](#).

13.140.1.2 [static int md5_check](#) (char * *filename*) [**static**]

Definition at line 58 of file md5sum.c.

References [buf](#), [f](#), and [md5_wrapper\(\)](#).

Referenced by [main\(\)](#).

13.140.1.3 [static int md5_print](#) (char * *filename*) [**static**]

Definition at line 43 of file md5sum.c.

References [md5_wrapper\(\)](#).

Referenced by [main\(\)](#).

13.140.1.4 [static int md5_wrapper](#) (char * *filename*, unsigned char * *sum*) [**static**]

Definition at line 30 of file md5sum.c.

References [md5_file\(\)](#).

Referenced by [md5_check\(\)](#), and [md5_print\(\)](#).

13.141 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/sha1sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha1.h"
```

Functions

- static [int sha1_wrapper](#) (char *filename, unsigned char *sum)
- static [int sha1_print](#) (char *filename)
- static [int sha1_check](#) (char *filename)
- [int main](#) (int argc, char *argv[])

13.141.1 Function Documentation

13.141.1.1 [int main](#) (int *argc*, char * *argv*[])

Definition at line 131 of file sha1sum.c.

References [sha1_check\(\)](#), and [sha1_print\(\)](#).

13.141.1.2 [static int sha1_check](#) (char * *filename*) [**static**]

Definition at line 58 of file sha1sum.c.

References [buf](#), [f](#), and [sha1_wrapper\(\)](#).

Referenced by [main\(\)](#).

13.141.1.3 [static int sha1_print](#) (char * *filename*) [**static**]

Definition at line 43 of file sha1sum.c.

References [sha1_wrapper\(\)](#).

Referenced by [main\(\)](#).

13.141.1.4 [static int sha1_wrapper](#) (char * *filename*, unsigned char * *sum*) [**static**]

Definition at line 30 of file sha1sum.c.

References [sha1_file\(\)](#).

Referenced by [sha1_check\(\)](#), and [sha1_print\(\)](#).

13.142 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/hash/sha2sum.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/sha2.h"
```

Functions

- static [int sha2_wrapper](#) (char *filename, unsigned char *sum)
- static [int sha2_print](#) (char *filename)
- static [int sha2_check](#) (char *filename)
- [int main](#) (int argc, char *argv[])

13.142.1 Function Documentation

13.142.1.1 [int main](#) (int *argc*, char * *argv*[])

Definition at line 131 of file sha2sum.c.

References [sha2_check\(\)](#), and [sha2_print\(\)](#).

13.142.1.2 [static int sha2_check](#) (char * *filename*) [**static**]

Definition at line 58 of file sha2sum.c.

References [buf](#), [f](#), and [sha2_wrapper\(\)](#).

Referenced by [main\(\)](#).

13.142.1.3 [static int sha2_print](#) (char * *filename*) [**static**]

Definition at line 43 of file sha2sum.c.

References [sha2_wrapper\(\)](#).

Referenced by [main\(\)](#).

13.142.1.4 [static int sha2_wrapper](#) (char * *filename*, unsigned char * *sum*) [**static**]

Definition at line 30 of file sha2sum.c.

References [sha2_file\(\)](#).

Referenced by [sha2_check\(\)](#), and [sha2_print\(\)](#).

13.143 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/dh_client.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/aes.h"
#include "xyssl/dhm.h"
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
#include "xyssl/havege.h"
```

Defines

- `#define SERVER_NAME "localhost"`
- `#define SERVER_PORT 11999`

Functions

- `int main (void)`

13.143.1 Define Documentation

13.143.1.1 `#define SERVER_NAME "localhost"`

Definition at line 35 of file `dh_client.c`.

Referenced by `main()`.

13.143.1.2 `#define SERVER_PORT 11999`

Definition at line 36 of file `dh_client.c`.

Referenced by `main()`.

13.143.2 Function Documentation

13.143.2.1 `int main (void)`

Definition at line 38 of file `dh_client.c`.

References `aes_crypt_ecb()`, `AES_DECRYPT`, `aes_setkey_dec()`, `buf`, `dhm_calc_secret()`, `dhm_free()`, `dhm_make_public()`, `dhm_read_params()`, `rsa_context::E`, `f`, `havege_init()`, `havege_rand()`, `dhm_context::len`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `net_close()`, `net_connect()`, `net_recv()`, `net_send()`, `rsa_free()`, `rsa_init()`, `rsa_pkcs1_verify()`, `RSA_PKCS_V15`, `RSA_PUBLIC`, `RSA_SHA1`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, and `sha1()`.

13.144 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/dh_genprime.c File Reference

```
#include <stdio.h>
#include "xyssl/bignum.h"
#include "xyssl/config.h"
#include "xyssl/havege.h"
```

Defines

- `#define DH_P_SIZE 1024`
- `#define GENERATOR "4"`

Functions

- `int main (void)`

13.144.1 Define Documentation

13.144.1.1 `#define DH_P_SIZE 1024`

Definition at line 35 of file `dh_genprime.c`.

Referenced by `main()`.

13.144.1.2 `#define GENERATOR "4"`

Definition at line 36 of file `dh_genprime.c`.

Referenced by `main()`.

13.144.2 Function Documentation

13.144.2.1 `int main (void)`

Definition at line 38 of file `dh_genprime.c`.

References `DH_P_SIZE`, `GENERATOR`, `havege_init()`, `havege_rand()`, `mpi_div_int()`, `mpi_free()`, `mpi_gen_prime()`, `mpi_init()`, `mpi_is_prime()`, `mpi_read_string()`, `mpi_sub_int()`, `mpi_write_file()`, and `P`.

13.145 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/dh_server.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/aes.h"
#include "xyssl/dhm.h"
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
#include "xyssl/havege.h"
```

Defines

- `#define SERVER_PORT 11999`
- `#define PLAINTEXT "==Hello there!=="`

Functions

- `int main (void)`

13.145.1 Define Documentation

13.145.1.1 `#define PLAINTEXT "==Hello there!=="`

Definition at line 36 of file `dh_server.c`.

Referenced by `main()`.

13.145.1.2 `#define SERVER_PORT 11999`

Definition at line 35 of file `dh_server.c`.

13.145.2 Function Documentation

13.145.2.1 `int main (void)`

Definition at line 38 of file `dh_server.c`.

References `aes_crypt_ecb()`, `AES_ENCRYPT`, `aes_setkey_enc()`, `buf`, `client_fd`, `rsa_context::D`, `dhm_calc_secret()`, `dhm_free()`, `dhm_make_params()`, `dhm_read_public()`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `f`, `dhm_context::G`, `havege_init()`, `havege_rand()`, `dhm_context::len`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `net_accept()`, `net_bind()`, `net_close()`, `net_recv()`, `net_send()`, `dhm_context::P`, `rsa_context::P`, `PLAINTEXT`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `rsa_init()`, `rsa_pkcs1_sign()`, `RSA_PKCS_V15`, `RSA_PRIVATE`, `RSA_SHA1`, `SERVER_PORT`, and `sha1()`.

13.146 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/mpi_demo.c File Reference

```
#include <stdio.h>
#include "xyssl/bignum.h"
```

Functions

- `int main` (void)

13.146.1 Function Documentation

13.146.1.1 `int main` (void)

Definition at line 29 of file `mpi_demo.c`.

References `mpi_exp_mod()`, `mpi_free()`, `mpi_init()`, `mpi_inv_mod()`, `mpi_mul_mpi()`, `mpi_read_string()`, `mpi_sub_int()`, `mpi_write_file()`, and `P`.

13.147 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/rsa_genkey.c File Reference

```
#include <stdio.h>
#include "xyssl/havege.h"
#include "xyssl/bignum.h"
#include "xyssl/x509.h"
#include "xyssl/rsa.h"
```

Defines

- `#define KEY_SIZE 1024`
- `#define EXPONENT 65537`

Functions

- `int main (void)`

13.147.1 Define Documentation

13.147.1.1 `#define EXPONENT 65537`

Definition at line 33 of file `rsa_genkey.c`.

13.147.1.2 `#define KEY_SIZE 1024`

Definition at line 32 of file `rsa_genkey.c`.

13.147.2 Function Documentation

13.147.2.1 `int main (void)`

Definition at line 35 of file `rsa_genkey.c`.

References `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `EXPONENT`, `havege_init()`, `havege_rand()`, `KEY_SIZE`, `mpi_write_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_free()`, `rsa_gen_key()`, `rsa_init()`, and `RSA_PKCS_V15`.

13.148 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/rsa_sign.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

Functions

- [int main](#) ([int](#) argc, char *argv[])

13.148.1 Function Documentation

13.148.1.1 [int main](#) ([int](#) *argc*, char * *argv*[])

Definition at line 31 of file `rsa_sign.c`.

References `buf`, `rsa_context::D`, `rsa_context::DP`, `rsa_context::DQ`, `rsa_context::E`, `f`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `rsa_context::P`, `rsa_context::Q`, `rsa_context::QP`, `rsa_init()`, `rsa_pkcs1_sign()`, `RSA_PKCS_V15`, `RSA_PRIVATE`, `RSA_SHA1`, and `sha1_file()`.

13.149 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/pkey/rsa_verify.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/rsa.h"
#include "xyssl/sha1.h"
```

Functions

- `int main (int argc, char *argv[])`

13.149.1 Function Documentation

13.149.1.1 `int main (int argc, char * argv[])`

Definition at line 31 of file `rsa_verify.c`.

References `buf`, `rsa_context::E`, `f`, `rsa_context::len`, `mpi_msb()`, `mpi_read_file()`, `rsa_context::N`, `rsa_init()`, `rsa_pkcs1_verify()`, `RSA_PKCS_V15`, `RSA_PUBLIC`, `RSA_SHA1`, and `sha1_file()`.

13.150 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/ssl/ssl_client1.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/havege.h"
```

Defines

- `#define SERVER_PORT 443`
- `#define SERVER_NAME "xyssl.org"`
- `#define GET_REQUEST`
- `#define DEBUG_LEVEL 0`

Functions

- `void my_debug (void *ctx, int level, char *str)`
- `int main (void)`

13.150.1 Define Documentation

13.150.1.1 `#define DEBUG_LEVEL 0`

Definition at line 42 of file `ssl_client1.c`.

Referenced by `my_debug()`.

13.150.1.2 `#define GET_REQUEST`

Value:

```
"GET /hello/ HTTP/1.1\r\n" \
"Host: xyssl.org\r\n\r\n"
```

Definition at line 38 of file `ssl_client1.c`.

Referenced by `main()`.

13.150.1.3 `#define SERVER_NAME "xyssl.org"`

Definition at line 37 of file `ssl_client1.c`.

13.150.1.4 `#define SERVER_PORT 443`

Definition at line 32 of file `ssl_client1.c`.

13.150.2 Function Documentation

13.150.2.1 `int main (void)`

Definition at line 53 of file `ssl_client1.c`.

References `buf`, `GET_REQUEST`, `havege_init()`, `havege_rand()`, `my_debug()`, `net_close()`, `net_connect()`, `net_recv()`, `net_send()`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, `ssl_close_notify()`, `ssl_default_ciphers`, `ssl_free()`, `ssl_init()`, `SSL_IS_CLIENT`, `ssl_read()`, `ssl_set_authmode()`, `ssl_set_bio()`, `ssl_set_ciphers()`, `ssl_set_dbg()`, `ssl_set_endpoint()`, `ssl_set_rng()`, `ssl_set_session()`, `SSL_VERIFY_NONE`, `ssl_write()`, `XYSSL_ERR_NET_TRY_AGAIN`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

13.150.2.2 `void my_debug (void * ctx, int level, char * str)`

Definition at line 44 of file `ssl_client1.c`.

References `DEBUG_LEVEL`.

Referenced by `main()`, and `ssl_test()`.

13.151 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/ssl/ssl_client2.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/havege.h"
#include "xyssl/certs.h"
#include "xyssl/x509.h"
```

Defines

- `#define SERVER_PORT 443`
- `#define SERVER_NAME "xyssl.org"`
- `#define GET_REQUEST`
- `#define DEBUG_LEVEL 0`

Functions

- `void my_debug (void *ctx, int level, char *str)`
- `int main (void)`

13.151.1 Define Documentation

13.151.1.1 `#define DEBUG_LEVEL 0`

Definition at line 44 of file `ssl_client2.c`.

13.151.1.2 `#define GET_REQUEST`

Value:

```
"GET /hello/ HTTP/1.1\r\n" \
"Host: xyssl.org\r\n\r\n"
```

Definition at line 40 of file `ssl_client2.c`.

13.151.1.3 `#define SERVER_NAME "xyssl.org"`

Definition at line 39 of file `ssl_client2.c`.

13.151.1.4 `#define SERVER_PORT 443`

Definition at line 34 of file `ssl_client2.c`.

13.151.2 Function Documentation

13.151.2.1 `int main (void)`

Definition at line 55 of file `ssl_client2.c`.

References `BADCERT_CN_MISMATCH`, `BADCERT_EXPIRED`, `BADCERT_NOT_TRUSTED`, `BADCERT_REVOKED`, `buf`, `GET_REQUEST`, `havege_init()`, `havege_rand()`, `net_close()`, `net_connect()`, `net_recv()`, `net_send()`, `_ssl_context::peer_cert`, `rsa_free()`, `server_fd`, `SERVER_NAME`, `SERVER_PORT`, `ssl_close_notify()`, `ssl_default_ciphers`, `ssl_free()`, `ssl_get_cipher()`, `ssl_get_verify_result()`, `ssl_handshake()`, `ssl_init()`, `SSL_IS_CLIENT`, `ssl_read()`, `ssl_set_authmode()`, `ssl_set_bio()`, `ssl_set_ca_chain()`, `ssl_set_ciphers()`, `ssl_set_endpoint()`, `ssl_set_hostname()`, `ssl_set_own_cert()`, `ssl_set_rng()`, `ssl_set_session()`, `SSL_VERIFY_OPTIONAL`, `ssl_write()`, `test_cli_cert`, `test_cli_key`, `x509_free()`, `x509parse_cert_info()`, `x509parse_cert()`, `x509parse_key()`, `xyssl_ca_cert`, `XYSSL_ERR_NET_TRY_AGAIN`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

13.151.2.2 `void my_debug (void * ctx, int level, char * str)`

Definition at line 46 of file `ssl_client2.c`.

References `DEBUG_LEVEL`.

13.152 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/ssl/ssl_server.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include "xyssl/havege.h"
#include "xyssl/certs.h"
#include "xyssl/x509.h"
#include "xyssl/ssl.h"
#include "xyssl/net.h"
```

Defines

- #define [HTTP_RESPONSE](#)
- #define [DEBUG_LEVEL](#) 0

Functions

- void [my_debug](#) (void *ctx, [int](#) level, char *str)
- static [int](#) [my_get_session](#) ([ssl_context](#) *ssl)
- static [int](#) [my_set_session](#) ([ssl_context](#) *ssl)
- [int](#) [main](#) (void)

Variables

- char * [my_dhm_P](#)
- char * [my_dhm_G](#) = "4"
- [int](#) [my_ciphers](#) []
- [ssl_session](#) * [s_list_1st](#) = NULL
- [ssl_session](#) * [cur](#)
- [ssl_session](#) * [prv](#)

13.152.1 Define Documentation

13.152.1.1 #define [DEBUG_LEVEL](#) 0

Definition at line 75 of file `ssl_server.c`.

13.152.1.2 #define [HTTP_RESPONSE](#)

Value:

```
"HTTP/1.0 200 OK\r\nContent-Type: text/html\r\n\r\n" \
"<h2><p><center>Successful connection using: %s\r\n"
```

Definition at line 39 of file `ssl_server.c`.

Referenced by `http_ParseRequest()`, `main()`, and `mtp_http_Parse()`.

13.152.2 Function Documentation

13.152.2.1 `int main (void)`

Definition at line 160 of file `ssl_server.c`.

References `buf`, `client_fd`, `havege_init()`, `havege_rand()`, `HTTP_RESPONSE`, `my_ciphers`, `my_debug()`, `my_dhm_G`, `my_dhm_P`, `my_get_session()`, `my_set_session()`, `net_accept()`, `net_bind()`, `net_close()`, `net_recv()`, `net_send()`, `_ssl_session::next`, `_x509_cert::next`, `rsa_free()`, `ssl_close_notify()`, `ssl_free()`, `ssl_get_cipher()`, `ssl_handshake()`, `ssl_init()`, `SSL_IS_SERVER`, `ssl_read()`, `ssl_set_authmode()`, `ssl_set_bio()`, `ssl_set_ca_chain()`, `ssl_set_ciphers()`, `ssl_set_dbg()`, `ssl_set_dh_param()`, `ssl_set_endpoint()`, `ssl_set_own_cert()`, `ssl_set_rng()`, `ssl_set_scb()`, `ssl_set_session()`, `SSL_VERIFY_NONE`, `ssl_write()`, `test_ca_cert`, `test_srv_cert`, `test_srv_key`, `x509_free()`, `x509parse_cert()`, `x509parse_key()`, `XYSSL_ERR_NET_CONN_RESET`, `XYSSL_ERR_NET_TRY_AGAIN`, and `XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY`.

13.152.2.2 `void my_debug (void * ctx, int level, char * str)`

Definition at line 77 of file `ssl_server.c`.

References `DEBUG_LEVEL`.

13.152.2.3 `static int my_get_session (ssl_context * ssl) [static]`

Definition at line 93 of file `ssl_server.c`.

References `_ssl_session::cipher`, `_ssl_session::id`, `_ssl_session::length`, `_ssl_session::master`, `_ssl_session::next`, `_ssl_context::resume`, `_ssl_context::session`, `_ssl_session::start`, and `_ssl_context::timeout`.

Referenced by `main()`.

13.152.2.4 `static int my_set_session (ssl_context * ssl) [static]`

Definition at line 125 of file `ssl_server.c`.

References `_ssl_session::id`, `_ssl_session::length`, `_ssl_session::next`, `_ssl_context::session`, `_ssl_session::start`, and `_ssl_context::timeout`.

Referenced by `main()`.

13.152.3 Variable Documentation

13.152.3.1 `ssl_session* cur`

Definition at line 91 of file `ssl_server.c`.

Referenced by `http_GetToken()`, `http_ParseRequest()`, and `x509parse_verify()`.

13.152.3.2 `int my_ciphers[]`

Initial value:

```
{
    SSL_EDH_RSA_AES_256_SHA,
    SSL_EDH_RSA_DES_168_SHA,
    SSL_RSA_AES_256_SHA,
    SSL_RSA_AES_128_SHA,
    SSL_RSA_DES_168_SHA,
    SSL_RSA_RC4_128_SHA,
    SSL_RSA_RC4_128_MD5,
    0
}
```

Definition at line 63 of file ssl_server.c.

Referenced by main().

13.152.3.3 char* my_dhm_G = "4"

Definition at line 58 of file ssl_server.c.

Referenced by main().

13.152.3.4 char* my_dhm_P

Initial value:

```
"E4004C1F94182000103D883A448B3F80"
"2CE4B44A83301270002C20D0321CFD00"
"11CCEF784C26A400F43DFB901BCA7538"
"F2C6B176001CF5A0FD16D2C48B1D0C1C"
"F6AC8E1DA6BCC3B4E1F96B0564965300"
"FFA1D0B601EB2800F489AA512C4B248C"
"01F76949A60BB7F00A40B1EAB64BDD48"
"E8A700D60B7F1200FA8E77B0A979DABF"
```

Definition at line 48 of file ssl_server.c.

Referenced by main().

13.152.3.5 ssl_session * prv

Definition at line 91 of file ssl_server.c.

Referenced by aes_self_test(), and des_self_test().

13.152.3.6 ssl_session* s_list_1st = NULL

Definition at line 90 of file ssl_server.c.

13.153 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/benchmark.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include "xyssl/config.h"
#include "xyssl/md4.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include "xyssl/sha2.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
#include "xyssl/aes.h"
#include "xyssl/rsa.h"
#include "xyssl/timing.h"
```

Defines

- #define [BUFSIZE](#) 1024

Functions

- static [int myrand](#) (void *rng_state)
- [int main](#) (void)

Variables

- unsigned char [buf](#) [BUFSIZE]

13.153.1 Define Documentation

13.153.1.1 #define BUFSIZE 1024

Definition at line 41 of file benchmark.c.

Referenced by [main\(\)](#).

13.153.2 Function Documentation

13.153.2.1 int main (void)

Definition at line 53 of file benchmark.c.

References aes_crypt_cbc(), AES_ENCRYPT, aes_setkey_enc(), alarmed, arc4_crypt(), arc4_setup(), buf, BUFSIZE, des3_crypt_cbc(), des3_set3key_enc(), des_crypt_cbc(), DES_ENCRYPT, des_setkey_enc(), hardclock(), md4(), md5(), myrand(), rsa_free(), rsa_gen_key(), rsa_init(), RSA_PKCS_V15, rsa_private(), rsa_public(), set_alarm(), sha1(), and sha2().

13.153.2.2 static int myrand (void *rng_state) [static]

Definition at line 43 of file benchmark.c.

Referenced by main().

13.153.3 Variable Documentation

13.153.3.1 unsigned char buf[BUFSIZE]

Definition at line 51 of file benchmark.c.

Referenced by aes_self_test(), agent_xml_compose__create_row_nodes(), agent_xml_compose__data(), agent_xml_compose__task(), agent_xml_compose__tasks(), agent_xml_parse__fill_row_data(), agent_xml_parse__tasks(), arc4_self_test(), cmd_prompt_Thread(), des_self_test(), fipa_DateTime_Compose(), fipa_datetime_Parse(), fipa_envelope_Compose__date(), fipa_envelope_Compose__from(), fipa_envelope_HandleParams(), main(), MC_Initialize(), MC_LoadAgentFromFile(), MC_RetrieveAgentCode(), MC_SendAgentMigrationMessageFile(), md5_check(), md5_file(), md5_self_test(), message_InitializeFromAgent(), message_send_Thread(), message_xml_parse__message(), mtp_http_ComposeMessage(), mtp_http_CreateMessage(), mxml_fd_getc(), mxml_fd_putc(), mxmlLoadFd(), mxmlSaveFd(), net_htons(), rsa_pkcs1_decrypt(), rsa_pkcs1_verify(), sha1_check(), sha1_file(), sha1_self_test(), sha2_check(), sha2_file(), sha2_self_test(), sha4_file(), sha4_self_test(), ssl_flush_output(), ssl_parse_client_hello(), ssl_parse_finished(), ssl_parse_server_hello(), ssl_write_certificate_request(), ssl_write_client_hello(), ssl_write_server_hello(), udplisten_Thread(), x509parse_cert_info(), x509parse_crtfile(), x509parse_keyfile(), and xml_get_cdata().

13.154 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/selftest.c File Reference

```
#include <string.h>
#include <stdio.h>
#include "xyssl/config.h"
#include "xyssl/md2.h"
#include "xyssl/md4.h"
#include "xyssl/md5.h"
#include "xyssl/sha1.h"
#include "xyssl/sha2.h"
#include "xyssl/sha4.h"
#include "xyssl/arc4.h"
#include "xyssl/des.h"
#include "xyssl/aes.h"
#include "xyssl/base64.h"
#include "xyssl/bignum.h"
#include "xyssl/rsa.h"
#include "xyssl/x509.h"
```

Functions

- `int main (int argc, char *argv[])`

13.154.1 Function Documentation

13.154.1.1 `int main (int argc, char * argv[])`

Definition at line 44 of file selftest.c.

References `aes_self_test()`, `arc4_self_test()`, `base64_self_test()`, `des_self_test()`, `md2_self_test()`, `md4_self_test()`, `md5_self_test()`, `mpi_self_test()`, `rsa_self_test()`, `sha1_self_test()`, `sha2_self_test()`, `sha4_self_test()`, and `x509_self_test()`.

13.155 /home/dko/Projects/mobilec/trunk/src/security/xyssl-0.9/programs/test/ssl_test.c File Reference

```
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include "xyssl/net.h"
#include "xyssl/ssl.h"
#include "xyssl/havege.h"
#include "xyssl/timing.h"
#include "xyssl/certs.h"
```

Data Structures

- struct [options](#)

Defines

- #define [OPMODE_NONE](#) 0
- #define [OPMODE_CLIENT](#) 1
- #define [OPMODE_SERVER](#) 2
- #define [IOMODE_BLOCK](#) 0
- #define [IOMODE_NONBLOCK](#) 1
- #define [COMMAND_READ](#) 1
- #define [COMMAND_WRITE](#) 2
- #define [COMMAND_BOTH](#) 3
- #define [DFL_OPMODE](#) OPMODE_NONE
- #define [DFL_IOMODE](#) IOMODE_BLOCK
- #define [DFL_SERVER_NAME](#) "localhost"
- #define [DFL_SERVER_PORT](#) 4433
- #define [DFL_COMMAND](#) COMMAND_READ
- #define [DFL_BUFFER_SIZE](#) 1024
- #define [DFL_MAX_BYTES](#) 0
- #define [DFL_DEBUG_LEVEL](#) 0
- #define [DFL_CONN_TIMEOUT](#) 0
- #define [DFL_MAX_CONNECTIONS](#) 0
- #define [DFL_SESSION_REUSE](#) 1
- #define [DFL_SESSION_LIFETIME](#) 86400
- #define [DFL_FORCE_CIPHER](#) 0
- #define [USAGE](#)

Functions

- unsigned long [lcppm5](#) (unsigned long [int](#) *state)
- void [my_debug](#) (void *ctx, [int](#) level, char *str)
- static [int](#) [ssl_test](#) (struct [options](#) *opt)
- [int](#) [main](#) ([int](#) argc, char *argv[])

Variables

- char * [dhm_G](#) = "4"
- char * [dhm_P](#)
- int [server_fd](#) = -1

13.155.1 Define Documentation

13.155.1.1 #define COMMAND_BOTH 3

Definition at line 44 of file `ssl_test.c`.

Referenced by `main()`.

13.155.1.2 #define COMMAND_READ 1

Definition at line 42 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

13.155.1.3 #define COMMAND_WRITE 2

Definition at line 43 of file `ssl_test.c`.

Referenced by `main()`, and `ssl_test()`.

13.155.1.4 #define DFL_BUFFER_SIZE 1024

Definition at line 51 of file `ssl_test.c`.

Referenced by `main()`.

13.155.1.5 #define DFL_COMMAND COMMAND_READ

Definition at line 50 of file `ssl_test.c`.

Referenced by `main()`.

13.155.1.6 #define DFL_CONN_TIMEOUT 0

Definition at line 54 of file `ssl_test.c`.

Referenced by `main()`.

13.155.1.7 #define DFL_DEBUG_LEVEL 0

Definition at line 53 of file `ssl_test.c`.

Referenced by `main()`.

13.155.1.8 #define DFL_FORCE_CIPHER 0

Definition at line 58 of file ssl_test.c.

Referenced by main(), and ssl_test().

13.155.1.9 #define DFL_IOMODE IOMODE_BLOCK

Definition at line 47 of file ssl_test.c.

Referenced by main().

13.155.1.10 #define DFL_MAX_BYTES 0

Definition at line 52 of file ssl_test.c.

Referenced by main().

13.155.1.11 #define DFL_MAX_CONNECTIONS 0

Definition at line 55 of file ssl_test.c.

Referenced by main().

13.155.1.12 #define DFL_OPMODE OPMODE_NONE

Definition at line 46 of file ssl_test.c.

Referenced by main().

13.155.1.13 #define DFL_SERVER_NAME "localhost"

Definition at line 48 of file ssl_test.c.

Referenced by main().

13.155.1.14 #define DFL_SERVER_PORT 4433

Definition at line 49 of file ssl_test.c.

Referenced by main().

13.155.1.15 #define DFL_SESSION_LIFETIME 86400

Definition at line 57 of file ssl_test.c.

Referenced by main().

13.155.1.16 #define DFL_SESSION_REUSE 1

Definition at line 56 of file ssl_test.c.

Referenced by main().

13.155.1.17 #define IOMODE_BLOCK 0

Definition at line 39 of file ssl_test.c.

Referenced by main().

13.155.1.18 #define IOMODE_NONBLOCK 1

Definition at line 40 of file ssl_test.c.

Referenced by main(), and ssl_test().

13.155.1.19 #define OPMODE_CLIENT 1

Definition at line 36 of file ssl_test.c.

Referenced by main(), and ssl_test().

13.155.1.20 #define OPMODE_NONE 0

Definition at line 35 of file ssl_test.c.

13.155.1.21 #define OPMODE_SERVER 2

Definition at line 37 of file ssl_test.c.

Referenced by main(), and ssl_test().

13.155.1.22 #define USAGE**Value:**

```
"\n usage: ssl_test opmode=<> command=<>...\n"
"\n acceptable parameters:\n"
"  opmode=client/server          default: <none>\n"
"  iomode=block/nonblock        default: block\n"
"  server_name=%s               default: localhost\n"
"  server_port=%d               default: 4433\n"
"  command=read/write/both      default: read\n"
"  buffer_size=%d (bytes)       default: 1024\n"
"  max_bytes=%d (bytes)         default: 0 (no limit)\n"
"  debug_level=%d               default: 0 (disabled)\n"
"  conn_timeout=%d (ms)         default: 0 (no timeout)\n"
"  max_connections=%d           default: 0 (no limit)\n"
"  session_reuse=on/off         default: on (enabled)\n"
"  session_lifetime=%d (s)      default: 86400\n"
"  force_cipher=<name>          default: all enabled\n"
" acceptable cipher names:\n"
"  SSL_RSA_RC4_128_MD5          SSL_RSA_RC4_128_SHA\n"
"  SSL_RSA_DES_168_SHA          SSL_EDH_RSA_DES_168_SHA\n"
"  SSL_RSA_AES_128_SHA          SSL_EDH_RSA_AES_256_SHA\n"
"  SSL_RSA_AES_256_SHA\n"
```

Definition at line 372 of file ssl_test.c.

13.155.2 Function Documentation

13.155.2.1 unsigned long int lcppm5 (unsigned long int * *state*)

Definition at line 96 of file ssl_test.c.

Referenced by ssl_test().

13.155.2.2 int main (int *argc*, char * *argv*[])

Definition at line 394 of file ssl_test.c.

References options::buffer_size, options::command, COMMAND_BOTH, COMMAND_READ, COMMAND_WRITE, options::conn_timeout, options::debug_level, DFL_BUFFER_SIZE, DFL_COMMAND, DFL_CONN_TIMEOUT, DFL_DEBUG_LEVEL, DFL_FORCE_CIPHER, DFL_IOMODE, DFL_MAX_BYTES, DFL_MAX_CONNECTIONS, DFL_OPMODE, DFL_SERVER_NAME, DFL_SERVER_PORT, DFL_SESSION_LIFETIME, DFL_SESSION_REUSE, options::force_cipher, options::iomode, IOMODE_BLOCK, IOMODE_NONBLOCK, options::max_bytes, options::max_connections, options::opmode, OPMODE_CLIENT, OPMODE_SERVER, options::server_name, options::server_port, options::session_lifetime, options::session_reuse, SSL_EDH_RSA_AES_256_SHA, SSL_EDH_RSA_DES_168_SHA, SSL_RSA_AES_128_SHA, SSL_RSA_AES_256_SHA, SSL_RSA_DES_168_SHA, SSL_RSA_RC4_128_MD5, SSL_RSA_RC4_128_SHA, ssl_test(), and USAGE.

13.155.2.3 void my_debug (void * *ctx*, int *level*, char * *str*)

Definition at line 113 of file ssl_test.c.

13.155.2.4 static int ssl_test (struct options * *opt*) [static]

Definition at line 122 of file ssl_test.c.

References options::buffer_size, client_fd, options::command, COMMAND_READ, COMMAND_WRITE, options::conn_timeout, DFL_FORCE_CIPHER, dhm_G, dhm_P, options::force_cipher, get_timer(), havege_init(), havege_rand(), options::iomode, IOMODE_NONBLOCK, lcppm5(), options::max_bytes, my_debug(), net_accept(), net_bind(), net_close(), net_connect(), net_recv(), net_send(), net_set_nonblock(), x509_cert::next, options::opmode, OPMODE_CLIENT, OPMODE_SERVER, rsa_free(), server_fd, options::server_name, options::server_port, options::session_lifetime, options::session_reuse, ssl_close_notify(), ssl_default_ciphers, ssl_free(), ssl_init(), SSL_IS_CLIENT, SSL_IS_SERVER, ssl_read(), ssl_set_authmode(), ssl_set_bio(), ssl_set_ca_chain(), ssl_set_ciphers(), ssl_set_dbg(), ssl_set_dh_param(), ssl_set_endpoint(), ssl_set_own_cert(), ssl_set_rng(), ssl_set_session(), SSL_VERIFY_NONE, ssl_write(), test_ca_cert, test_srv_cert, test_srv_key, x509_free(), x509parse_cert(), x509parse_key(), XYSSL_ERR_NET_CONN_RESET, XYSSL_ERR_NET_TRY_AGAIN, and XYSSL_ERR_SSL_PEER_CLOSE_NOTIFY.

Referenced by main().

13.155.3 Variable Documentation

13.155.3.1 char* dhm_G = "4"

Definition at line 63 of file ssl_test.c.

Referenced by ssl_test().

13.155.3.2 char* dhm_P

Initial value:

```
"E4004C1F94182000103D883A448B3F802CE4B44A83301270002C20D0321CFD00"  
"11CCEF784C26A400F43DFB901BCA7538F2C6B176001CF5A0FD16D2C48B1D0C1C"  
"F6AC8E1DA6BCC3B4E1F96B0564965300FFA1D0B601EB2800F489AA512C4B248C"  
"01F76949A60BB7F00A40B1EAB64BDD48E8A700D60B7F1200FA8E77B0A979DABF"
```

Definition at line 64 of file ssl_test.c.

Referenced by ssl_test().

13.155.3.3 int server_fd = -1

Definition at line 70 of file ssl_test.c.

13.156 /home/dko/Projects/mobilec/trunk/src/util/mc_genkey.c File Reference

```
#include "../security/interface.h"
```

Functions

- void [print_usage](#) ()
- int [main](#) (int argc, char *argv[])

13.156.1 Function Documentation

13.156.1.1 int main (int *argc*, char * *argv*[])

Definition at line 19 of file mc_genkey.c.

References [generate_RSA_keys_ciphertext\(\)](#), [generate_RSA_keys_plaintext\(\)](#), and [print_usage\(\)](#).

13.156.1.2 void print_usage ()

Definition at line 11 of file mc_genkey.c.

Referenced by [main\(\)](#).

13.157 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net File Reference

Data Structures

- struct [EmbeddedCh::ChInfo_t](#)

Namespaces

- namespace [EmbeddedCh](#)

Enumerations

- enum [EmbeddedCh::ChType_t](#) {
[EmbeddedCh::CH_UNDEFINETYPE](#), [EmbeddedCh::CH_CHARTYPE](#) = 10, [EmbeddedCh::CH_-
UCHARTYPE](#), [EmbeddedCh::CH_SHORTTYPE](#),
[EmbeddedCh::CH_USHORTTYPE](#), [EmbeddedCh::CH_INTTYPE](#), [EmbeddedCh::CH_-
UINTTYPE](#), [EmbeddedCh::CH_LLINTTYPE](#),
[EmbeddedCh::CH_ULLINTTYPE](#), [EmbeddedCh::CH_FLOATTYPE](#), [EmbeddedCh::CH_-
DOUBLETTYPE](#), [EmbeddedCh::CH_LDOUBLETTYPE](#),
[EmbeddedCh::CH_COMPLEXTYPE](#), [EmbeddedCh::CH_LCOMPLEXTYPE](#),
[EmbeddedCh::CH_STRINGTYPE](#), [EmbeddedCh::CH_FILETYPE](#),
[EmbeddedCh::CH_VOIDTYPE](#), [EmbeddedCh::CH_PROCTYPE](#), [EmbeddedCh::CH_-
STRUCTTYPE](#), [EmbeddedCh::CH_CLASSTYPE](#),
[EmbeddedCh::CH_UNIONTYPE](#), [EmbeddedCh::CH_ENUMTYPE](#), [EmbeddedCh::CH_-
CARRYTYPE](#) = 80, [EmbeddedCh::CH_CARRYPTRTYPE](#),
[EmbeddedCh::CH_CARRYVLATTYPE](#), [EmbeddedCh::CH_CHARRAYTYPE](#),
[EmbeddedCh::CH_CHARRAYPTRTYPE](#), [EmbeddedCh::CH_CHARRAYVLATTYPE](#),
[EmbeddedCh::CH_NULLTYPE](#) = 100, [EmbeddedCh::CH_VOIDPTRTYPE](#), [EmbeddedCh::CH_-
CHARPTRTYPE](#), [EmbeddedCh::CH_UCHARPTRTYPE](#),
[EmbeddedCh::CH_SHORTPTRTYPE](#), [EmbeddedCh::CH_USHORTPTRTYPE](#),
[EmbeddedCh::CH_INTPTRTYPE](#), [EmbeddedCh::CH_UINTPTRTYPE](#),
[EmbeddedCh::CH_LLINTPTRTYPE](#), [EmbeddedCh::CH_ULLINTPTRTYPE](#), [EmbeddedCh::CH_-
FLOATPTRTYPE](#), [EmbeddedCh::CH_DOUBLEPTRTYPE](#),
[EmbeddedCh::CH_LDOUBLEPTRTYPE](#), [EmbeddedCh::CH_COMPLEXPTRTYPE](#),
[EmbeddedCh::CH_LCOMPLEXPTRTYPE](#), [EmbeddedCh::CH_STRINGPTRTYPE](#),
[EmbeddedCh::CH_PROCPTRTYPE](#), [EmbeddedCh::CH_FILEPTRTYPE](#), [EmbeddedCh::CH_-
STRUCTPTRTYPE](#), [EmbeddedCh::CH_CLASSPTRTYPE](#),
[EmbeddedCh::CH_UNIONPTRTYPE](#), [EmbeddedCh::CH_ENUMPTRTYPE](#), [EmbeddedCh::CH_-
VOIDPTR2TYPE](#) = 200, [EmbeddedCh::CH_CHARPTR2TYPE](#),
[EmbeddedCh::CH_UCHARPTR2TYPE](#), [EmbeddedCh::CH_SHORTPTR2TYPE](#),
[EmbeddedCh::CH_USHORTPTR2TYPE](#), [EmbeddedCh::CH_INTPTR2TYPE](#),
[EmbeddedCh::CH_UINTPTR2TYPE](#), [EmbeddedCh::CH_LLINTPTR2TYPE](#), [EmbeddedCh::CH_-
ULLINTPTR2TYPE](#), [EmbeddedCh::CH_FLOATPTR2TYPE](#),
[EmbeddedCh::CH_DOUBLEPTR2TYPE](#), [EmbeddedCh::CH_LDOUBLEPTR2TYPE](#),
[EmbeddedCh::CH_COMPLEXPTR2TYPE](#), [EmbeddedCh::CH_LCOMPLEXPTR2TYPE](#),

```
EmbeddedCh::CH_STRINGPTR2TYPE, EmbeddedCh::CH_FILEPTR2TYPE,  
EmbeddedCh::CH_STRUCTPTR2TYPE, EmbeddedCh::CH_CLASSPTR2TYPE,  
EmbeddedCh::CH_UNIONPTR2TYPE, EmbeddedCh::CH_ENUMPTR2TYPE }  
• enum EmbeddedCh::ChRetVal { EmbeddedCh::CH_OK = 0, EmbeddedCh::CH_ERROR = -1,  
EmbeddedCh::CH_ABORT = 1 }
```

13.158 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net File Reference

Data Structures

- class [EmbeddedCh::ChInterp](#)

Namespaces

- namespace [EmbeddedCh](#)

13.159 /home/dko/Projects/mobilec/trunk/src/
win32/EmbeddedCh.Net/EmbeddedCh.Net/ChUserDefinedTag.cs File

Reference

13.159⁹⁴⁷ /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net/ChUserDefinedTag.cs File

File Reference

Data Structures

- class [EmbeddedCh::ChUserDefinedTag](#)

Namespaces

- namespace [EmbeddedCh](#)

13.160 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net File Reference

Data Structures

- class [EmbeddedCh::ChVaList](#)

Namespaces

- namespace [EmbeddedCh](#)

File Reference

Data Structures

- struct [EmbeddedCh::ChOptions_t](#)
- struct [EmbeddedCh::ChBlock_t](#)
- struct [EmbeddedCh::ChUserDefinedInfo_t](#)
- struct [EmbeddedCh::ChMemInfo_t](#)

Namespaces

- namespace [EmbeddedCh](#)

Enumerations

- enum [EmbeddedCh::ChFuncType_t](#) {
[EmbeddedCh::CH_NOTFUNCTYPE](#), [EmbeddedCh::CH_FUNCTYPE](#), [EmbeddedCh::CH_FUNC-
 FUNCPROTOTYPE](#), [EmbeddedCh::CH_FUNCPTRTYPE](#),
[EmbeddedCh::CH_FUNCMEMBERTYPE](#), [EmbeddedCh::CH_FUNCCONSTYPE](#),
[EmbeddedCh::CH_FUNCDESTTYPE](#) }
- enum [EmbeddedCh::ChVarType_t](#) { [EmbeddedCh::CH_NOTVARTYPE](#), [EmbeddedCh::CH_-
 GLOBALVARTYPE](#), [EmbeddedCh::CH_LOCALVARTYPE](#) }
- enum [EmbeddedCh::ChShellType](#) { [EmbeddedCh::CH_REGULARARCH](#) = 0, [EmbeddedCh::CH_-
 SAFECH](#) = 1 }
Ch shell type.
- enum [EmbeddedCh::ChFileDescriptor](#) { [EmbeddedCh::STDIN_FILENO](#) = 0,
[EmbeddedCh::STDOUT_FILENO](#) = 1, [EmbeddedCh::STDERR_FILENO](#) = 2 }
- enum [EmbeddedCh::ChCallbackMask](#) {
[EmbeddedCh::CH_MASKNONE](#) = 0X0000, [EmbeddedCh::CH_MASKCALL](#) = 0X0001,
[EmbeddedCh::CH_MASKRET](#) = 0X0002, [EmbeddedCh::CH_MASKBLOCK](#) = 0X0004,
[EmbeddedCh::CH_MASKEND](#) = 0X0008, [EmbeddedCh::CH_MASKLINE](#) = 0X0010,
[EmbeddedCh::CH_MASKCOUNT](#) = 0X0020, [EmbeddedCh::CH_MASKABORT](#) = 0X0040
 }

13.162 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/EmbeddedCh.Net File Reference

Variables

- using System. [Reflection](#)

13.162.1 Variable Documentation

13.162.1.1 using System. Reflection

Definition at line 1 of file AssemblyInfo.cs.

13.163 /home/dko/Projects/mobilec/trunk/src/
win32/EmbeddedCh.Net/Program1/Properties/AssemblyInfo.cs File

Reference

13.163 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/Program1

File Reference

Variables

- using System. [Reflection](#)

13.163.1 Variable Documentation

13.163.1.1 using System. Reflection

Definition at line 1 of file AssemblyInfo.cs.

13.164 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties File Reference

Variables

- using System. [Reflection](#)

13.164.1 Variable Documentation

13.164.1.1 using System. Reflection

Definition at line 1 of file AssemblyInfo.cs.

13.165 /home/dko/Projects/mobilec/trunk/src/win32/EmbeddedCh.Net/Program1 File Reference

Data Structures

- class [Program1::Program](#)

Namespaces

- namespace [Program1](#)

13.166 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAclM File Reference

Data Structures

- class [LibMC::MCAclMessage](#)
Encapsulates ACL messages in the Mobile-C library.

Namespaces

- namespace [LibMC](#)
Namespace for the .NET wrapper for Mobile-C.

13.166.1 Detailed Description

Defines the MCAclMessage object and its member functions.

Definition in file [MCAclMessage.cs](#).

13.167 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgen File Reference

Data Structures

- class [LibMC::MCAgency](#)
Wrapper class for [MCAgency_t](#) structure.
- class [LibMC::InvalidAgencyException](#)
Exception class for use with null agency pointers.

Namespaces

- namespace [LibMC](#)
Namespace for the .NET wrapper for Mobile-C.

13.167.1 Detailed Description

Defines the MCAgency object and its member functions.

Definition in file [MCAgency.cs](#).

13.168 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCAgent File Reference

Data Structures

- class [LibMC::MCAgent](#)
Wrapper class for MCAgent_t structure.
- class [LibMC::InvalidAgentException](#)
Exception class for use with null agent pointers.

Namespaces

- namespace [LibMC](#)
Namespace for the .NET wrapper for Mobile-C.

13.168.1 Detailed Description

Defines the MCAgent object and its member functions.

Definition in file [MCAgent.cs](#).

13.169 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/MCExports.cs File Reference

Data Structures

- class [LibMC::MCAgency](#)
Wrapper class for [MCAgency_t](#) structure.
- struct [LibMC::MCAgency::MCAgency_t](#)
- struct [LibMC::MCAgency::MCAgencyOptions_t](#)
- struct [LibMC::MCAgency::ChOptions_t](#)
ChOptions structures.

Namespaces

- namespace [LibMC](#)
Namespace for the .NET wrapper for Mobile-C.

13.169.1 Detailed Description

Imports functions, structs, and enums from the Mobile-C library.

Definition in file [MCExports.cs](#).

13.170 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Properties File Reference

Data Structures

- class [LibMC::Properties::Settings](#)

Namespaces

- namespace [LibMC::Properties](#)
Namespace for the .NET wrapper properties class.

13.171 /home/dko/Projects/mobilec/trunk/src/win32/LibMC.net/LibMC/Settings.cs File Reference

Data Structures

- class [LibMC::Properties::Settings](#)

Namespaces

- namespace [LibMC::Properties](#)
Namespace for the .NET wrapper properties class.

13.172 /home/dko/Projects/mobilec/trunk/src/winconfig.h File Reference

Defines

- #define [PACKAGE_STRING](#) "MobileC V1.10.0"
- #define [PACKAGE_VERSION](#) "1.10.0"
- #define [strtok_r](#)(buf, delim, save_ptr) strtok(buf, delim)

13.172.1 Define Documentation

13.172.1.1 #define PACKAGE_STRING "MobileC V1.10.0"

Definition at line 4 of file winconfig.h.

Referenced by mtp_http_InitializeFromConnection().

13.172.1.2 #define PACKAGE_VERSION "1.10.0"

Definition at line 5 of file winconfig.h.

Referenced by mtp_http_ComposeMessage(), mtp_http_CreateMessage(), and udplisten_Thread().

13.172.1.3 #define strtok_r(buf, delim, save_ptr) strtok(buf, delim)

Definition at line 6 of file winconfig.h.

Referenced by agent_xml_parse__fill_row_data(), message_InitializeFromAgent(), message_send_Thread(), and message_xml_parse__message().

13.173 /home/dko/Projects/mobilec/trunk/src/xml_compose.c File Reference

```
#include "config.h"
#include <mxml.h>
#include "include/agent.h"
#include "include/xml_compose.h"
#include "include/xml_helper.h"
```

Functions

- [mxml_node_t * agent_xml_compose \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__gaf_message \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__message \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__mobile_agent \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__agent_data \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__name \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__owner \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__home \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__wg_code \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__tasks \(agent_p agent\)](#)
- [mxml_node_t * agent_xml_compose__task \(agent_p agent, int index\)](#)
- [mxml_node_t * agent_xml_compose__data \(agent_p agent, int index, interpreter_variable_data_t *interp_variable\)](#)
- [mxml_node_t * agent_xml_compose__agent_code \(agent_p agent, int index\)](#)
- [mxml_node_t * agent_xml_compose__row \(interpreter_variable_data_t *interp_variable, int index\)](#)
- [mxml_node_t * agent_xml_compose__create_row_nodes \(void *data, int index, int *extent, ChType_t type, int dim, int extent_index\)](#)

13.173.1 Function Documentation

13.173.1.1 [mxml_node_t* agent_xml_compose \(agent_p agent\)](#)

Definition at line 46 of file `xml_compose.c`.

References `agent_xml_compose__gaf_message()`, `MXML_ADD_AFTER`, `MXML_ADD_TO_PARENT`, `MXML_NO_CALLBACK`, `mxmlAdd()`, `mxmlLoadString()`, and `node`.

Referenced by `message_InitializeFromAgent()`.

13.173.1.2 [mxml_node_t* agent_xml_compose__agent_code \(agent_p agent, int index\)](#)

Definition at line 521 of file `xml_compose.c`.

References `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_s::datastate`, `MXML_NO_PARENT`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, and `xml_new_cdata()`.

Referenced by `agent_xml_compose__tasks()`.

13.173.1.3 mxml_node_t* agent_xml_compose__agent_data (agent_p agent)

Definition at line 150 of file xml_compose.c.

References agent_xml_compose__home(), agent_xml_compose__name(), agent_xml_compose__owner(), agent_xml_compose__tasks(), agent_xml_compose__wg_code(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose__mobile_agent().

13.173.1.4 mxml_node_t* agent_xml_compose__create_row_nodes (void * data, int index, int * extent, CType_t type, int dim, int extent_index)

Definition at line 572 of file xml_compose.c.

References agent_xml_compose__create_row_nodes(), buf, CH_DATATYPE_SIZE, CH_DATATYPE_VALUE_STRING, MXML_ADD_AFTER, MXML_ADD_TO_PARENT, MXML_NO_PARENT, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), mxmlNewText(), node, and size.

Referenced by agent_xml_compose__create_row_nodes(), and agent_xml_compose__row().

13.173.1.5 mxml_node_t* agent_xml_compose__data (agent_p agent, int index, interpreter_variable_data_t * interp_variable)

Definition at line 447 of file xml_compose.c.

References agent_xml_compose__row(), interpreter_variable_data_s::array_dim, buf, CH_DATATYPE_STRING, CH_DATATYPE_VALUE_STRING, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, MXML_ADD_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), interpreter_variable_data_s::name, and node.

Referenced by agent_xml_compose__task().

13.173.1.6 mxml_node_t* agent_xml_compose__gaf_message (agent_p agent)

Definition at line 66 of file xml_compose.c.

References agent_xml_compose__message(), MXML_ADD_AFTER, mxmlAdd(), mxmlNewElement(), and node.

Referenced by agent_xml_compose().

13.173.1.7 mxml_node_t* agent_xml_compose__home (agent_p agent)

Definition at line 253 of file xml_compose.c.

References agent_s::home, mxmlNewElement(), mxmlNewText(), and node.

Referenced by agent_xml_compose__agent_data().

13.173.1.8 mxml_node_t* agent_xml_compose__message (agent_p agent)

Definition at line 85 of file xml_compose.c.

References `agent_s::agent_type`, `agent_xml_compose__mobile_agent()`, `MC_LOCAL_AGENT`, `MC_REMOTE_AGENT`, `MC_RETURN_AGENT`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__gaf_message()`.

13.173.1.9 `mxml_node_t* agent_xml_compose__mobile_agent (agent_p agent)`

Definition at line 129 of file `xml_compose.c`.

References `agent_xml_compose__agent_data()`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlNewElement()`, and `node`.

Referenced by `agent_xml_compose__message()`.

13.173.1.10 `mxml_node_t* agent_xml_compose__name (agent_p agent)`

Definition at line 221 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `agent_s::name`, and `node`.

Referenced by `agent_xml_compose__agent_data()`.

13.173.1.11 `mxml_node_t* agent_xml_compose__owner (agent_p agent)`

Definition at line 237 of file `xml_compose.c`.

References `mxmlNewElement()`, `mxmlNewText()`, `node`, and `agent_s::owner`.

Referenced by `agent_xml_compose__agent_data()`.

13.173.1.12 `mxml_node_t* agent_xml_compose__row (interpreter_variable_data_t * interp_variable, int index)`

Definition at line 549 of file `xml_compose.c`.

References `agent_xml_compose__create_row_nodes()`, `interpreter_variable_data_s::array_dim`, `interpreter_variable_data_s::array_extent`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, and `node`.

Referenced by `agent_xml_compose__data()`.

13.173.1.13 `mxml_node_t* agent_xml_compose__task (agent_p agent, int index)`

Definition at line 342 of file `xml_compose.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_compose__data()`, `buf`, `agent_task_s::code_id`, `agent_s::datastate`, `MXML_ADD_AFTER`, `mxmlAdd()`, `mxmlElementSetAttr()`, `mxmlNewElement()`, `node`, `agent_task_s::persistent`, `agent_datastate_s::persistent`, `agent_task_s::server_name`, `agent_datastate_s::tasks`, and `agent_task_s::var_name`.

Referenced by `agent_xml_compose__tasks()`.

13.173.1.14 mxml_node_t* agent_xml_compose__tasks (agent_p *agent*)

Definition at line 285 of file xml_compose.c.

References agent_xml_compose__agent_code(), agent_xml_compose__task(), buf, agent_s::datastate, MXML_ADD_AFTER, mxmlAdd(), mxmlElementSetAttr(), mxmlNewElement(), node, agent_datastate_s::number_of_tasks, and agent_datastate_s::task_progress.

Referenced by agent_xml_compose__agent_data().

13.173.1.15 mxml_node_t* agent_xml_compose__wg_code (agent_p *agent*)

Definition at line 269 of file xml_compose.c.

References mxmlNewElement(), mxmlNewText(), node, and agent_s::wg_code.

Referenced by agent_xml_compose__agent_data().

13.174 /home/dko/Projects/mobilec/trunk/src/xml_helper.c File Reference

```
#include "config.h"
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <mxml.h>
#include "include/xml_helper.h"
#include "include/macros.h"
```

Functions

- [mxml_node_t * xml_find_sibling](#) (const [mxml_node_t *node](#), const char *sibling_name)
- char * [xml_get_cdata](#) (const [mxml_node_t *node](#))
- [mxml_node_t * xml_get_child](#) (const [mxml_node_t *node](#), const char *child_name, [int](#) descend)
- [mxml_node_t * xml_get_deep_child](#) (const [mxml_node_t *parent](#), const char **child_path)
- [mxml_node_t * xml_get_next_element](#) (const [mxml_node_t *node](#))
- char * [xml_get_text](#) (const [mxml_node_t *node](#))
- const char * [xml_get_element_name](#) (const [mxml_node_t *node](#))
- [mxml_node_t * xml_new_cdata](#) ([mxml_node_t *parent](#), const char *text)
- const char * [whitespace_cb](#) ([mxml_node_t *node](#), [int](#) where)

13.174.1 Function Documentation

13.174.1.1 const char* whitespace_cb (mxml_node_t * node, int where)

Definition at line 254 of file xml_helper.c.

References `MXML_WS_AFTER_CLOSE`, and `MXML_WS_BEFORE_OPEN`.

Referenced by `main()`.

13.174.1.2 mxml_node_t* xml_find_sibling (const mxml_node_t * node, const char * sibling_name)

Definition at line 54 of file xml_helper.c.

References `MXML_NO_DESCEND`, `mxmlFindElement()`, `mxml_node_s::parent`, and `xml_get_element_name()`.

13.174.1.3 char* xml_get_cdata (const mxml_node_t * node)

Definition at line 76 of file xml_helper.c.

References `buf`, `CHECK_NULL`, `MXML_ELEMENT`, `mxml_node_s::type`, and `xml_get_element_name()`.

Referenced by `xml_get_text()`.

13.174.1.4 mxml_node_t* xml_get_child (const mxml_node_t * node, const char * child_name, int descend)

Definition at line 109 of file xml_helper.c.

References mxmlFindElement().

Referenced by agent_return_xml_parse(), agent_xml_parse__agent_data(), agent_xml_parse__data(), agent_xml_parse__mobile_agent(), message_xml_parse(), message_xml_parse__message(), and xml_get_deep_child().

13.174.1.5 mxml_node_t* xml_get_deep_child (const mxml_node_t * parent, const char ** child_path)

Definition at line 128 of file xml_helper.c.

References MXML_NO_DESCEND, node, and xml_get_child().

13.174.1.6 const char* xml_get_element_name (const mxml_node_t * node)

Definition at line 222 of file xml_helper.c.

References mxml_value_u::element, MXML_ELEMENT, mxml_value_s::name, mxml_node_s::type, and mxml_node_s::value.

Referenced by agent_xml_parse__data(), agent_xml_parse__mobile_agent(), agent_xml_parse__row(), message_xml_parse(), xml_find_sibling(), and xml_get_cdata().

13.174.1.7 mxml_node_t* xml_get_next_element (const mxml_node_t * node)

Definition at line 142 of file xml_helper.c.

References MXML_ELEMENT, mxml_node_s::next, and mxml_node_s::type.

13.174.1.8 char* xml_get_text (const mxml_node_t * node)

Definition at line 160 of file xml_helper.c.

References CHECK_NULL, mxml_node_s::child, mxml_value_u::element, MXML_ELEMENT, MXML_TEXT, mxml_value_s::name, mxml_node_s::next, mxml_text_s::string, mxml_value_u::text, mxml_node_s::type, mxml_node_s::value, and xml_get_cdata().

Referenced by agent_xml_parse__agent_code(), agent_xml_parse__home(), agent_xml_parse__name(), agent_xml_parse__owner(), agent_xml_parse__sender(), and agent_xml_parse__wg_code().

13.174.1.9 mxml_node_t* xml_new_cdata (mxml_node_t * parent, const char * text)

Definition at line 235 of file xml_helper.c.

References CHECK_NULL, mxmlNewElement(), and node.

Referenced by agent_xml_compose__agent_code().

13.175 /home/dko/Projects/mobilec/trunk/src/xml_parser.c File Reference

```
#include <mxml.h>
#include <string.h>
#include <stdlib.h>
#include "config.h"
#include "include/interpreter_variable_data.h"
#include "include/message.h"
#include "include/xml_parser.h"
#include "include/xml_helper.h"
```

Defines

- `#define _XOPEN_SOURCE 600`

Functions

- `error_code_t agent_xml_parse (agent_p agent)`
- `error_code_t agent_xml_parse__mobile_agent (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__agent_data (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__name (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__owner (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__home (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__sender (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__wg_code (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__tasks (agent_p agent, xml_parser_p xml_parser)`
- `error_code_t agent_xml_parse__task (agent_p agent, xml_parser_p xml_parser, int index)`
- `error_code_t agent_xml_parse__data (agent_p agent, xml_parser_p xml_parser, int index)`
- `error_code_t agent_xml_parse__row (interpreter_variable_data_t *interp_variable, xml_parser_p xml_parser, int index)`
- `void agent_xml_parse__fill_row_data (void *data, ChType_t type, int *extent, const mxml_node_t *node, int *index)`
- `error_code_t agent_xml_parse__agent_code (agent_p agent, int index, xml_parser_p xml_parser)`
- `error_code_t agent_return_xml_parse (agent_p agent)`
- `error_code_t message_xml_parse (message_p message)`
- `error_code_t message_xml_parse__message (message_p message, xml_parser_p xml_parser)`

13.175.1 Define Documentation

13.175.1.1 `#define _XOPEN_SOURCE 600`

Definition at line 38 of file `xml_parser.c`.

13.175.2 Function Documentation

13.175.2.1 `error_code_t agent_return_xml_parse (agent_p agent)`

Definition at line 913 of file `xml_parser.c`.

References `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__tasks()`, `agent_s::datastate`, `MC_SUCCESS`, `xml_get_child()`, and `agent_datastate_s::xml_root`.

13.175.2.2 `error_code_t agent_xml_parse (agent_p agent)`

Definition at line 52 of file `xml_parser.c`.

References `agent_xml_parse__mobile_agent()`, `agent_s::datastate`, `MC_SUCCESS`, and `agent_datastate_s::xml_agent_root`.

Referenced by `agent_initialize()`.

13.175.2.3 `error_code_t agent_xml_parse__agent_code (agent_p agent, int index, xml_parser_p xml_parser)`

Definition at line 872 of file `xml_parser.c`.

References `agent_datastate_s::agent_code`, `agent_datastate_s::agent_code_ids`, `agent_datastate_s::agent_codes`, `agent_task_s::code_id`, `agent_s::datastate`, `MC_SUCCESS`, `mxmlElementGetAttr()`, `agent_datastate_s::number_of_tasks`, `agent_datastate_s::task_progress`, `agent_datastate_s::tasks`, and `xml_get_text()`.

Referenced by `agent_xml_parse__tasks()`.

13.175.2.4 `error_code_t agent_xml_parse__agent_data (agent_p agent, xml_parser_p xml_parser)`

Definition at line 93 of file `xml_parser.c`.

References `agent_xml_parse__home()`, `agent_xml_parse__name()`, `agent_xml_parse__owner()`, `agent_xml_parse__sender()`, `agent_xml_parse__tasks()`, `agent_xml_parse__wg_code()`, `MC_ERR_PARSE`, `MC_SUCCESS`, and `xml_get_child()`.

Referenced by `agent_xml_parse__mobile_agent()`.

13.175.2.5 `error_code_t agent_xml_parse__data (agent_p agent, xml_parser_p xml_parser, int index)`

Definition at line 538 of file `xml_parser.c`.

References `agent_task_s::agent_return_data`, `agent_task_s::agent_variable_list`, `agent_xml_parse__row()`, `interpreter_variable_data_s::array_dim`, `CH_DATATYPE_SIZE`, `CH_DATATYPE_STR_TO_VAL`, `CH_STRING_DATATYPE`, `interpreter_variable_data_s::data`, `interpreter_variable_data_s::data_type`, `agent_s::datastate`, `interpreter_variable_data_New()`, `MC_ERR_PARSE`, `MC_SUCCESS`, `mxmlElementGetAttr()`, `interpreter_variable_data_s::name`, `mxml_node_s::parent`, `agent_task_s::persistent`, `agent_datastate_s::tasks`, `xml_get_child()`, and `xml_get_element_name()`.

Referenced by `agent_xml_parse__task()`.

13.175.2.6 void agent_xml_parse__fill_row_data (void * *data*, ChType_t *type*, int * *extent*, const mxml_node_t * *node*, int * *index*)

Definition at line 751 of file xml_parser.c.

References agent_xml_parse__fill_row_data(), buf, CH_DATATYPE_SIZE, mxml_node_s::child, MXML_DESCEND_FIRST, MXML_ELEMENT, MXML_TEXT, mxmlFindElement(), mxml_text_s::string, strtok_r, mxml_value_u::text, mxml_node_s::type, and mxml_node_s::value.

Referenced by agent_xml_parse__fill_row_data(), and agent_xml_parse__row().

13.175.2.7 error_code_t agent_xml_parse__home (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 219 of file xml_parser.c.

References CHECK_NULL, agent_s::home, MC_SUCCESS, and xml_get_text().

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.175.2.8 error_code_t agent_xml_parse__mobile_agent (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 65 of file xml_parser.c.

References agent_xml_parse__agent_data(), MC_ERR_PARSE, xml_get_child(), and xml_get_element_name().

Referenced by agent_xml_parse().

13.175.2.9 error_code_t agent_xml_parse__name (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 166 of file xml_parser.c.

References CHECK_NULL, MC_ERR_PARSE, MC_SUCCESS, agent_s::name, and xml_get_text().

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.175.2.10 error_code_t agent_xml_parse__owner (agent_p *agent*, xml_parser_p *xml_parser*)

Definition at line 192 of file xml_parser.c.

References CHECK_NULL, MC_SUCCESS, agent_s::owner, and xml_get_text().

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.175.2.11 error_code_t agent_xml_parse__row (interpreter_variable_data_t * *interp_variable*, xml_parser_p *xml_parser*, int *index*)

Definition at line 684 of file xml_parser.c.

References agent_xml_parse__fill_row_data(), interpreter_variable_data_s::array_dim, interpreter_variable_data_s::array_extent, CH_DATATYPE_SIZE, interpreter_variable_data_s::data, interpreter_variable_data_s::data_type, MC_SUCCESS, and xml_get_element_name().

Referenced by agent_xml_parse__data().

13.175.2.12 error_code_t agent_xml_parse__sender (agent_p agent, xml_parser_p xml_parser)

Definition at line 245 of file xml_parser.c.

References CHECK_NULL, MC_SUCCESS, agent_s::sender, and xml_get_text().

Referenced by agent_xml_parse__agent_data().

13.175.2.13 error_code_t agent_xml_parse__task (agent_p agent, xml_parser_p xml_parser, int index)

Definition at line 442 of file xml_parser.c.

References agent_xml_parse__data(), CHECK_NULL, agent_task_s::code_id, agent_s::datastate, MC_ERR_PARSE, MC_SUCCESS, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent_task_s::persistent, agent_task_s::server_name, agent_datastate_s::tasks, and agent_task_s::var_name.

Referenced by agent_xml_parse__tasks().

13.175.2.14 error_code_t agent_xml_parse__tasks (agent_p agent, xml_parser_p xml_parser)

Definition at line 304 of file xml_parser.c.

References agent_datastate_s::agent_code, agent_datastate_s::agent_code_ids, agent_datastate_s::agent_codes, agent_task_New(), agent_xml_parse__agent_code(), agent_xml_parse__task(), buf, agent_s::datastate, MC_ERR_PARSE, MXML_DESCEND, MXML_DESCEND_FIRST, MXML_NO_DESCEND, mxmlElementGetAttr(), mxmlFindElement(), agent_datastate_s::number_of_tasks, agent_datastate_s::task_progress, and agent_datastate_s::tasks.

Referenced by agent_return_xml_parse(), and agent_xml_parse__agent_data().

13.175.2.15 error_code_t agent_xml_parse__wg_code (agent_p agent, xml_parser_p xml_parser)

Definition at line 273 of file xml_parser.c.

References MC_SUCCESS, agent_s::wg_code, and xml_get_text().

Referenced by agent_xml_parse__agent_data().

13.175.2.16 error_code_t message_xml_parse (message_p message)

Definition at line 948 of file xml_parser.c.

References MC_ERR_PARSE, message_xml_parse__message(), MXML_DESCEND, MXML_NO_DESCEND, mxmlFindElement(), xml_get_child(), xml_get_element_name(), and message_s::xml_root.

Referenced by acc_connection_Thread(), message_InitializeFromConnection(), and message_InitializeFromString().

13.175.2.17 error_code_t message_xml_parse__message (message_p message, xml_parser_p xml_parser)

Definition at line 1003 of file xml_parser.c.

References message_s::addr, buf, CHECK_NULL, ENCRYPTED_DATA, ENCRYPTION_INITIALIZE, FIPA_ACL, message_s::from_address, MC_ERR_PARSE, MC_SUCCESS, message_s::message_type, MOBILE_AGENT, mxmlElementGetAttr(), port, REQUEST_ENCRYPTION_INITIALIZE, RETURN_MSG, strtok_r, xml_get_child(), and message_s::xml_payload.

Referenced by message_xml_parse().

Chapter 14

Example Documentation

14.1 cs2ch/cs2chfunc.c

C library example that allows agents to call CLI functions.

14.2 LibMCCConsole/Program.cs

Basic Mobile-C console demo program

14.3 LibMCCppEx/LibMCCppEx.cpp

Demonstrates using LibMC.NET from a VC++ program.

14.4 LibMCFipaTest/Program.cs

Mobile-C FIPA ACL message demo program.

14.5 LibMCGui/Form1.cs

Basic Mobile-C Windows Forms demo program

14.6 LibMCInterop/Program.cs

Demonstrates interfacing with a .dl file and calling agent functions.

14.7 LibMCMiscTest/Program.cs

Demonstrates miscellaneous Mobile-C functions.

14.8 LibMCVbEx/Form1.vb

Demonstrates using LibMC.NET from a VB program.