

## List of Fixed Bugs in SIP Toolkit v3.1.1.30

The following issues have been fixed or added in version 3.1.1.30 since version 3.1.1.24. 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
24	<b>BUG</b>	When a single SIP message that was received over TCP divided into several packets in a way that a single packet contained one header only, the accumulation process ended with an error. This was fixed.	TransportMsgBuilder.c	Everyone
23	<b>BUG</b>	A TCP message can be sent in several parts. There was a situation in which the first part of the message was sent and immediately after that, the message owner was terminated. When the owner was terminated, it also freed its messages and therefore the second part of the message was never sent. This caused a parse error in the remote party that received only a partial message. This was fixed and now the full message will be sent even though the owner was terminated.	TransportConnection.c TransportTCP.c	Everyone
22	<b>BUG</b>	Using TLS without implementing the RvSipTransportConnectionTlsSequenceStartedEv() callback was impossible. This was fixed and it is now possible to run without implementing this callback.	TransportCallbacks.c	Applications that use TLS
21	<b>BUG</b>	When doing a handshake on TLS, a timer was set to prevent the situation in which the handshake would linger forever. This timer was too short, and in some cases, the timer expired before the handshake was completed, resulting in a connection shutdown. This timer was extended to 5 minutes.	RvSipStack.c	Applications that use TLS

No.	Classification	Bug Description	Files Changed	Scope
20	<b>BUG</b>	When sending a final response, the Stack first called the RvSipCallLegTranscStateChangedEv() callback and then updated the Session-Timer parameters, which are related to the call-leg. The order of these two function calls was switched so that an application that created an UPDATE transaction will be able to retrieve the new Session-Timer parameters within the RvSipCallLegTranscStateChangedEv() callback.	CallLegTranscEv.c	Applications that use the Session Timer.
19	<b>RESOURCES</b>	When resetting a call-leg within the High Availability module (in case of error during the storing/restoring process), the call-leg authentication headers were not freed properly and caused a memory leak. This was fixed.	CallLegHighAvailability.c	Applications that implement the High Availability feature.
18	<b>BUG</b>	Race condition in timeout evaluation led to the malfunctioning (stopping) of the timer queue mechanism. This was fixed and now race condition is prevented by putting all timeout-related evaluations in a critical section.	rvselect.c rvselectinternal.h	Multithreaded applications
17	<b>RESOURCES</b>	The ares_destruct() function did not free all the domains that were allocated by the ares_construct() function. This was fixed.	ares_init.c	Everyone
16	<b>BUG</b>	When using a multithreaded application that tried to send data from several threads on the same TLS connection, the Stack could sometimes try to send the information before the handshake was completed, causing a connection shutdown. This error was fixed.	_SipTransport.c	Applications that use TLS

No.	Classification	Bug Description	Files Changed	Scope
15	<b>BUG</b>	A transmitter always terminated with a NETWORK_ERROR reason instead of a USER_COMMAND reason. This was fixed.	TransmitterObject.c	Everyone
14	<b>BUG</b>	On a connected call, if the remote party sent SUSBCRIBE and the NOTIFY for that SUBSCRIBE was received before the 2xx for the SUBSCRIBE, the subscription set the From header tag of the NOTIFY in the To header tag of the dialog. This behavior is correct only for subscriptions that are sent outside of a call-leg and resulted in a bad To header tag. This was fixed.	_SipSubs.c	Everyone
13	<b>BUG</b>	The Stack could not handle the situation where the select engine was destructed in the middle of RvSipMidPollEventsHandling(). This was fixed.	RvSipMid.c	Everyone
12	<b>CRASH</b>	When the construction of the SigComp Manager fails the Manager self-destruction is initiated. If the construction failed due to lack of memory for compartment or algorithm objects, the destruction caused a crashed.	SigCompCompartmentHandlerObject.c, SipComStateHandlerObject.c	Applications that use the SigComp add-on
11	<b>FEATURE</b>	Support for the UDPS transport protocol that is the recommended transport protocol for T.38 (besides TCP) was added to the SDP Stack.	rvsdpprsaux.c, rvsdpsymb.c, rvsdpsymb.h	Everyone
10	<b>FEATURE</b>	Additional functionality was added at the middle layer: RvSipMidTimerGetNextExpiration() and RvSipMidTimerGetTimerExpiration() were added to enable the application to obtain details on timer data.	RvSipMid.c, RvSipMid.h	Everyone

No.	Classification	Bug Description	Files Changed	Scope
9	<b>CRASH</b>	When the Stack was compiled without enhanced DNS and a UDP message failed to be sent synchronously, the Stack crashed. This was fixed.	TransactionControl.c	Everyone
8	<b>RESOURCES</b>	If the process of Subscription object creation is interrupted due to concurrent termination of the underlying call-leg object, the page allocated for the Subscription was never freed.	SubsMgrObject.c, SubsObject.c, SubsObject.h	Everyone
7	<b>BUG</b>	In case of the intensive use of sockets in WinCE.Net, the performance was reduced due to incremental socket fds, done by the OS, which the Stack did not take into consideration. This was fixed.	rvselect.c, rvselectinternal.h	WinCE.NET applications
6	<b>BUG</b>	When an outbound proxy was used and the call-leg object did not have a Route list, the proxy address was used for all requests instead of for initial requests only. This was fixed.	CallLegObject.c	Everyone
5	<b>FEATURE</b>	Some VxWorks BSPs were not able to identify the sin_family parameter of an incoming connection. The Stack now bypasses this problem by assuming that the opened socket is of IPv4 type, and the incoming connection will also be of IPv4 type.	rvsocket.c, TransportUDP.c	VxWorks users
4	<b>BUG</b>	When the preemption callback was called, the receive function was called only once. To clean the pipe, we now perform multiple read operations.	rvselect.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
3	<b>BUG</b>	The Stack DNS module tried to load the OS DNS configuration before applying the DNS parameters from the Stack configuration. If loading from the OS failed, the Stack failed to be constructed. Now if there is a DNS parameter in the configuration, the Stack will not try to load the OS configuration.	are.h, ares_init.h, rvares.c, rvares.h, RvSipStack.c, _SipTransmitterMgr.c, _SipTransmitterMgr.h, TransmitterMgrObject.c, TransmitterMgrObject.h	Everyone
2	<b>BUG</b>	When getting the local addresses of the computer, 127.0.0.1 was not returned. Because of this, it was impossible to connect a call with 127.0.0.1 when the network was down. This was fixed.	rghost.c	Windows users
1	<b>FEATURE</b>	Support was added to the following operating systems: pSOS, Nucleus, Symbian.	rvsocket.c, rvsocket.h, rvassert.h, rvcorestrings.h, rvclock.c, rvsemaphore.c, rvthread.c, rvthread.h, rvrandomgeneratore.c, _SipTransport.c	Customers that are interested in these specific operating systems

---

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

---

---

**Note** For updated information on known bugs and patches please visit the Technology Business Unit Customer Support section in the RADVISION Web site ([www.radvision.com](http://www.radvision.com)), or contact Customer Support directly.

---