



HS SMTP Library v2.0.0 User Manual

Revision: 1.4
Date: 21 October 2009

HS SMTP LIBRARY V2.0.0 USER MANUAL	1
REVISION: 1.4	1
DATE: 21 OCTOBER 2009	1
1 INTRODUCTION	2
2 HS SMTP API	3
2.1 HSMTPINIT	3
2.2 HSMTPDESTROY	3
2.3 HSMTPTICK	3
2.4 HSMTPSENDMAIL	4
2.5 HSMTPABORTMAIL	6
2.6 HSMTPCONTINUE	6
3 HS SMTP TO USER EVENT CALLBACK AND EVENTS	7
3.1 EVENT CALLBACK PROTOTYPE	7
3.2 EVENT CODES	7

1 Introduction

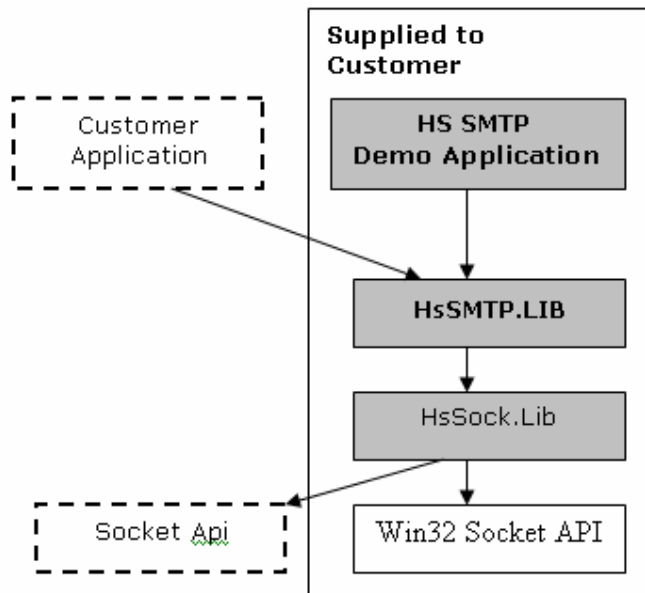
HS SMTP is a software library in C (supplied with full source code) which implements the client side of Simple Mail Transfer Protocol (SMTP) over TCP socket layer according to RFC 821. Support for transfer of basic message header and text is provided.

From version 1.1, HS SMTP now supports ESMTP extension for LOGIN Authentication using Base64 encoding and message sending to multiple recipients from address list.

From version 1.2 HS SMTP supports sending binary file attachments using MIME version 1.0 base64 encoding.

From version 2.0.0 HS SMTP supports SSL / TLS and secure SMTP over SSL / TLS (RFC 3207)

The library is a stand-alone module which links directly to customer application:



HS SMTP Library incorporates the necessary state machine, transparency procedures, and server response processing required to provide for a simple and robust SMTP client implementation

2 HS SMTP API

2.1 HsSmtplnit

int HsSmtplnit(void);

DESCRIPTION

This function initialises HS SMTP Library. It must be called at initialisation, before any other functions are called. Calling HsSmtplnit twice will return an error. You can call HsSmtplDestroy first to deinitialise HS SMTP Library and then call HsSmtplnit again

PARAMETERS

Type	Name	Description
None	None	None

RETURNS

- *HS SMTP_RC_OK – success, HS SMTP Library successfully initialised*
- *HS SMTP_RC_ALRINIT – HS SMTP Library is already initialised*
- *HS SMTP_RC_SOCKINIT_FAIL – General socket interface module initialisation problem*
- *HS SMTP_RC_SOCKINIT2_FAIL – Socket layer could not get local hostname*

2.2 HsSmtplDestroy

int HsSmtplDestroy(void);

DESCRIPTION

This function de-initialises HS SMTP Library, releases all used resources and cleans up Socket Interface Layer.

PARAMETERS

Type	Name	Description
None	None	None

RETURNS

- *HS SMTP_RC_OK – success, HS SMTP Library successfully de-initialised*
- *HS SMTP_RC_NOTINIT – cannot destroy, HS SMTP Library not yet initialised*

2.3 HsSmtplTick

int HsSmtplTick (void);

DESCRIPTION

This function must be called from the user application periodically and as often as possible. This function drives internal operation of the socket layer (reading events from TCP sockets) and pacing internal SMTP timers.

PARAMETERS

Type	Name	Description
None	None	None

RETURNS

- *HS SMTP_RC_OK – success*
- *HS SMTP_RC_NOTINIT – HS SMTP Library not initialised*

2.4 HsSmtplibSendMail

Int HsSmtplibSendMail(hs_smtplib_mail_t *m, long *session_handle)

DESCRIPTION

This function sends an email text message to remote SMTP server according to RFC821. Fill out hs_smtplib_mail_t structure members and set pointer to storage for SMTP session handle prior to calling the function. Various structure fields are explained below. The function is non-blocking and complete asynchronously. The completion of mail transfer operation is reported via event callback function. The pointer to callback function is also specified in hs_smtplib_mail_t structure.

PARAMETERS

Type	Name	Description
------	------	-------------

hs_smtp_mail_t	*m	<p>Pointer to parameter structure as follows:</p> <table border="1" data-bbox="737 241 1406 1759"> <thead> <tr> <th>Field name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>srv_name</td> <td>Pointer to SMTP server name Max. length limited to 80 bytes (Note 1)</td> </tr> <tr> <td>srv_ip</td> <td>Pointer to SMTP IP address string in the format "n.n.n.n" (Note 1)</td> </tr> <tr> <td>srv_port</td> <td>Decimal SMTP server port number. (usually 25)</td> </tr> <tr> <td>send_addr</td> <td>Sender email address string Max. length limited to 80 bytes</td> </tr> <tr> <td>recv_addr</td> <td>Destination email address string Max. length limited to 80 bytes</td> </tr> <tr> <td>subj</td> <td>Message subject Max. length limited to 80 bytes</td> </tr> <tr> <td>cc</td> <td>Cc email address Max. length limited to 80 bytes</td> </tr> <tr> <td>msgp</td> <td>Pointer to message text buffer</td> </tr> <tr> <td>msglen</td> <td>message text buffer length</td> </tr> <tr> <td>do_auth</td> <td>Integer. Set to 1 if SMTP server requires authentication. LOGIN method authentication of ESMTP shall be performed</td> </tr> <tr> <td>usrname</td> <td>Pointer to username string for authentication, ignored if do_auth is 0</td> </tr> <tr> <td>password</td> <td>Pointer to password string for authentication, ignored if do_auth is 0</td> </tr> <tr> <td>send_file</td> <td>1=send file attachment, 0=no file attachment</td> </tr> <tr> <td>filename</td> <td>filename of the file to send, valid only if sen_file = 1</td> </tr> <tr> <td>callback</td> <td>Pointer to callback function in user code to receive completion and error event notifications from HS SMTP Library. (See section 3 for more details)</td> </tr> <tr> <td>session_ref</td> <td>User data associated with this session. This parameter is passed to the callback function in used code without modification</td> </tr> <tr> <td>use_latin1</td> <td>Integer flag which switches on the use of iso-8859-1 encoding. for characters from 0x7f to 0xff use escape sequence ==<hex code> Set this flag to TRUE when sending file attachments. Set to false when sending mail without attachments.</td> </tr> <tr> <td>useSSL</td> <td>1= Run SMTP over SSL / TLS; 0=standard SMTP, not secure If you set this to 1, set srv_port to secure SMTP protocol server port, for example 578 for GMAIL (smtp.gmail.com)</td> </tr> <tr> <td>check_peer_certificate</td> <td>If useSSL=1, set this to 1 to check peer certificate during SSL negotiation</td> </tr> <tr> <td>logging_enabled</td> <td>Enabled logging of debug events using HS_SMTP_USR_EV_LOGININFO event</td> </tr> </tbody> </table> <p>Note 1: Either server name or server IP address string can be supplied to the function. If srv_ip is non NULL, srv_ip string shall be used to connect to SMTP server. If srv_ip is null, srv_name string shall be used to resolve server IP address first before connecting to it.</p>	Field name	Description	srv_name	Pointer to SMTP server name Max. length limited to 80 bytes (Note 1)	srv_ip	Pointer to SMTP IP address string in the format "n.n.n.n" (Note 1)	srv_port	Decimal SMTP server port number. (usually 25)	send_addr	Sender email address string Max. length limited to 80 bytes	recv_addr	Destination email address string Max. length limited to 80 bytes	subj	Message subject Max. length limited to 80 bytes	cc	Cc email address Max. length limited to 80 bytes	msgp	Pointer to message text buffer	msglen	message text buffer length	do_auth	Integer. Set to 1 if SMTP server requires authentication. LOGIN method authentication of ESMTP shall be performed	usrname	Pointer to username string for authentication, ignored if do_auth is 0	password	Pointer to password string for authentication, ignored if do_auth is 0	send_file	1=send file attachment, 0=no file attachment	filename	filename of the file to send, valid only if sen_file = 1	callback	Pointer to callback function in user code to receive completion and error event notifications from HS SMTP Library. (See section 3 for more details)	session_ref	User data associated with this session. This parameter is passed to the callback function in used code without modification	use_latin1	Integer flag which switches on the use of iso-8859-1 encoding. for characters from 0x7f to 0xff use escape sequence ==<hex code> Set this flag to TRUE when sending file attachments. Set to false when sending mail without attachments.	useSSL	1= Run SMTP over SSL / TLS; 0=standard SMTP, not secure If you set this to 1, set srv_port to secure SMTP protocol server port, for example 578 for GMAIL (smtp.gmail.com)	check_peer_certificate	If useSSL=1, set this to 1 to check peer certificate during SSL negotiation	logging_enabled	Enabled logging of debug events using HS_SMTP_USR_EV_LOGININFO event
Field name	Description																																											
srv_name	Pointer to SMTP server name Max. length limited to 80 bytes (Note 1)																																											
srv_ip	Pointer to SMTP IP address string in the format "n.n.n.n" (Note 1)																																											
srv_port	Decimal SMTP server port number. (usually 25)																																											
send_addr	Sender email address string Max. length limited to 80 bytes																																											
recv_addr	Destination email address string Max. length limited to 80 bytes																																											
subj	Message subject Max. length limited to 80 bytes																																											
cc	Cc email address Max. length limited to 80 bytes																																											
msgp	Pointer to message text buffer																																											
msglen	message text buffer length																																											
do_auth	Integer. Set to 1 if SMTP server requires authentication. LOGIN method authentication of ESMTP shall be performed																																											
usrname	Pointer to username string for authentication, ignored if do_auth is 0																																											
password	Pointer to password string for authentication, ignored if do_auth is 0																																											
send_file	1=send file attachment, 0=no file attachment																																											
filename	filename of the file to send, valid only if sen_file = 1																																											
callback	Pointer to callback function in user code to receive completion and error event notifications from HS SMTP Library. (See section 3 for more details)																																											
session_ref	User data associated with this session. This parameter is passed to the callback function in used code without modification																																											
use_latin1	Integer flag which switches on the use of iso-8859-1 encoding. for characters from 0x7f to 0xff use escape sequence ==<hex code> Set this flag to TRUE when sending file attachments. Set to false when sending mail without attachments.																																											
useSSL	1= Run SMTP over SSL / TLS; 0=standard SMTP, not secure If you set this to 1, set srv_port to secure SMTP protocol server port, for example 578 for GMAIL (smtp.gmail.com)																																											
check_peer_certificate	If useSSL=1, set this to 1 to check peer certificate during SSL negotiation																																											
logging_enabled	Enabled logging of debug events using HS_SMTP_USR_EV_LOGININFO event																																											

long	*session_handle	HS SMTP fills in the variable pointed to by this parameter with SMTP session reference. Use this parameter to close a session with HsSmtplibAbortMail if required.
------	-----------------	--

R E T U R N S

- *HS SMTP_RC_OK – Success, mail transfer initiated*
- *HS SMTP_RC_NOTINIT – HS SMTP Library not initialised*
- *HS SMTP_RC_INV_PAR – Invalid parameters specified*
- *HS SMTP_RC_NO_CTX – No free contexts (Maximum 5 concurrent sessions supported)*
- *HS SMTP_RC_NONAME – Socket layer cannot resolve SMTP server name*
- *HS SMTP_RC_TCPCONNFAIL – Socket layer failed to connect to SMTP server*

2.5 HsSmtplibAbortMail

Int HsSmtplibAbortMail(long session_handle)

D E S C R I P T I O N

This function is used to abort current mail transfer already in progress. The function returns immediately, but the clean mail session termination completes asynchronously and the result is reported via event callback function.

P A R A M E T E R S

Type	Name	Description
Long	Session_handle	SMTP session handle (returned with HsSmtplibSendMail)

R E T U R N S

- *HS SMTP_RC_OK – success*
- *HS SMTP_RC_NOTINIT – HS SMTP Library not initialised*
- *HS SMTP_RC_INV_PAR – Invalid parameters specified*
- *HS SMTP_RC_NOT_OPEN – mail session not open*

2.6 HsSmtplibContinue

Int HsSmtplibContinue(long session_handle)

D E S C R I P T I O N

This function is only applicable if SMTP over SSL /TLS has been enabled when calling HsSmtplibSendMail. This function is used to continue to connect to SMTP server after server SSL certificate validation failed and the user wants to ignore server certificate error and continue with mail session. If SSL server certificate validation fails, HSSMTP shall call event callback function with event HS SMTP_USR_EV_BADCERT. The user may then call this function if server certificate is not important.

P A R A M E T E R S

Type	Name	Description
Long	Session_handle	SMTP session handle (returned with HsSmtplibSendMail)

R E T U R N S

- *HS SMTP_RC_OK – success*
- *HS SMTP_RC_NOTINIT – HS SMTP Library not initialised*
- *HS SMTP_RC_INV_PAR – Invalid parameters specified*
- *HS SMTP_RC_NOT_OPEN – mail session not open*

3 HS SMTP to User Event Callback and Events

3.1 Event Callback Prototype

```
typedef int smtp_callback_t(void *session_ref, int ev, long arg1, long arg2);
```

Parameter	Description
*session_ref	User data, passed unchanged from the call to HsSmtplibSendMail
ev	Event code (see event code table below)
Arg1	Parameter 1 (dependant on event code)
Arg2	Parameter 2 (dependant on event code)

3.2 Event Codes

Event	Arg1	Arg2	Description
HS SMTP_USR_EV_DONE	0	0	Mail transfer complete, mail has been successfully sent to SMTP server and mail session is terminated
HS SMTP_USR_EV_CLOSED	0	0	TCP session closed by remote peer
HS SOCK_EV_CONN_FAILED	0	0	TCP connect attempt failed
HS SMTP_USR_EV_SRVERR	Unsigned char *buffer pointer to server reply string	Int length length of server reply string	Mail session closed because SMTP server returned error. The error string returned by the server is passed in parameters arg1 and arg2
HS SMTP_USR_EV_TIMEDOUT	Unsigned char *error Null terminated error string	0	Mail session closed because current protocol operation timed out. Arg1 parameter has pointer to error string with additional information about the timeout condition.
HS SMTP_USR_EV_PROGRESS	Integer number from 0 to 100 representing percentage of transmitted message text	0	Progress indication, parameter 1 has a number from 1 to 100, representing percentage of message text sent so far.
HS SMTP_USR_EV_PROGRESS_F	Integer number from 0 to 100 representing percentage of file attachment transmitted so far	0	Progress indication, parameter 1 has a number from 1 to 100, representing percentage of of file attachment transmitted so far.

HS SMTP_USR_EV_GETNEXT	Pointer to hs_smtp_mail_t message structure	0	When a message has been successfully transmitted to server, the callback function is called with this primitive. If the user application wishes to send the same message to another recipient, it should set the new *recv_addr in the hs_smtp_mail_t structure and return TRUE. If the user application returns FALSE, HS SMTP will exit the mail session
HS SMTP_USR_EV_BADCERT	Pointer to error string	Length of error string	<p>SSL peer certificate verification is on and certificate received failed validation.</p> <p>User must either disconnect (HsSmtAbortMail) or call continue (HsSmtContinue) to ignore this error</p> <p>Until HsSmtContinue is called HS SMTP remains waiting for user decision.</p>
HS SMTP_USR_EV_LOGINFO	Pointer to debug event string	Length of debug event string	If debug event logging is enabled via loggin_enabled member of hs_smtp_mail_t structure, this event carries a debug event string to be printed in application event log