RTI Secure WAN Transport

Release Notes

Version 5.3.0

Your systems. Working as one.

Printed in U.S.A. First printing.
June 2017.

Trademarks

Copy and Use Restrictions

Technical Support

Real-Time Innovations, Inc.
232 E. Java Drive
Sunnyvale, CA 94089
Phone: (408) 990-7444
Email: support@rti.com
Website: https://support.rti.com/

Release Notes

1 Supported Platforms

This release of *RTI® Secure WAN Transport* is supported on the platforms in Table 1.1.

Table 1.1 **Supported Platforms**

Operating System		CPU	Compiler	RTI Architecture Abbreviation
Android®	All Android platforms in the *RTI Core Libraries Platform Notes* for the same version number. Note: RTI WAN Server is not supported.			
iOS®	All iOS platforms in the *RTI Core Libraries Platform Notes* for the same version number. Note: RTI WAN Server is not supported.			
Linux®	All Linux platforms in the *RTI Core Libraries Platform Notes* for the same version number, except not supported on SUSE 11 or any custom target platform.			
OS X®	All OS X platforms in the *RTI Core Libraries Platform Notes*.			
QNX®	QNX Neutrino® 6.5	x86	qcc 4.4.2 with GNU C++ libraries	i86QNX6.5qcc_gpp4.4.2
	QNX Neutrino 6.5.0 SP1	ARMv7a Cortex	qcc 4.4.2 with Dinkum libraries	armv7aQNX6.5.0SP1qcc_cpp4.4.2
Solaris™	Solaris 2.10	UltraSPARC®	gcc3.4.2	sparcSol2.10gcc3.4.2
Windows®	All Windows platforms in the *RTI Core Libaries Platform Notes*.			

2 Compatibility with Current Software

RTI Secure WAN Transport is an optional product for use with *RTI Connext® DDS* with the same version number.

This release uses OpenSSL 1.0.2j.

3 Backward Compatibility

- In *Connext DDS* 5.1.0, the default value for **message_size_max** for this transport changed. *Secure WAN Transport* also uses this value. Consequently, *Secure WAN Transport* 5.1.0 and higher is not off-the-shelf compatible with applications running older versions of this transport. See the *RTI Core Libraries Release Notes* for instructions on how to resolve the compatibility issue with older *Connext DDS* and *RTI Data Distribution Service* applications.
- If you were using OpenSSL 1.0.1: Because *Connext DDS* 5.2.3 and higher uses OpenSSL 1.0.2, the number of bits of any Diffie-Hellman (DH) parameters must now be at least 1024 (see https://www.openssl.org/blog/blog/2015/05/20/logjam-freak-upcoming-changes/). Therefore, if you are using the property **tls.cipher.dh_param_files** and there is a DH parameter file that has fewer than 1024 bits, you must regenerate the file with at least 1024 bits.

4 What's New in 5.3.0

4.1 New Platforms

This release adds support for platforms on the following operating systems:

- OS X 10.12
- Red Hat Enterprise Linux 6.8
- Ubuntu 16.04 LTS
- Windows Server 2016

See the *RTI Core Libraries Platform Notes* for details.

4.2 Platforms on Legacy Operating Systems

The following legacy operating systems have reached end-of-life from their corresponding vendors. Please contact RTI support or your account manager if you require version 5.3 to run on these platforms:

- Red Hat Enterprise Linux 5.0
- Wind River Linux 4

4.3 Removed Platforms

Platforms on the following operating systems are no longer supported:

- OS X 10.8
- Windows 2003, Windows Vista, Windows XP

4.4 OpenSSL

This release uses OpenSSL 1.02g.

5 What's Fixed in 5.3.0

5.1 DTLS Communication Failed when Using More than One Network Interface

Participants using the DTLS transport did not communicate if one of the machines used more than one network interface. That machine reported errors such as:

```
NDDS_Transport_DTLS_Connection_process_receive:OpenSSL error:
[04091068]:[INT_RSA_VERIFY]:[bad signature]
```

This problem has been resolved.

[RTI Issue ID COREPLG-385]

5.2 Segmentation Fault when Creating DTLS DomainParticipants in Multiple Threads on QNX and Solaris Systems

On QNX and Solaris systems, the creation of DTLS-enabled DomainParticipants was not thread-safe and may have led to a segmentation fault in the function RTIOsapiSemaphore_take(). This problem has been resolved for all RTI Secure WAN Transport architectures.

[RTI Issue ID COREPLG-264]

---

6 Known Issues

- When communicating over some networks, the Secure WAN Transport plug-ins may fail to send data larger than the MTU (maximum transmission unit) size available for the network. This is especially likely over wide-area networks. This scenario is also a suggested configuration of the DTLS protocol, according to the DTLS specification, which is IETF RFC 4347.

  If problems occur while sending large packets, set the maximum_message_size transport property to the MTU of your network *minus 28 bytes for the DTLS header* and set up your application according to the Large Data Use Cases "How To" provided in the online (HTML) documentation. For example, for an MTU size of 1500 bytes (for standard Ethernet), set **maximum_message_size** to 1500 - 28 = 1472.

  One instance of this problem for which there is no workaround is the case where the discovery packets are larger than your network's MTU. This could occur if user data, propagated properties, or type-codes are configured.

- An application using the WAN transport may appear to hang for several minutes if the WAN server is shut down and not restarted before the application tries to contact it, or if the application is unable to communicate with the WAN server.

  Two scenarios under which the application tries to contact the STUN server are during shut down and while establishing a connection with the initial peers.

  This issue is due to a sequence of synchronous STUN transactions with the STUN server. If you need to run WAN transport without a STUN server, here are some recommendations:

  - Decrease the blocking time by decreasing the number of STUN retransmissions. To do so, change the property, **stun_number_of_retransmissions**. For example, a change from the default of 7 retries to 5 retries will result in a total period of 3.1 seconds per synchronous operation. Note however, that this may impact the ability to reliably set up connections to peers over a WAN.
  - Decrease the blocking time by using a participant ID limit of zero when configuring the initial peer descriptors.

    For example, when the peer descriptor **wan://::1:10.10.1.150** is specified, DDS will try to contact five participants with the same WAN ID in different ports. Usually there is only one participant using the same WAN ID. Although the other four participants will never be reachable, the application still tries to establish communication with them by contacting the STUN server.

You can reduce the number of participants to which the application will try to contact to one by using a participant ID limit of zero in the peer descriptor. For example, **0@wan://::1:10.10.1.150**.

For information on peer descriptors, see the *Discovery* chapter in the *RTI Core Libraries User's Manual*.

7 Third-Party Licenses

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

   This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

   This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).