

List of Fixed Bugs in the SIP Toolkit v3.1.1.24

The following issues have been fixed or added since in version 3.1.1.24 since version 3.1.1.20. 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
13	BUG	When the Stack spanned several threads and a WRITE event was received, the event was placed in the Stack processing queue, but the Stack continued to wait for a WRITE event on the specific socket. As a result, a loop of retries occurred to process the WRITE event. The loop stopped only after the previous WRITE event was removed from the processing queue and handled by another thread. This was fixed.	TransportTCP.c	Multithreaded applications
12	BUG	There were some situations where TLS connections were closed ungracefully, sometimes resulting in log errors. While this had no effect on TLS security or performance, it is preferable to close a connection gracefully. The Stack now closes TLS connections gracefully.	TransportTCP.c	TLS users
11	BUG	When a TCP message is received, it is inserted into the processing queue. If the connection received a close event while the message is in the queue, this message was ignored. The fix was that the message should be ignored only if the local party closed the connection.	TransportMsgBuilder.c	TCP problem
10	BUG	When querying a DNS server with SRV queries, the resulting answers may have been sorted wrongly. This resulted in DNS lists that had a higher order record before a lower order record. The sorting error was fixed.	TransportDNS.c	Applications that use the enhanced DNS feature.

No.	Classification	Bug Description	Files Changed	Scope
9	BUG	When destructing the Sip Stack, if the Stack failed to close a UDP local address, the destruct function returned without freeing all resources and therefore it was impossible to construct the Stack again. This was fixed.	TransportMgrObject.c	Everyone
8	BUG	In case of a plain SIP transmission session, followed by unanswered SigComp transmissions, and a late SigComp compressed response to this message, there was a problem to continue the general retransmission mechanism of the plain (decompressed) SIP message, as part of the SIP Stack recovery mechanism.	_SipTransmitter.c	Applications that use SigComp.
7	BUG	When using UDP transport, if the send function returned a WOULD_BLOCK status code, the Stack treated it as failure. This was fixed.	_SipTransport.c	Everyone
6	RESOURCES	When working with SigComp, a TCP request can be retransmitted several times with different sigcomp content (compressed, compressed with bytecode, plain). For each request, the remote side needs to retransmit the response. On receipt of more than one response on TCP, the client stopped the timer and the transaction was never freed. This was fixed.	TransactionTimer.c	Applications that use SigComp
5	BUG	When the application registers on the Stack log callback function, it can also supply a context that will be returned when the callback is called. This context was not returned by mistake and this was fixed.	RvSipStack.c	Everyone
4	CRASH	If the Stack failed to construct before a certain initialization in the select module, the Stack would crash. This error was fixed.	RvSipStack.c	Everyone

No.	Classification	Bug Description	Files Changed	Scope
3	BUG	When a call-leg received a request with a CSeq that is smaller than the last remote CSeq, this request was rejected. The fix was to reject the request with a 500 response code instead of a 400 response code.	CallLegTranscEv.c	Everyone
2	CRASH	When using the SDP Stack default allocator, the SDP Stack crashed when trying to parse a non-standard SDP message. This was fixed.	rvsdpprsaux.c	Everyone
1	FEATURE	Support was added to the following operating systems: Embedded Linux, WinCE 3.0, WinCE.Net and Integrity.	ares_init.c, rvoscomp.h, rvselect.c, rvsocket.c, rvsocket.h, rtp.c, rtcp.c, rvsdpvector.h	Customers that are interested in these specific operating systems

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 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), or contact Customer Support directly.