

List of Fixed Bugs in SIP Toolkit v3.0.5.1

The following issues have been fixed or added in version 3.0.5.1 since version 3.0.4.0. Each item provides a classification, fix description, list of files modified in fixing the problem, and the scope of the fix (the party interested in this fix).

No.	Classification	Bug Description	Files Changed	Scope
1	BUG	When the Stack is configured to work with manual behavior and the call-leg reaches the RVSIP_CALL_LEG_STATE_CANCELLING state, responding to the INVITE with 487 from a different thread and directly from the state-changed callback failed, and the response was never sent since it was destructed from the original thread. This was fixed.	TransactionState.c	Multithreaded applications
2	FEATURE	A new callback was added to the Transport layer: RvSipTransportConnectionParserResultEv(). This callback indicates the parse status of a message received over TCP.	RvSipTransportTypes.h, TransportCallbacks.c, TransportCallbacks.h, TransportMsgBuilder.c	Everyone
3	BUG	RTP bug fix: problem in the rtpH261Unpack(), rtpH263Unpack() and rtpH263aUnpack() functions. These functions sent the wrong pointer to the LI_ConvertHeader2Host() function. This was fixed.	rtp/payload.c	RTP users
4	FEATURE	Enables the application to encode other headers as separate headers. Added under the ENCODE_OTHER_HEADER_SEPARATELY compilation flag.	RvSipMsg.c	Everyone
5	BUG	The call-leg object automatically rejected an initial INVITE with Cseq=0. This was fixed and now 0 is accepted.	CallLegObject.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
6	BUG	Contacts bug fix: when a provisional response is received, the contact is taken from it as it should be. If the final response after this is 3xx, 401, or 407, and the same call-leg is used for sending a new INVITE, the contact of the INVITE was the one that was taken from the 1xx, and not the original contact. This was fixed and now the original contact is taken.	CallLegMsgEv.c, CallLegObject.c, CallLegObject.h	Everyone
7	RESOURCES	Resource leak: when an out-of-context message is sent, the Stack encodes this message and then tries to create the connection. If the connection failed to be created, the encoded message page was never freed. This was fixed.	TransactionTransport.c	Applications that use the out-of-context mechanism
8	BUG	Timer roll over problem: the SIP Stack handles the time in milliseconds, stored in 32 bits. There is a situation of rollover, when the number of milliseconds is greater than 0xffffffff. In this case, the timer overlapped. The Stack maintains a list with all the timers that were set. The top of the list is the nearest timer. When there is a rollover, the Stack needs to know that a value of 0x00000001 is greater than 0xffffffff0 and that 0xffffffff0 should expire first. This ability was added in this version.	TIMER_API.c	Everyone
9	BUG	Subscription problem: when a subscription is terminated, the hidden call-leg that it uses remains in the hash until the terminated event of the subscription is out of the queue. Only then does the call-leg terminate as well, and is removed from the hash. The problem was that after the subscription terminated, and before the call-leg was removed from the hash, a retransmission of this subscription arrived, created a new transaction, found this call-leg in the hash and used it. To avoid this, we now immediately remove the call-leg from the hash when terminating the subscription.	SubsObject.c, _SipCallLeg.c, _SipCallLeg.h, CallLegSubs.c, CallLegSubs.h, _SipCallLegMgr.c	Applications that use the Subscription module

No.	Classification	Bug Description	Files Changed	Scope
10	BUG	SDP parser fix: the SDP parser crashed when a bad syntax field included a string longer than 120 bytes. This was fixed.	rvsdpprsaux.c	Everyone
11	BUG	When a reliable provisional response is received, the Transaction Manager creates a new PRACK transaction and supplies its handle to the call-leg object. If this PRACK was terminated due to an error, the call-leg was not notified and did not remove this transaction from its list, and therefore the call-leg never terminated. This was fixed.	TransactionControl.c	Everyone
12	BUG	SDP parser fix: enabling s=-	rvsdpprs.c,rvsdpprs.h	Everyone
13	RESOURCES	If there was an error while constructing the Transport layer--not all resources were freed on destruction. This was fixed.	TransportMgrObject.c	Everyone
14	CRASH	A server transaction can send more than one provisional and final response using the RvSipTransactionResponseMsg() function. If this function was called while the transaction was in the Terminated state, the Stack crashed. This was fixed and now the Stack does not allow calling this function in the Terminated state.	RvSipTransaction.c	Proxy applications
15	BUG	The parser failed to handle a bad syntax message with extra CRLF at the end of the header. This was fixed and we now skip this extra CRLF.	ParserProcess.c	Everyone
16	BUG	A call to the UnregisterClass function was missing for the LI_SERVICE_CLASS. This function call was added.	windows\LI_API.c	Windows applications
17	BUG	When a call-leg received 1xx with a route list and then 3xx, 401, or 407, the route list of the 1xx was used for the new INVITE. Now this route list is destructed when the Reject is received.	CallLegTranscEv.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
18	BUG	When working with TCP, if the application called the ByeReject() function and then the Terminate() function, the Reject might not get a chance to be sent. Therefore, the Bye transaction is now detached from the call-leg after calling this function.	CallLegSession.c	Application that register to the ByeStateChange callback.
19	BUG	When a call-leg terminates, it terminates all its transactions. Some of the transaction may still not know that this call-leg is their owner. The call-leg will never get the terminated event for these transactions and will never terminate. The fix checks that the transaction owner is indeed the call-leg. If it is not, the call-leg will not wait for the terminated state and will immediately remove the transaction from its list. This bug can happens only on multithreaded applications.	CallLegObject.c	Multithreaded applications
20	BUG	When the application called the RvSipAuthenticatorVerifyCredentials() function with the Authorization header that is placed on the application pool, this pool was not used and the general pool was used by mistake. This caused an exception in the log.	_SipAuthenticator.c	Everyone
21	BUG	An SRV query was applied to find the port event if the protocol was specified in the message. This is not according to RFC 3263. Now the SRV is applied only if no protocol is specified.	TransportDNS.c	Everyone
22	BUG	If a response to an INVITE request was received on a different connection than the one on which the INVITE was sent, the transaction tried to use this connection for the ACK. This behavior was changed and now the transaction does not use server connections for requests.	_SipTransactionMgr.c, TransactionObject.c, TransactionObject.h, TransactionTransport.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
23	BUG	The host name that was used for sending the DNS query and the host name returned from the query were compared case-sensitively. This was fixed and they are now compared case-insensitively.	core\WindowsLI_DNS_API.c	Windows users
24	BUG	When responding with 401/407 to UNSUBSCRIBE, the subscription returned to the ACTIVE state instead of returning to the previous state (pending/active...). This was fixed.	SubsNotify.c, SubsObject.c	
25	COMPILATION	When compiling the SDP Stack, there were 6 warning in the compilation. These warnings were fixed.	rvsdpprs.c,rvsdpprs.h	Everyone
26	BUG	The user part of the SIP URL did not allow the pound symbol (#). This was fixed.	ParserExpress.h, ParserExpress2.h	Everyone
27	BUG	If a SIP URI included a replaces header in the headers parameter, the replaces fields were handled case-sensitively. Now they are handled case-insensitively, as they should be.	rpool.c,rpool.h, RvSipReferToHeader.c, RvSipReferToHeader.c	Applications that implement the replaces feature
28	BUG	When the Stack received an incoming INVITE within another incoming INVITE with the same CSeq, the second INVITE was rejected properly, but when a 2xx response was sent on the first INVITE, the response did not contain the Record-Route list. This was fixed.	CallLegTranscEv.c	Everyone
29	BUG	When the BYE is handled manually, the Bye transaction was not detached from the call-leg after calling the Accept() function. This was fixed.	CallLegSession.c	Manual Bye users
30	FEATURE	A termination reason was added to the call-leg. Now when the call-leg terminates, the terminated state has a more indicative reason.	CallLegObject.c, CallLegObject.h, CallLegTranscEv.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
31	RESOURCES	Memory leak. A buffer with the size of 9 was allocated and the '\0' was set in the 10th element. This was fixed.	_SipAuthenticator.c	Everyone
32	BUG	The event header was set to SUBSCRIBE and NOTIFY response messages by mistake. This was fixed.	SubsNotify.c, SubsObject.c	Applications that use the Subscription module
33	BUG	There were situations where the server connection timeout timer was set again before it was released This was fixed.	TransportConnection.c	Everyone
34	BUG	When detaching from the transaction in the MSG_SEND_FAILURE state, the transaction did not terminate as it should have. This was fixed.	TransactionObject.c	Applications that use the Enhanced DNS feature
35	BUG	The Stack did not enable bad syntax start- line encoding. This was fixed and now the bad syntax string is taken.	RvSipMsg.c	Everyone
36	RESOURCES	When using the RvSipAuthenticatorSetProxyAuthInfo() function, setting the authentication information more than once caused a memory leak. This was fixed.	RvSipAuthenticator.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
37	COMPILATION	In order to support the Lint compiler, several minor changes were made in several files. (Changes such as casting.)	CallLegObject.c, core\common\rvtls.c, core\Windows\LI_API.c, core\Windows\LI_DNS_API.c, RvSipReplacesHeader.c, sdp\sdpparser\rvsdpmmedia.c, sdp\sdpparser\rvsdpmmsg.c, sdp\sdpparser\rvsdpobjs.c, sdp\sdpparser\rvsdpobjsh.h, sdp\sdpparser\rvsdpprsaux.c, sdp\sdpparser\rvsdpsymb.c, TransactionMgrObject.c, TransactionObject.c, RvSipTransport.c, TransportConnection.c	Users that compile with the Lint compiler
38	RESOURCES	A memory leak did not enable warm restart of the Stack on the VxWorks operating system. This was fixed.	unix/LOCK_API.c, unix/LOCK_API.h, unix/THREADS.c, unix/THREADS_API.h, RvSipStack.c, StackInternal.h	VxWorks users that want to do warm restart.
39	BUG	When the Stack was configured to work with Enhanced DNS and a 503 was received in the CANCELLING state, the transaction moved to the MSG_SEND_FAILURE state instead of handling the response as a regular reject. This was fixed.	TransactionState.c	Applications that compile with the Enhanced DNS feature.
40	BUG	According to the SIP standard, if a connection is broken before the response was sent, a new connection needs to be established according to the Via header of the request. If the Via does not contain a port, the default port should be used for the specific transport. The bug was that 5060 was used for TLS instead of 5061. This was fixed.	TransportDNS.c	Applications that use TLS

No.	Classification	Bug Description	Files Changed	Scope
41	BUG	An ACK was not sent when the call was terminated from the redirected state and the transport was TCP. The fix was to detach the Invite transaction from the call-leg so that it would not be terminated before the ACK is sent.	CallLegTranscEv.c	Everyone
42	BUG	The pNegotiationSessionTimer pointer was not initialized to NULL on call-leg initialization. This was fixed.	CallLegObject.c	Everyone

STANDARD	SIP Definition corrected
BUG	Bug fix in the code or the test application
CRASH	Crash occurred because of this bug
RESOURCES	Resource of the Stack leaked
PACKAGE	Bug was in the way the package was archived or supplied
COMPILATION	Compilation problems.
FEATURE	Added feature to the version that wasn't included previously