RTI Security Plugins Release Notes

Version 7.7.0



Contents

1 Supported Platforms **1**

2 Compatibility **2**
2.1 Compatibility with OpenSSL 3.5 . . . 2
2.2 Compatibility with wolfSSL 5.8 . . . 2

3 What's New in 7.7.0 **4**
3.1 Distributed secure logging for the Lightweight Security Plugins . . . 4
3.2 Upgrade to wolfSSL 5.8.2 . . . 5
3.3 Alternative Certificate Authorities configuration easier by using data directly . . . 5
3.4 Faster repairs of lost samples due to shorter default heartbeat periods for Secure Key Exchange Builtin Topic Writer . . . 5
3.5 Improved network utilization and memory footprint for intraparticipant communication . . . 6
3.6 Simplified configuration when enabling origin authentication for message protection . . . 6
3.7 Support for configuring flags used to load Security Plugins dynamic library . . . 7
3.8 Deprecations and Removals . . . 8
3.8.1 Authentication properties deprecated . . . 8
3.8.2 Removed support for AES-192-GCM symmetric cipher algorithm . . . 8
3.8.3 Removal of the deprecated SecurityPluginSuite_create entry point . . . 8
3.9 Third-Party Software Changes . . . 8

4 Whats Fixed in 7.7.0 **9**
4.1 Cryptography . . . 9
4.1.1 [Critical] Unable to write samples when using batching and serialized data protection 9
4.2 Dynamic Participant Renewal, Revocation, and Expiration . . . 10
4.2.1 [Critical] Unexpected loss of keyed data when Persistence Service enabled Key Revision History . . . 10
4.2.2 [Major] RTI Security Plugins were not compatible with monotonic clock . . . 10
4.2.3 [Major] Authentication failures when discovering a DomainParticipant and simultaneously updating the local Permissions Document * . . . 11
4.3 Crashes . . . 11
4.3.1 [Critical] Crash when using Monitoring Library 2.0 and logging.mode_mask set to SECURITY_TOPIC . . . 11
4.3.2 [Critical] Potential undefined behavior after error acquiring mutex during remote identity validation or local Permissions Document Update * . . . 11
4.3.3 [Critical] Invalid locking situation when creating DomainParticipant with Security enabled * . . . 12

4.3.4 [Critical] Invalid read when simultaneously changing Certificate Revocation List and validating local DomainParticipant's identity status * . . . 12
4.3.5 [Critical] Problems on VxWorks when multiple threads tried creating the first DomainParticipant or deleting the last DomainParticipant . . . 12
4.3.6 [Critical] Potential stack buffer read overflow in Builtin Security Plugins when registering local endpoint . . . 13
4.3.7 [Critical] Potential segmentation fault in Builtin Security Plugins when discovering a DomainParticipant . . . 13
4.3.8 [Major] Potential precondition failure when removing a remote DomainParticipant that has secure endpoints . . . 13
4.4 Hangs . . . 14
4.4.1 [Critical] Application hang after failing to post a local credential validation event during DomainParticipant enablement . . . 14
4.4.2 [Major] Excessive resource consumption in certain scenarios * . . . 14
4.5 Vulnerabilities . . . 14
4.5.1 [Critical] Potential Denial of Service in RTI Security Plugins for OpenSSL when loading Private Key . . . 15
4.5.2 [Critical] Potential stack buffer overflow in RTI Security Plugins for OpenSSL when loading Private Key . . . 15
4.5.3 [Critical] Potential Denial of Service in RTI Security Plugins for OpenSSL when parsing Permissions Document . . . 16
4.5.4 [Critical] Potential Denial of Service in RTI Security Plugins for OpenSSL when configuring client proxy . . . 16
4.5.5 [Critical] A malicious authenticated participant could impersonate other participants 17
4.5.6 [Critical] Origin Authentication could be bypassed by crafting malicious RTPS message 18

5 Previous Releases 19
5.1 What's New in 7.6.0 . . . 19
5.1.1 Additional Authenticated Data (AAD) enabled by default for Builtin Security Plugins when using PSK or RTPS Header Extension . . . 19
5.1.2 Enforcement of DomainParticipant permissions expiration enhances security . . . 20
5.1.3 Support for runtime extension of Permissions Document validity and enforcement of expiration . . . 20
5.1.4 Pre-Shared Key Protection now fully aligns with DDS Security 1.2 specification . . . 20
5.1.5 Key revisions enabled by default to improve out-of-the-box security . . . 21
5.1.6 RTI Lightweight Security Plugins support ShangMi cryptographic algorithms . . . 21
5.1.7 New error message when Core and Security Plugins library versions differ . . . 21
5.1.8 Upgrade OpenSSL to 3.5.1 . . . 21
5.1.9 Deprecations and Removals . . . 22
Deprecation of authentication.crl and removal of authentication.crl_file . . . 22
Support for cryptography.enable_additional_authenticated_data property removed . . . 22
5.2 Whats Fixed in 7.6.0 . . . 22
5.2.1 Discovery . . . 23
[Critical] No communication when using Pre-Shared Key Protection if remote participant did not propagate PID_DOMAIN_ID during discovery . . . 23
5.2.2 Discovery and Authentication . . . 23
[Critical] Authenticated participant could not replace unauthenticated one if QoS changed during authentication . . . 23

5.2.3 Usability . . . 24
[Minor] Using properties “dds.license.license_string” or “dds.license.license_file” led to DomainParticipant creation failure . . . 24
5.2.4 Cryptography . . . 24
[Critical] Repair of dispose messages failed when Key Revision History enabled . . . 24
5.2.5 Dynamic Participant Renewal, Revocation, and Expiration . . . 25
[Minor] Local DomainParticipant did not detect more recent expiration date in updated Identity Certificate . . . 25
5.2.6 Crashes . . . 25
[Critical] Segmentation fault after creating and immediately deleting a DomainParticipant . . . 25
5.2.7 Memory Leaks/Growth . . . 25
[Major] Memory leak when setting a subject name whitelist while security was not enabled . . . 25
[Minor] Memory leak when dds.participant.trust_plugins.expiration_advance_notice_duration.sec property value was invalid . . . 26
5.2.8 OMG Specification Compliance . . . 26
[Trivial] RTI Lightweight Security Plugins propagated incorrect supported symmetric cipher algorithms value . . . 26
5.2.9 Interoperability . . . 26
[Major] Lightweight Security Plugins did not use Additional Authenticated Data by default . . . 26
5.2.10 Vulnerabilities . . . 26
[Critical] Potential heap buffer read overflow in Security Plugins when using a malicious identity certificate file . . . 26
5.3 What’s New in 7.5.0 . . . 27
5.3.1 Key revisions now enabled in a manner compliant with the DDS Security specification 28
5.3.2 Support for kicking participants off a system that has non-RTI Participants . . . 28
5.3.3 Support for Online Certificate Status Protocol . . . 28
5.3.4 New Governance configuration for CRL and OCSP . . . 29
5.3.5 Improved compatibility of C++ applications using RTI Security Plugins SDK . . . 29
5.3.6 Deprecations and Removals . . . 29
HMAC-Only mode removed from Builtin Security Plugins . . . 29
5.4 What’s Fixed in 7.5.0 . . . 29
5.4.1 Discovery and Authentication . . . 30
[Critical] Unable to communicate with DomainParticipants that propagate 1.0 class_id and use P384 or experimental algorithms . . . 30
5.4.2 Dynamic Participant Renewal, Revocation, and Expiration . . . 30
[Critical] Loss of data when banishing more than 16,777,208 times . . . 30
5.4.3 Crashes . . . 31
[Critical] Log messages possibly corrupted or lost due to memory reordering on weak memory architectures . . . 31
[Critical] NULL dereference when simultaneously changing certificate or CRL file while deleting DomainParticipant . . . 31
[Critical] Segmentation fault risk with concurrent updates of identity certificates . . . 31
5.4.4 Memory Leaks/Growth . . . 31
[Critical] Unbounded memory leak if identity certificate file was polled and emptied repeatedly . . . 31

5.5 What's New in 7.4.0 . . . 32
5.5.1 Improve isolation in large-scale secure systems using domain tags . . . 32
5.5.2 Adapt permissions at run-time with new support for participant partitions . . . 32
5.5.3 Simplify your permissions by matching a rule with a group of identities . . . 32
5.5.4 Inactive Secure Key Exchange DataReader now implies inactive DomainParticipant . 33
5.5.5 Removed risk of unbounded memory growth on Secure Key Exchange DataWriter . 33
5.5.6 Authentication class_id now conforms to the value in the DDS Security 1.2 Specification 33
5.5.7 New standard property to configure shared secret algorithm, to comply with specification . . . 34
5.5.8 Deprecations and Removals . . . 34
authentication.key_establishment_algorithm property deprecated . . . 34
5.6 What's Fixed in 7.4.0 . . . 34
5.6.1 Discovery and Authentication . . . 35
[Critical] Unable to communicate with DomainParticipants that propagate a different authentication class_id version . . . 35
5.6.2 Logging . . . 35
[Minor] Setting logging.verbosity to NOTICE did not list provider information . . . 35
[Trivial] Security category missing in some security-related log messages . . . 36
5.6.3 Access Control . . . 36
[Trivial] Grant without a subject name silently ignored when parsing Permissions Document . . . 36
5.6.4 Crashes . . . 37
[Critical] Java: segmentation fault while enabling a DataWriter that protects payloads and enables batching . . . 37
[Critical] Parsing malformed protected RTPS packets may have caused bus error on some platforms . . . 37
[Critical] Invalid read when simultaneously changing identity certificate file and validating local DomainParticipant's identity status . . . 37
[Critical] Invalid read when simultaneously changing CRL file and validating remote DomainParticipant's identity . . . 38
[Major] Possible crash when using Python API due to collision with OpenSSL symbols when using certain Python releases . . . 38
5.6.5 Hangs . . . 39
[Critical] Potential uncontrolled CPU usage when parsing Governance or Permissions Documents due to a vulnerability in Expat . . . 39
5.6.6 Vulnerabilities . . . 39
[Critical] Heap buffer overflow while creating a DomainParticipant that uses a malformed Identity Certificate . . . 39

6 Known Issues **41**
6.1 No Support for ECDSA-ECDH with Static OpenSSL Libraries and Certicom Security Builder 41
6.2 No Support for Writing >65kB Unfragmented Samples Using Metadata or RTPS Message Protection . . . 41
6.3 subscription_data and publication_data in check_local_datawriter_match / check_local_datareader_match are not Populated . . . 42
6.4 relay_only parameter in check_remote_datareader is not Populated . . . 42
6.5 'Allow Rule' Patterns Incorrectly do not Allow Subset Patterns in QoS . . . 42
6.6 Lightweight Security Plugins do not support Distributed Logging . . . 42

7 Copyrights and Notices **44**

Chapter 1

Supported Platforms

See the Core Libraries Platform Notes for supported platforms.

Chapter 2

Compatibility

This release of the Security Plugins *(RTI Security Plugins)* includes partial support for the DDS Security 1.2 Specification from the Object Management Group (OMG).

The Security Plugins 7.7.0 are interoperable with the *Security Plugins* 5.2.7 and higher.

Persistence Service databases secured with the Security Plugins 7.7.0 are incompatible with databases generated by versions of *Persistence Service* older than 7.0.0.

When using the Security Plugins SDK *(RTI Security Plugins SDK)*, the required minimum version of CMake is 3.12 if linking dynamically and 3.13 if linking statically.

In release 7.7.0, the Security Plugins are available for use with OpenSSL® 3.5 and wolfSSL® 5.8. There are separate installation packages for each of these options.

For more information about other backward compatibility issues, see the *Migration Guide* on the RTI Community Portal (https://community.rti.com/documentation).

2.1 Compatibility with OpenSSL 3.5

The Security Plugins 7.7.0 for OpenSSL are API-compatible with OpenSSL 3.5. The Security Plugins 7.7.0 have only been tested by RTI using OpenSSL 3.5.5. OpenSSL 3.5.5 is not compatible with Security Plugins versions that were tested using OpenSSL 3.0.9 or below.

The Security Plugins SDK has been tested with OpenSSL 3.5.5.

2.2 Compatibility with wolfSSL 5.8

The Security Plugins 7.7.0 for wolfSSL have been tested with wolfSSL 5.8.2. You can find a list of the supported target platforms in Supported Platforms, in the RTI Connext Core Libraries Release Notes.

Limitations when using wolfSSL:

The Security Plugins for wolfSSL are interoperable with the Security Plugins for OpenSSL in most configurations. However, there are some features that are not supported by the Security Plugins for wolfSSL:

- Diffie-Hellman: The SECURITY PLUGINS for wolfSSL only support the ECDHE-CEUM+P256 and ECDHE-CEUM+P384 Elliptic Curve Diffie-Hellman (ECDHE) key establishment algorithms.
- RSASSA-PSS-MGF1SHA256+2048+SHA256: The SECURITY PLUGINS for wolfSSL support for digital signatures is limited to the RSASSA-PKCS1-V1_5+2048+SHA256, ECDSA-P256+SHA256, and ECDSA-P384+SHA384 algorithms.
- OpenSSL engines/providers are not supported.
- The SECURITY PLUGINS for wolfSSL do not support OCSP (Online Certificate Status Protocol).
- If your system configures an RSA Permissions Certificate Authority, generate the PKCS7 signed Governance and Permissions Documents using the *openssl smime* tool instead of the *openssl cms* tool. Otherwise, verification of the PKCS7 document will fail.

Chapter 3

What's New in 7.7.0

This section describes what's new in *RTI® Security Plugins* 7.7.0, compared to *7.6.0*.

SECURITY PLUGINS 7.7.0 is the first production release in the 7.7 long-term support (LTS) series. It is built upon and combines all of the features in releases 7.4.0, 7.5.0, and 7.6.0. See Connext Versions and Lifecycle for more information on RTI's software release model.

For what's new in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation. Or start with RTI Connext What's New for a launchpad to all products.

Note: For backward compatibility information between 7.7.0 and previous releases, see the *Migration Guide* on the RTI Community Portal.

3.1 Distributed secure logging for the Lightweight Security Plugins

Lightweight *DomainParticipants* can now distribute security logging information over DDS. You can subscribe to this data via the new `DDS:Security:PSK:LogTopic` Builtin Secure Logging *Topic*. You can subscribe to the new builtin *Topic* from *DomainParticipants* that are using either the *Lightweight Builtin Security Plugins* or the *Builtin Security Plugins* in a configuration that can interoperate with Lightweight security. Note that Lightweight *DomainParticipants* remain unable to subscribe to the `DDS:Security:LogTopicV2` logging *Topic* because it is protected at the *Topic* level.

See Security Events and Logging for more information.

3.2 Upgrade to wolfSSL 5.8.2

The Security Plugins for wolfSSL are now based on wolfSSL version 5.8.2.

For release 7.7.0, the *Security Plugins* are API-compatible with wolfSSL version 5.8.2, not with earlier versions. Note that the *Security Plugins* 7.7.0 have only been tested by RTI using wolfSSL 5.8.2.

In addition to the upgraded version of wolfSSL, the Security Plugins for wolfSSL now support partial chains. The certificate of the Identity Certificate Authority no longer has to be self-signed; intermediate Certificate Authorities are also supported.

3.3 Alternative Certificate Authorities configuration easier by using data directly

Starting in this release, you should no longer configure the alternative Certificate Authorities as a list of files. Instead, the format of the alternative Certificate Authorities is now consistent with the Identity Certificate and Certificate Revocation List (CRL).

The new format accepts both the `data:,` and `file:` (default) URI schemes. If the property value starts with the `data:,` prefix, then it must continue with the concatenation of one or more certificates in PEM format, separated by the newline characters. Otherwise, the property must point to a file containing the list in the same format.

The `authentication.alternative_ca_files` and `access_control.alternative_permissions_authority_files` properties are now deprecated. You should migrate to the new `authentication.alternative_identity_cas` and `access_control.alternative_permissions_cas` properties. These properties accept the previously described format and support configuring the alternative Identity and Permissions CA by either pointing to a file or directly as data.

See Properties for Configuring Authentication and Properties for Configuring Access Control, in the *Security Plugins User's Manual*, for more information on these properties.

3.4 Faster repairs of lost samples due to shorter default heartbeat periods for Secure Key Exchange Builtin Topic Writer

The out-of-the-box default values for the `SecureVolatileWriter::heartbeat_period`, `fast_heartbeat_period`, and `late_joiner_heartbeat_period` for the Secure Key Exchange Builtin *Topic DataWriter* have been changed to make them faster. `SecureVolatileWriter::max_heartbeat_retries` has also been changed to maintain the same amount of time it takes before the Secure Key Exchange Builtin *Topic DataReader* is considered inactive. Finally, `SecureVolatileWriter::heartbeats_per_max_samples` increased so the heartbeats are sent with DATA packets more often. The changes are as follows:

DataWriter Kind	heartbeat_period	fast_heartbeat_period	late_joiner_heartbeat_period	max_heartbeat_retries	heartbeats_per_max_samples
Secure Volatile DataWriter	1 sec → 200 ms	250 ms → 200 ms	1 sec → 200 ms	800 → 1,000	1 → 100,000,000

The new defaults speed up the repair process when a symmetric key sample has been lost. Faster repairs mean that communication can begin faster. With faster heartbeat periods, you may notice an increase in network traffic because more heartbeats and acknowledgments are sent. You may choose to slow down the defaults if your network is not lossy, and you can wait longer for samples to be repaired if they are lost.

The `max_heartbeat_retries` QoS value was also changed. It was changed from 800 to 1000 so that the time it takes before the Secure Key Exchange Builtin *Topic DataWriter* considers the Secure Key Exchange Builtin *Topic DataReader* inactive remains the same as in previous releases (200 seconds).

3.5 Improved network utilization and memory footprint for intraparticipant communication

The Secure Key Exchange Channel is used to exchange keys between *DomainParticipants*. In previous releases, it was also used to exchange keys within the same *DomainParticipant*. The intraparticipant key exchange is not necessary because a *DomainParticipant* always knows its own keys. This release improves network utilization and memory footprint by removing the intraparticipant key exchange.

If you use the *Security Plugins* SDK, please see the *Security Plugins Interface (SPI) Notes* section of the SDK documentation for details on the interface changes that were required for this improvement.

3.6 Simplified configuration when enabling origin authentication for message protection

Previous releases required you to set the `cryptography.max_receiver_specific_macs` property in order to use `WITH_ORIGIN_AUTHENTICATION` in any of the Governance Document `protection_kind` fields. This requirement resulted in one or more of the following:

- Lack of support for an unbounded number of MACs. The transport message_size_max (which could be very small in systems with a small maximum transmission unit (MTU)) limited the maximum potential value of `cryptography.max_receiver_specific_macs` due to the need to send all applicable destinations' MACs associated with an RTPS message.
- Security degradation when the configured limit was exceeded for a given RTPS message.
- Potential waste of resources (memory and inefficient usage of the network MTU) because of pre-allocating all the keys necessary to support the configured limit.

This release removes this requirement for the `rtps_protection_kind`. If your `rtps_protection_kind` contains `WITH_ORIGIN_AUTHENTICATION` and no other `protection_kind`

contains `WITH_ORIGIN_AUTHENTICATION`, then you no longer need to set `cryptography.max_receiver_specific_macs`.

For more information, see Origin Authentication Protection Implications in the *RTI Security Plugins User's Manual*.

3.7 Support for configuring flags used to load Security Plugins dynamic library

If you are linking the *Security Plugins* library dynamically, you can now configure the flags used to load the *Security Plugins* library. This option is supported on Linux, macOS, and QNX systems. It is also supported on Android and VxWorks systems when using RTP mode. It is not supported on Windows systems.

In addition to the `com.rti.serv.secure.library` and `com.rti.serv.secure.create_function` properties, now you can configure the `com.rti.serv.secure.library_flags` property. This property takes an integer representing the flags passed to the `dlopen()` function. You must look for the architecture-specific integer value for each flag you want to configure, and calculate the final property value as the *OR* operation of all these flags. If the `com.rti.serv.secure.library_flags` property is not set, the default flags are:

```
RTLD_NOW | RTLD_DEEPBIND
```

For example, on Linux systems, these flags translate to `0x00002` and `0x00008`, resulting in a final value of `0x0000A`; not configuring the `com.rti.serv.secure.library_flags` property has the same behavior as setting the property value to `0x0000A`.

This feature is useful if you need to customize how to load the *Security Plugins* dynamic library. For example, you may need to customize the flags if you want to enable Address Sanitizer (ASan) in applications using the *Security Plugins*. Enabling modern versions of ASan in combination with the *Security Plugins* will result in the following error (if the default flags are used):

```
WAR==683==You are trying to dlopen a libnddssecurityd.so shared library with␣
↪RTLD_DEEPBIND flag which is incompatible with sanitizer runtime (see https:/
↪/github.com/google/sanitizers/issues/611 for details). If you want to run␣
↪libnddssecurityd.so library under sanitizers please remove RTLD_DEEPBIND␣
↪from dlopen flags.
```

This error is due to an Address Sanitizer bug. Continuing with our Linux example, you can work around the error by setting the `com.rti.serv.secure.library_flags` property to `0x00002` (`RTLD_NOW`). You should only do this if you don't intend to use the Python API, or if you use the Python API but are sure your Python version doesn't have a builtin OpenSSL. For more information, see *[Major] Possible crash when using Python API due to collision with OpenSSL symbols when using certain Python releases*.

3.8 Deprecations and Removals

Deprecated means that the item is still supported in this release, but will be removed in a future release. *Removed* means that the item is discontinued or no longer supported.

Any deprecations or removals noted in RTI's documentation serve as notice under the Real-Time Innovations, Inc., Maintenance Policy #4220 and/or any other agreements by and between RTI and customer regarding maintenance and support of RTI's software. RTI's current standard terms and support and maintenance policies are available at https://www.rti.com/terms.

3.8.1 Authentication properties deprecated

As mentioned in *Alternative Certificate Authorities configuration easier by using data directly*, the `authentication.alternative_ca_files` and `access_control.alternative_permissions_authority_files` properties are now deprecated.

3.8.2 Removed support for AES-192-GCM symmetric cipher algorithm

Following the deprecation of AES-192-GCM in release 7.3.0, release 7.7.0 now completely removes this cryptographic algorithm from the list of available symmetric cipher algorithms.

If you are still using AES-192-GCM in your system, please transition to AES-256-GCM or AES-128-GCM. For more information about symmetric cipher algorithms, see Algorithms Involved in Protecting Secure Entities Traffic in the *RTI Security Plugins User's Manual*.

3.8.3 Removal of the deprecated SecurityPluginSuite_create entry point

As described in the *Migration Guide* here, the Security Plugins have removed the `SecurityPluginSuite_create` entry point.

3.9 Third-Party Software Changes

The following third-party software used by Security Plugins has been upgraded:

Table 3.1: Third-Party Software Upgrades

Third-Party Software	Old Version	New Version
OpenSSL	3.5.1	3.5.5

For information on third-party software used by *Connext* products, see the "3rdPartySoftware" documents in your installation: <NDDSHOME>/doc/manuals/connext_dds_professional/release_notes_3rdparty.

Chapter 4

Whats Fixed in 7.7.0

This section describes bugs fixed in *RTI® Security Plugins* 7.7.0. These are fixes made since *7.6.0*.

Security Plugins 7.7.0 is the first production release in the 7.7 long-term support (LTS) series. It is built upon and combines all of the features in releases 7.4.0, 7.5.0, and 7.6.0. See Connext Versions and Lifecycle for more information on RTI's software release model.

For what's fixed in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation.

[Critical]: System-stopping issue, such as a crash or data loss. [Major]: Significant issue with no easy workaround. [Minor]: Issue that usually has a workaround. [Trivial]: Small issue, such as a typo in a log.

4.1 Cryptography

4.1.1 [Critical] Unable to write samples when using batching and serialized data protection

If you had:

- A *Topic* that protected serialized data. That is, a data_protection_kind in the Governance Document other than `NONE`.
- A *DataWriter* that used batching. That is, DDS_DataWriterQos::batch.enable set to DDS_BOOLEAN_TRUE.

The FooDataWriter_write function incorrectly returned DDS_RETCODE_ERROR after generating the following error messages:

```
ERROR [[...]{[...]}|WRITE] COMMENDFacade_canSampleBeSent:NOT SUPPORTED |␣
↪Fragment data not supported by this writer.
ERROR [[...]{[...]}|WRITE] [...]:sample cannot be sent
ERROR [[...]{[...]}|WRITE] PRESPsWriter_flushBatchWithCursor:FAILED TO SEND |␣
↪Reliable write
ERROR [[...]{[...]}|WRITE] PRESPsWriter_writeBatchInternal:!error flushing␣
↪batch
```

This occurred when *Connext* attempted to flush a batch because it reached the message_size_max of the transport (instead of other limits, such as batch.max_data_bytes or batch.max_samples).

[RTI Issue ID SEC-2838]

4.2 Dynamic Participant Renewal, Revocation, and Expiration

4.2.1 [Critical] Unexpected loss of keyed data when Persistence Service enabled Key Revision History

Consider the following scenario:

- You are using *Persistence Service*.
- You are setting the *Persistence Service DomainParticipant* property `dds.participant.trust_plugins.key_revision_max_history_depth` to a non-zero value.
- You are using a keyed data type.
- You trigger enough key regenerations (e.g., by changing the Certificate Revocation List or the Subject Name Whitelist) to reach the `key_revision_max_history_depth` limit.

In this scenario, *Persistence Service* incorrectly failed to re-encode the historical samples of the keyed data type that were encoded with the oldest key revision; as a result, those samples were unexpectedly lost to newly discovered *DataReaders*. The following error message occurred during the failed re-encoding:

```
attempt to decrypt empty input
```

[RTI Issue ID SEC-2906]

4.2.2 [Major] RTI Security Plugins were not compatible with monotonic clock

The *RTI Security Plugins DomainParticipant's* identity certificate (OCSP included) and permissions document expirations did not work properly when the internal clock (`dds.clock.internal_clock` property) was set to "monotonic". This issue affected both local and remote artifacts. It caused *DomainParticipants* to not react to their expiration; that is, users were not notified about the expiration, and remote *DomainParticipants* with expired artifacts were not banished.

Expirations are now tracked regardless of the *Connext* clock settings, provided that the host system clock is set to the correct time and date.

[RTI Issue ID SEC-2813]

4.2.3 [Major] Authentication failures when discovering a DomainParticipant and simultaneously updating the local Permissions Document *

You may have faced authentication errors when discovering remote *DomainParticipants* right after updating the local Permissions Document:

```
ERROR [PARSE MESSAGE|[...]{Entity=DR,MessageKind=DATA}|RECEIVE FROM [...
↪]|RECEIVE SAMPLE|PROCESS HANDSHAKE|GET SECURITY STATE|LC:Security] RTI_
↪Security_AccessControl_validate_remote_permissions:{"DDS:Security:LogTopicV2
↪":{[...], "x":[{"DDS":[{"domain_id":[...]},{"guid":[...]},{"plugin_class":
↪"DDS:Access:Permissions"},{"plugin_method":"RTI_Security_AccessControl_
↪validate_remote_permissions"}]}],"m":"failed to validate remote permissions
↪"}}
```

These errors would last until authentication timeout (`dds.participant.trust_plugins.authentication_timeout.sec` property value), after which the re-authentication would succeed.

[RTI Issue ID SEC-2863]

* *This bug does not affect you if you are upgrading from 7.3.x or earlier.*

4.3 Crashes

4.3.1 [Critical] Crash when using Monitoring Library 2.0 and logging.mode_mask set to SECURITY_TOPIC

When the `logging.mode_mask` property was set to `SECURITY_TOPIC` while *Monitoring Library 2.0* was enabled, a crash occurred when a security event was logged in an application.

[RTI Issue ID SEC-2897]

4.3.2 [Critical] Potential undefined behavior after error acquiring mutex during remote identity validation or local Permissions Document Update *

In the unlikely scenario of failing to acquire a mutex used to protect against concurrent updates of security artifacts (specifically, Certificate Revocation Lists and Permissions Documents), the *Builtin Security Plugins* could have exhibited undefined behavior due to attempting to release the not-acquired mutex.

[RTI Issue ID SEC-2885]

4.3.3 [Critical] Invalid locking situation when creating DomainParticipant with Security enabled *

An unlikely crash or locked mutex escape could happen during the creation of a secure *DomainParticipant*, when using the *Builtin Security Plugins*. This could happen only during the creation of the secure *DomainParticipant* when facing additional issues, such as memory exhaustion.

[RTI Issue ID SEC-2883]

4.3.4 [Critical] Invalid read when simultaneously changing Certificate Revocation List and validating local DomainParticipant's identity status *

Suppose you had OCSP (Online Certificate Status Protocol) enabled, through either the `dds.sec.auth.ocsp_responder_uri` property or the Identity Certificate's `authorityInfoAccess` field. If you changed the contents of your Certificate Revocation List while validating the identity status of the local *DomainParticipant*, a race condition would have occurred because those two operations were not thread-safe with respect to each other. Validation of the local *DomainParticipant* identity status happens when:

- There is a change in the Identity Certificate.
- The Identity Certificate expiration date is closer than the value of the `dds.participant.trust_plugins.expiration_advance_notice_duration.sec` property.
- The expiration date of the OCSP response is closer than the value of the `dds.participant.trust_plugins.certificate_status_advance_update_duration` property.

This race condition led to the reading of invalid memory. A memory-checking tool such as Valgrind™ would have reported invalid reads in a function due to accessing an address freed by a different function. Your application may have crashed with this log message:

```
100: ../crypto/x509/x509cset.c:82: OpenSSL internal error: refcount error
```

[RTI Issue ID SEC-2853]

4.3.5 [Critical] Problems on VxWorks when multiple threads tried creating the first DomainParticipant or deleting the last DomainParticipant

On VxWorks® platforms, a segmentation fault may have occurred if multiple threads concurrently tried to create the initial *DomainParticipant*. Additionally, a memory leak might have resulted from multiple threads simultaneously trying to delete the final *DomainParticipant* in an application.

[RTI Issue ID SEC-2810]

4.3.6 [Critical] Potential stack buffer read overflow in Builtin Security Plugins when registering local endpoint

An application using the *Builtin Security Plugins* may have attempted to read invalid memory during the creation of a local secure entity (*DomainParticipant*, *DataWriter*, or *DataReader*). This could have resulted in a segmentation fault. As a result of reading invalid data, there was also a low chance of a discovery incompatibility due to a mismatch of security algorithms.

[RTI Issue ID SEC-2849]

4.3.7 [Critical] Potential segmentation fault in Builtin Security Plugins when discovering a DomainParticipant

Assume that in the Governance Document, at least one `protection_kind` field, other than `rtps_protection_kind`, is set to a value other than `NONE`. When one *DomainParticipant* was discovering another, there was a race condition leading to a NULL pointer dereference in the internal function `RTI_Security_Cryptography_registerEndpoint`. This problem only affected versions 7.3.0 and above.

[RTI Issue ID SEC-2913]

4.3.8 [Major] Potential precondition failure when removing a remote DomainParticipant that has secure endpoints

Consider the following scenario:

- *DomainParticipant* ParticipantA has endpoints that have `data_protection_kind` or `metadata_protection_kind` set to a value other than `NONE` in the Governance Document.
- *DomainParticipant* ParticipantB was communicating with ParticipantA but is no longer doing so.
- ParticipantB is using debug libraries.

While ParticipantB was removing ParticipantA, there was a race condition that made ParticipantB sometimes generate the following precondition error in the internal function `RTI_Security_CryptographyKeyHandle_removeFromEndpointList`:

```
!precondition: "self->entityType == RTI_SECURITY_ENDPOINT_TYPE_UNKNOWN &&␣
↪self->nextRemoteEndpointKey != ((void *)0)"
```

The precondition error was followed by invalid writes due to improper cleanup.

This problem did not affect release libraries and has been fixed by removing the incorrect precondition.

[RTI Issue ID SEC-2909]

* *This bug does not affect you if you are upgrading from 7.3.x or earlier.*

4.4 Hangs

4.4.1 [Critical] Application hang after failing to post a local credential validation event during DomainParticipant enablement

DomainParticipant enablement failure caused by a local credential event error caused an application hang. The logged message included the following errors:

```
ERROR [[...]|CREATE DP|ENABLE|ENABLE PRES DP|LC:Discovery,Security]␣
↪PRESParticipant_enable:FAILED TO POST | An event for [...] validation.
ERROR [[...]|CREATE DP|ENABLE|LC:Discovery,Security] DDS_
↪DomainParticipantPresentation_enable:!create enable
ERROR [[...]|CREATE DP|ENABLE|LC:Discovery] DDS_DomainParticipant_enableI:!
↪enable participant
ERROR [[...]|CREATE DP|ENABLE] DDS_DomainParticipantFactory_create_
↪participant:ERROR: Failed to auto-enable entity
```

Trying to use or delete the *DomainParticipant* caused the application hang.

[RTI Issue ID SEC-2916]

4.4.2 [Major] Excessive resource consumption in certain scenarios *

In certain situations, the *Security Plugins* consumed excessive resources in threads responsible for local security artifacts' renewal and expiration. The primary symptom was high CPU consumption, which slowed *Connext's* runtime execution. Examples of situations triggering this problem included a period of time between creating a *DomainParticipant* and fully enabling it, or disabling OCSP. The latter could be triggered by setting the `<ocsp>` tag within the identity_credential_authority_validation tag to `IGNORED` or by setting the `com.rti.serv.secure.authentication.ocsp_ignored` property to `TRUE`.

[RTI Issue ID SEC-2884]

* *This bug does not affect you if you are upgrading from 7.3.x or earlier.*

4.5 Vulnerabilities

The following vulnerabilities are fixed in this release.

See also RTI Connext Security Bulletins and Advisories for a complete list of vulnerabilities in RTI releases that have been published through the CVE® Program. That list may be more up-to-date.

4.5.1 [Critical] Potential Denial of Service in RTI Security Plugins for OpenSSL when loading Private Key

The *RTI Security Plugins* depended on OpenSSL 3.5.1, which is known to be affected by the CVE-2026-22795 publicly disclosed vulnerability.

This vulnerability has been fixed by upgrading OpenSSL to the latest stable version, 3.5.5.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

- Exploitable invalid or NULL pointer dereference in the *RTI Security Plugins* when loading a private key.
- CVSS v3.1 Base Score: 5.5 Medium
- CVSS v3.1 Vector: CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:N/I:N/A:H
- CVSS 4.0 Base Score: 6.7 Medium
- CVSS 4.0 Vector: CVSS:4.0/AV:L/AC:L/AT:N/PR:N/UI:A/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2895]

4.5.2 [Critical] Potential stack buffer overflow in RTI Security Plugins for OpenSSL when loading Private Key

The *RTI Security Plugins* depended on OpenSSL 3.5.1, which is known to be affected by the CVE-2025-11187 publicly disclosed vulnerability.

This vulnerability has been fixed by upgrading OpenSSL to the latest stable version, 3.5.5.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

- Exploitable stack buffer overflow in the *RTI Security Plugins* when loading a private key.
- CVSS v3.1 Base Score: 6.1 Medium
- CVSS v3.1 Vector: CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:H
- CVSS 4.0 Base Score: 5.2 Medium
- CVSS 4.0 Vector: CVSS:4.0/AV:L/AC:L/AT:N/PR:L/UI:A/VC:L/VI:L/VA:H/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2894]

4.5.3 [Critical] Potential Denial of Service in RTI Security Plugins for OpenSSL when parsing Permissions Document

The *RTI Security Plugins* depended on OpenSSL 3.5.1, which is known to be affected by the CVE-2026-22796 publicly disclosed vulnerability.

This vulnerability has been fixed by upgrading OpenSSL to the latest stable version, 3.5.5.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

- Exploitable invalid or NULL pointer dereference in the *RTI Security Plugins* when parsing a Permissions Document.
- CVSS v3.1 Base Score: 7.5 High
- CVSS v3.1 Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H
- CVSS 4.0 Base Score: 8.7 High
- CVSS 4.0 Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2893]

4.5.4 [Critical] Potential Denial of Service in RTI Security Plugins for OpenSSL when configuring client proxy

The *RTI Security Plugins* depended on OpenSSL 3.5.1, which is known to be affected by the CVE-2025-9232 publicly disclosed vulnerability.

This vulnerability has been fixed by upgrading OpenSSL to the latest stable version, 3.5.5.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

- Exploitable Denial of Service in the *RTI Security Plugins* when the URL of the *DomainParticipant's* OCSP responder has an IPv6 address and you configure a client proxy through the `no_proxy` and `http_proxy` (or `https_proxy`) environment variables. Note that the *RTI Security Plugins* do not support configuring a client proxy for OCSP responders.
- CVSS v3.1 Base Score: 5.9 Medium
- CVSS v3.1 Vector: CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:N/A:H
- CVSS 4.0 Base Score: 8.2 High
- CVSS 4.0 Vector: CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:N/VI:N/VA:H/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2823]

4.5.5 [Critical] A malicious authenticated participant could impersonate other participants

A malicious participant (A) could bypass RTPS origin authentication to impersonate a participant (B). This allowed participant A to send messages to participant B that appeared to be sent by participant B itself. This exploit was only possible if participant B had already successfully completed the authentication process with the participant it intended to impersonate.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

- An authenticated remote attacker within the same domain may publish messages it is not authorized to publish; these messages may look as if they come from the local, legitimate victim *DomainParticipant*.
- CVSS v3.1 Score: 6.5 MEDIUM
- CVSS v3.1 Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:N
- CVSS v4.0 Score: 6.0 MEDIUM
- CVSS v4.0 Vector: CVSS:4.0/AV:N/AC:L/AT:P/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2881]

4.5.6 [Critical] Origin Authentication could be bypassed by crafting malicious RTPS message

When Origin Authentication Protection was enabled, a malicious insider could bypass Origin Authentication Protection by crafting a malicious RTPS message. The only GUID that Origin Authentication Protection validates is the DDS_SampleInfo::publication_handle. Other GUIDs, such as the DDS_SampleInfo::original_publication_virtual_guid and the DDS_SampleInfo::source_guid (which may appear in the inline QoS), are not validated by Origin Authentication Protection. Depending on how the destination *DataReader* interprets these other GUIDs, this behavior could allow the malicious insider to send a message that appears to come from a different *DataWriter*. If the *DataReader* has not set `dds.data_reader.state.filter_redundant_samples` to `0`, then the malicious insider could prevent the *DataReader* from receiving future samples from a legitimate *DataWriter*.

This problem has been fixed by adding a new Permissions Document tag called `proxy`, and a new Governance Document tag called `default_proxy_permissions`. When a *DataWriter* does not have `proxy` permissions, its samples are required to have original_publication_virtual_guid and source_guid equal to publication_handle. The *DataReader* drops any samples that do not meet this requirement. For more information on proxy permissions, see DataWriters Acting as a Proxy for other DataWriters, in the RTI Security Plugins User's Manual 7.7.0.

The issue also affected endpoint discovery. When an endpoint does not have `proxy` permissions, its discovery messages are now required to have a DDS_PublicationBuiltinTopicData::virtual_guid (or DDS_SubscriptionBuiltinTopicData::virtual_guid) equal to the source_guid.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

The impact on *Security Plugins* applications of using the previous version was as follows:

- Exploitable by crafting a malicious RTPS message.
- CVSS v3.1 Base Score: 8.1 HIGH
- CVSS v3.1 Vector: CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:H
- CVSS v4.0 Base Score: 6.1 MEDIUM
- CVSS v4.0 Vector: CVSS:4.0/AV:N/AC:L/AT:P/PR:L/UI:N/VC:N/VI:H/VA:H/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2791]

* *This bug does not affect you if you are upgrading from 7.3.x or earlier.*

Chapter 5

Previous Releases

5.1 What's New in 7.6.0

This section describes what's new in *Security Plugins* 7.6.0, compared to *What's New in 7.5.0*.

Connext 7.6.0 is an early access release (EAR). See Connext Versions and Lifecycle for more information on RTI's software release model.

For what's new in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation. Or start with RTI Connext What's New for a launchpad to all products.

Note: For backward compatibility information between 7.6.0 and previous releases, see the *Migration Guide* on the RTI Community Portal.

5.1.1 Additional Authenticated Data (AAD) enabled by default for Builtin Security Plugins when using PSK or RTPS Header Extension

In the *Builtin Security Plugins*, Additional Authenticated Data (AAD) is now enabled by default for pre-shared key protected communication and for RTPS messages that include an RTPS Header Extension. You can change this default behavior by setting `cryptography.enable_additional_authenticated_data` to a value other than `AUTO`. For details, see RTPS Protection in the *RTI Security Plugins User's Manual*.

Note: This change only applies to the *Builtin Security Plugins;* the *Lightweight Builtin Security Plugins* always use AAD. See *[Major] Lightweight Security Plugins did not use Additional Authenticated Data by default*.

5.1.2 Enforcement of DomainParticipant permissions expiration enhances security

Previously, the *Builtin Security Plugins* only enforced the times provided in the `validity` tag of the Permissions Document upon creation of the local *DomainParticipant* or authentication of remote *DomainParticipants*. Now, permissions are enforced at run-time, as long as Key Revisions are enabled.

When permissions expire with Key Revisions enabled, remote *DomainParticipants* are removed from the system, and local *DomainParticipants* log an error message. You can also configure the local *DomainParticipant* to notify you about the upcoming expiration of the local Permissions Document. For more information, see Permissions Expiration.

5.1.3 Support for runtime extension of Permissions Document validity and enforcement of expiration

The *Builtin Security Plugins* now allow you to extend the validity period of Permissions Documents at runtime. This allows *DomainParticipants* to continue operating without interruption as long as their permissions remain valid. Local *DomainParticipants* propagate the updated Permissions Document to remote *DomainParticipants*. Only changes to the validity dates are supported; the allowed actions cannot be modified.

In addition, this release of the *Builtin Security Plugins* enforces permissions expiration. Together with permissions renewal, it is now possible to implement a short-lived permissions approach that requires *DomainParticipants* to periodically re-validate their permissions to maintain access. This practice enables secure, short-lived permissions in dynamic environments and limits access to required participants only.

5.1.4 Pre-Shared Key Protection now fully aligns with DDS Security 1.2 specification

This release of the *Security Plugins* implements in full the DDS Security 1.2 specification. The following enhancements were applied:

- The passphrase identifier can extend beyond the value of 254, up to 2^32-1 (specification limits apply). For details, see Passphrase Identifier in the *RTI Security Plugins User's Manual*.
- Now you can use extra passphrases configured via the `dds.sec.crypto.rtps_psk_secret_passphrase_extra` property to ensure seamless transition between PSK passphrases without communication interruptions. For details, see Updating the PSK Passphrase Without Message Loss in the *RTI Security Plugins User's Manual*.
- Other AAD (Additional Authenticated Data) related changes. For details, see RTPS Protection in the *RTI Security Plugins User's Manual*, and these related issues:
  - *Additional Authenticated Data (AAD) enabled by default for Builtin Security Plugins when using PSK or RTPS Header Extension* in *What's New in 7.6.0*
  - *[Major] Lightweight Security Plugins did not use Additional Authenticated Data by default* in *What's Fixed in 7.6.0*

5.1.5 Key revisions enabled by default to improve out-of-the-box security

Key revisions are now enabled by default in the *Security Plugins*. Key revisions must be enabled to support certificate and permissions mutability, *DomainParticipant* removal, and OCSP. This setting also ensures that new data cannot be deciphered by *DomainParticipants* that have been removed from the system, while legitimate remote *DomainParticipants* retain access. Enabling key revisions by default makes the *Security Plugins* out-of-the-box security settings stronger.

For more information, see Limiting the Usage of Specific Key Material in the *RTI Security Plugins User's Manual*.

5.1.6 RTI Lightweight Security Plugins support ShangMi cryptographic algorithms

In this release, RTI added full support for ShangMi cryptographic algorithms, consisting of SM3 and SM4 algorithms. This addition enables *Connext* to be used in applications where ShangMi is required by applicable regulations, such as GB 44495. The algorithms are only available in RTI Lightweight Builtin Security Plugins. This error notes the library mismatch but does not stop or prevent creation of *DomainParticipants*. Although *DomainParticipant* creation succeeds, the *Security Plugins* only support a *Connext* Core library with a matching version number.

5.1.7 New error message when Core and Security Plugins library versions differ

If the *Security Plugins* library version is different from the *Connext* Core library version, the *Security Plugins* log the following error message during *DomainParticipant* creation:

```
ERROR [CREATE DP|CREATE DP|LC:Discovery,Security] RTI_Security_PluginSuite_
↪create:!There is a mismatch between the version of the Builtin Security␣
↪Plugins library and the version of the Core library. SP: 1.2.3.4, Core: 7.6.
↪0.0.
```

This error notes the library mismatch but does not stop or prevent creation of *DomainParticipants*.

5.1.8 Upgrade OpenSSL to 3.5.1

The following third-party software used by the *Security Plugins* has been upgraded:

Third party tool	Old version	New version
OpenSSL	3.0.12	3.5.1

In addition to the upgrade, the OpenSSL target packages for QNX and Android now include assembly implementations for cryptographic operations (the packages were built using the `asm_arch` and `perlasm_scheme` options described in this OpenSSL README file). This change affects only the `libcrypto` and `libssl` libraries; the `openssl` executable is still configured with `no-asm`.

For information on third-party software used by *Connext* products, see the "3rdPartySoftware" documents in your installation: <NDDSHOME>/doc/manuals/connext_dds_professional/release_notes_3rdparty and <NDDSHOME>/doc/manuals/connext_dds_secure/release_notes_3rdparty.

For more information on the OpenSSL upgrade, see RTI Security Plugins in the *Migration Guide*.

5.1.9 Deprecations and Removals

Deprecated means that the item is still supported in this release, but will be removed in a future release. *Removed* means that the item is discontinued or no longer supported.

Any deprecations or removals noted in RTI's documentation serve as notice under the Real-Time Innovations, Inc., Maintenance Policy #4220 and/or any other agreements by and between RTI and customer regarding maintenance and support of RTI's software. RTI's current standard terms and support and maintenance policies are available at https://www.rti.com/terms.

Deprecation of authentication.crl and removal of authentication.crl_file

This release of the Security Plugins deprecates the plugin-specific `authentication.crl` property, and removes the plugin-specific `authentication.crl_file` property. Instead, you should set the `dds.sec.auth.crl` property defined in the OMG DDS Security specification. The behavior and configuration of the deprecated and the new property is the same: they are both optional properties that configure the Certificate Revocation List (CRL) used by the *Builtin Security Plugins*.

Support for cryptography.enable_additional_authenticated_data property removed

As described in *[Major] Lightweight Security Plugins did not use Additional Authenticated Data by default*, support for the `cryptography.enable_additional_authenticated_data` property has been removed.

5.2 Whats Fixed in 7.6.0

This section describes bugs fixed in Security Plugins 7.6.0. These are fixes made since *What's Fixed in 7.5.0*.

Security Plugins 7.6.0 is an early access release (EAR). See Connext Versions and Lifecycle for more information on RTI's software release model.

For what's fixed in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation.

[Critical]: System-stopping issue, such as a crash or data loss. [Major]: Significant issue with no easy workaround. [Minor]: Issue that usually has a workaround. [Trivial]: Small issue, such as a typo in a log.

5.2.1 Discovery

[Critical] No communication when using Pre-Shared Key Protection if remote participant did not propagate PID_DOMAIN_ID during discovery

DomainParticipants using Pre-Shared Key Protection did not communicate if the remote *DomainParticipant* did not propagate the `PID_DOMAIN_ID` parameterID (`0x000f`) as part of participant discovery. When this issue triggered, you would see errors like the following as *Security Plugins* tried to decode the pre-shared key protected messages.

```
ERROR [PARSE MESSAGE|[...]{Domain=xxx}|DECODE RTPS message from [...
↪]|LC:Security] RTI_Security_Cryptography[Lightweight]_decode:{
↪"DDS:Security:LogTopicV2":{[...]},{"plugin_method":"RTI_Security_
↪Cryptography[Lightweight]_decode"}]}],"m":"DecryptFinal failed. Possible␣
↪GCM authentication failure.[...]"}}
```

This issue affected both the *Lightweight Builtin Security Plugins* and the *Builtin Security Plugins*, when communicating with non-RTI *DomainParticipants*.

[RTI Issue ID SEC-2755]

5.2.2 Discovery and Authentication

[Critical] Authenticated participant could not replace unauthenticated one if QoS changed during authentication

Regarding the Governance Document tag `allow_unauthenticated_participants`, the DDS Security 1.2 specification states the following: :

```
Additionally, a DomainParticipant that later authenticates would kick out the␣
↪unauthenticated DomainParticipant if it has the same GUID.
```

The *Security Plugins* implemented this behavior in *Connext* 7.3.0, but the following scenario did not work as expected:

- ParticipantA discovers ParticipantC, which is unauthenticated.
- Then, ParticipantA discovers ParticipantB, which has the same GUID as ParticipantC.
- While ParticipantB is in the middle of being authenticated, ParticipantA receives a QoS change from ParticipantC.

In this scenario, when ParticipantB was about to finish being authenticated, ParticipantA incorrectly failed to authenticate ParticipantB and generated a log message similar to this one:

```
{"plugin_method":"PRESParticipant_authorizeRemoteParticipant"}]}],"m":
↪"authenticated trusted state remote participant aaaaaaaa.bbbbbbbb.cccccc␣
↪does not match the one received through discovery by local participant␣
↪dddddddd.eeeeeeee.ffffffff, removing participant...
```

Now, a QoS change by ParticipantC does not interfere with the authentication of ParticipantB.

[RTI Issue ID SEC-2738]

5.2.3 Usability

[Minor] Using properties "dds.license.license_string" or "dds.license.license_file" led to DomainParticipant creation failure

The properties `dds.license.license_string` and `dds.license.license_file` were not validated properly. If you used these properties, you would have seen an error like this one during *DomainParticipant* creation failure:

```
DDS_PropertyQosPolicy_validate_plugin_property_suffixes:Unexpected property:␣
↪com.rti.serv.secure.dds.license.license_string. Closest valid property: com.
↪rti.serv.secure.license_string. If you wish to proceed with this property␣
↪name anyway, change 'com.rti.serv.secure.property_validation_action' to
↪'VALIDATION_ACTION_SKIP' or 'VALIDATION_ACTION_WARNING'.
```

This problem only affected versions 6.0.1 and above. Now, *DomainParticipant* creation succeeds if you use either of those properties.

[RTI Issue ID SEC-2689]

5.2.4 Cryptography

[Critical] Repair of dispose messages failed when Key Revision History enabled

When the *DomainParticipant* property `dds.participant.trust_plugins.key_revision_max_history_depth` was set to a non-zero value, and the *DomainParticipant* contained a reliable *DataWriter* that called the `dispose()` function, the reliability protocol's attempt to repair the dispose message could have failed with the following error:

```
{"plugin_method":"RTI_Security_Cryptography_reencodeSerializedDataWithParams"}
↪]}],"m":"failed to get decode crypto state for local serialized data"}}
```

Dispose message repairs now succeed as expected.

[RTI Issue ID SEC-2746]

5.2.5 Dynamic Participant Renewal, Revocation, and Expiration

[Minor] Local DomainParticipant did not detect more recent expiration date in updated Identity Certificate

After updating the Identity Certificate of a *DomainParticipant* to one with a more recent "Not After" expiration date, only remote *DomainParticipants* accounted for the updated expiration date and banished the expired *DomainParticipant* accordingly.

The local *DomainParticipant* still notified the user about the upcoming expiration on the previous "Not After" date. The updated Identity Certificate would expire without a user notification from the *Security Plugins*.

Now, the local *DomainParticipant* correctly recognizes the new expiration date when notifying about certificate expiration.

Though both local and remote *DomainParticipants* now work as expected, RTI recommends using a later expiration date any time you update an Identity Certificate.

[RTI Issue ID SEC-2691]

5.2.6 Crashes

[Critical] Segmentation fault after creating and immediately deleting a DomainParticipant

A segmentation fault in the internal function `RTIEventActiveGeneratorThread_loop` may have occurred after creating, then immediately deleting, a *DomainParticipant* that used the *Security Plugins*. This segmentation fault was accompanied by the following log message:

```
Backtrace:
        #4      RTIEventActiveGenerator_delete
```

[RTI Issue ID SEC-2761]

5.2.7 Memory Leaks/Growth

[Major] Memory leak when setting a subject name whitelist while security was not enabled

When setting the *DomainParticipant* property `dds.participant.trust_plugins.subject_name_whitelist` without enabling the *Security Plugins*, a memory leak occurred in the internal function `PRESParticipant_storeMutableProperty`. To prevent this issue, setting `dds.participant.trust_plugins.subject_name_whitelist` is now allowed only when the *Security Plugins* are enabled. Otherwise, `DDS_DomainParticipantFactory_create_participant` returns NULL and `DDS_DomainParticipant_set_qos` returns `DDS_RETCODE_NOT_ALLOWED_BY_SECURITY`.

[RTI Issue ID SEC-2731]

[Minor] Memory leak when dds.participant.trust_plugins.expiration_advance_notice_duration.sec property value was invalid

A memory leak occurred when *DomainParticipant* creation failed, the `<enable_key_revision>` governance rule was set to `TRUE`, and the `dds.participant.trust_plugins.expiration_advance_notice_duration.sec` property value was configured incorrectly.

[RTI Issue ID SEC-2751]

5.2.8 OMG Specification Compliance

[Trivial] RTI Lightweight Security Plugins propagated incorrect supported symmetric cipher algorithms value

The OMG DDS Security 1.2 specification requires PSK-only Plugins to propagate that they support a single symmetric cipher algorithm equal to the required algorithm and configured with the `dds.sec.crypto.rtps_psk_symmetric_cipher_algorithm` property. The *RTI Lightweight Security Plugins* instead propagated that they supported several symmetric cipher algorithms (all algorithms that their implementation supported). The *RTI Lightweight Security Plugins* behavior now matches the specification.

[RTI Issue ID SEC-2765]

5.2.9 Interoperability

[Major] Lightweight Security Plugins did not use Additional Authenticated Data by default

The OMG DDS Security 1.2 specification mandates using Additional Authenticated Data (AAD) to protect the integrity of the RTPS Header; however, *Lightweight Security Plugins* used a different protection method by default. This discrepancy could have prevented out-of-the-box communication with other products implementing the DDS Security 1.2 PSK-only standard.

This release makes AAD the only mechanism protecting the integrity of the RTPS Header in *Lightweight Security Plugins*. This protection mechanism cannot be changed. In addition, support for the `cryptography.enable_additional_authenticated_data` property was removed.

For more information, see RTPS Protection in the *RTI Security Plugins User's Manual*.

[RTI Issue ID SEC-2718]

5.2.10 Vulnerabilities

[Critical] Potential heap buffer read overflow in Security Plugins when using a malicious identity certificate file

An out-of-bounds read on the heap could have occurred while processing a malicious identity certificate file. This problem did not affect the *Security Plugins* for wolfSSL.

User Impact without Security

Not applicable (affects a security-specific product/feature only).

User Impact with Security

A vulnerability in the *Connext* application could have resulted in the following:

- Heap buffer overread while processing a malicious identity certificate file.
- Exploitable by overwriting the identity certificate file on the file system with a malicious identity certificate file.
- Exploitable only when a race condition was won.
- Potential impact on confidentiality of *Connext* application.
- Potential crash of *Connext* application.
- CVSS v3.1 Base Score: 7.1 HIGH
- CVSS v3.1 Vector: CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:H
- CVSS v4.0 Base Score: 5.8 MEDIUM
- CVSS v4.0 Vector: CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:H/VI:N/VA:H/SC:N/SI:N/SA:N

[RTI Issue ID SEC-2781]

5.3 What's New in 7.5.0

This section describes what's new in *Security Plugins* 7.5.0, compared to 7.4.0.

Connext 7.5.0 is an early access release (EAR). See Connext Versions and Lifecycle for more information on RTI's software release model.

For what's new in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation. Or start with RTI Connext What's New for a launchpad to all products.

Note: For backward compatibility information between 7.5.0 and previous releases, see the *Migration Guide* on the RTI Community Portal.

5.3.1 Key revisions now enabled in a manner compliant with the DDS Security specification

In release 7.3.0, the *Security Plugins* introduced a mechanism to regenerate and redistribute the Key Material without needing to recreate the involved *DomainParticipants* or lose liveliness. This mechanism (described in Limiting the Usage of Specific Key Material) was enabled with the *DomainParticipant* property `dds.participant.trust_plugins.key_revision_max_history_depth`.

In release 7.5.0, this RTI-specific property is now merely used to configure a resource limit, while a new Governance Document Domain Rule, called `enable_key_revision`, now controls enabling the mechanism. This Domain Rule is part of the OMG DDS Security 1.2 specification.

If you previously set the `dds.participant.trust_plugins.key_revision_max_history_depth` property to a value other than `0` in your *DomainParticipantQos*, now you must also set the `enable_key_revision` Domain Rule to `true` in your Governance Document (see the Migration Guide for an example of how to do this). Then you must re-sign the Governance Document with the Permissions Certificate Authority.

See enable_key_revision in the *RTI Security Plugins User's Manual* for more information.

5.3.2 Support for kicking participants off a system that has non-RTI Participants

As described in Limiting the Usage of Specific Key Material, the *Security Plugins* support a mechanism to regenerate and redistribute Key Material without needing to recreate the involved *DomainParticipants* or lose liveliness. This key regeneration and redistribution is necessary to truly kick participants off the system. However, when the *Security Plugins* 7.0.0 was released, the redistribution of Key Material had not yet been standardized by the OMG DDS Security specification.

Now it has been standardized in the DDS Security 1.2 specification. This release uses the standard protocol for key redistribution when communicating with non-RTI *DomainParticipants*. Note that the protocol for key redistribution when communicating with RTI *DomainParticipants* has not changed, so this feature does not impact backward wire compatibility.

To further comply with the DDS Security 1.2 specification, the valid range of values of `dds.participant.trust_plugins.key_revision_max_history_depth` has changed from `[7, 1000000]` to `[8, 1000000]`. This change impacts backward configuration compatibility.

5.3.3 Support for Online Certificate Status Protocol

This release adds support for the Online Certificate Status Protocol (OCSP). If the *DomainParticipant*'s identity certificate's `authorityInfoAccess` extension field contains an OCSP Responder URL, the *DomainParticipant* will periodically validate its identity status by contacting the OCSP Responder. If the status is valid, the responder provides the *DomainParticipant* with a stapling token. The stapling token is a proof of identity validity that the *DomainParticipant* shares with other *DomainParticipants*.

This a new mechanism for determining the validity of the certificate and offers another (in addition to CRL) possibility for revoking the certificate before its expiration.

See Considerations when enabling OCSP in the *RTI Security Plugins User's Manual* for more information.

5.3.4 New Governance configuration for CRL and OCSP

This release adds a new domain-level rule in the Governance document to govern the use of identity credential validation mechanisms (currently CRL and OCSP) on a domain level. The new configuration enables you to either mandate, disable or leave as optional (by using the default `AUTO` value) the use of a certain mechanism. You can set this rule within the `identity_credential_authority_validation` tag under the existing `domain_rule`.

5.3.5 Improved compatibility of C++ applications using RTI Security Plugins SDK

This release improves compatibility between the *RTI Security Plugins SDK* code and C++ applications. Before this change, if you tried to include *Security Plugins SDK* functions from a C++ application, references may not have been found due to name-mangling issues.

5.3.6 Deprecations and Removals

HMAC-Only mode removed from Builtin Security Plugins

This change was made in release 7.4.0, but not documented at that time.

The *Security Plugins* included the HMAC-Only mode in previous releases. This mode protected RTPS messages with a user-provided HMAC Key, while disabling all other security features (authentication, access control, and encryption). The 7.1.0 release of the *Security Plugins* deprecated the HMAC-Only mode in favor of the Lightweight Builtin Security Plugins.

Starting with release 7.4.0, the Builtin Security Plugin's HMAC-Only mode is removed. Use the Lightweight Builtin Security Plugins instead. *RTI Routing Service* 7.3.0 can act as a bridge between legacy systems using HMAC-Only mode and modern systems using Pre-Shared Key Protection.

5.4 What's Fixed in 7.5.0

This section describes bugs fixed in Security Plugins 7.5.0. These are fixes made since *What's Fixed in 7.4.0*.

Connext 7.5.0 is an early access release (EAR). See Connext Versions and Lifecycle for more information on RTI's software release model.

For what's fixed in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation.

[Critical]: System-stopping issue, such as a crash or data loss. [Major]: Significant issue with no easy workaround. [Minor]: Issue that usually has a workaround. [Trivial]: Small issue, such as a typo in a log.

5.4.1 Discovery and Authentication

[Critical] Unable to communicate with DomainParticipants that propagate 1.0 class_id and use P384 or experimental algorithms

Starting in release 7.4.0, *DomainParticipants* propagate by default the `DDS:Auth:PKI-DH:1.2` **class_id**. This change was made to conform with the DDS Security 1.2 Specification. However, in some scenarios, the update was not backwards compatible with *DomainParticipants* still propagating the `DDS:Auth:PKI-DH:1.0` **class_id**. Specifically, backward interoperability was broken for the P384 or experimental digital signature (`ECDSA+P384+SHA384`, `EDDSA+ED25519+SHA512`, `EDDSA+ED448+SHAKE256`) and key establishment algorithms (`ECDHE-CEUM+P384`, `ECDHE-CEUM+X25519`, `ECDHE-CEUM+X448`). Discovery would not complete and the `unexpected key establishment algorithm` or `unexpected digital signature and digest type combination` messages were logged.

[RTI Issue ID SEC-2649]

5.4.2 Dynamic Participant Renewal, Revocation, and Expiration

[Critical] Loss of data when banishing more than 16,777,208 times

Consider the following scenario:

- The *DomainParticipantQos* property `dds.participant.trust_plugins.key_revision_max_history_depth` is set to a value other than 0.
- The Governance Rule `<data_protection_kind>` is set to a value other than `NONE` for a given topic.
- A given *DomainParticipant* banishes participants (e.g., by calling the API function `banish_ignored_participants`) more than 16,777,208 times but less than 16,777,215 times.
- A *DataWriter* of the given topic within this *DomainParticipant* has historical samples in its queue that were written using much older revisions.
- A *DataReader* of the given topic joins late and tries to receive these historical samples.

When this scenario occurred in previous releases, the *DataReader* would fail to get these historical samples. This log message accompanied this failure:

```
PRESReaderQueueSecurity_decode:FAILED TO DECODE | Incoming serialized data.
```

[RTI Issue ID SEC-2613]

5.4.3 Crashes

[Critical] Log messages possibly corrupted or lost due to memory reordering on weak memory architectures

On weak memory architectures (such as Arm), memory reordering negatively impacted the logging queue used for distributing security events and logging over DDS.

If the queue became full (which can happen if your application generates many log messages, such as when debug verbosity is set), some log messages may have been lost or corrupted.

If the queue was empty and a new message was produced, it could have caused a crash or an incomplete message being published.

These issues have been resolved by adding memory barriers to the logging queue implementation.

[RTI Issue ID SEC-2587]

[Critical] NULL dereference when simultaneously changing certificate or CRL file while deleting DomainParticipant

Suppose you had set `files_poll_interval` to a value other than 0. If you changed the contents of an identity certificate file or a CRL file while deleting the *DomainParticipant*, a race condition would have occurred because those two operations were not thread-safe with respect to each other. This race condition led to a NULL dereference in release libraries and a precondition failure in debug libraries in the internal function `DDS_DomainParticipantGlobals_get_worker_per_threadI`.

[RTI Issue ID SEC-2618]

[Critical] Segmentation fault risk with concurrent updates of identity certificates

There was the risk of a segmentation fault with concurrent updates of identity certificates. It was possible to cause a *DomainParticipant* crash when an identity certificate was updated via different avenues at the same time: that is, via identity certificate file change and via the `set_qos` API.

[RTI Issue ID SEC-2676]

5.4.4 Memory Leaks/Growth

[Critical] Unbounded memory leak if identity certificate file was polled and emptied repeatedly

Consider the following scenario:

- The `dds.sec.auth.identity_certificate` property is set to a file on the file system.
- The `files_poll_interval` property is unset or set to a value other than 0.
- After you create the *DomainParticipant*, the certificate file on the file system is emptied.

In this scenario, the *Builtin Security Plugins* will correctly fail to read the updated certificate file and print this error:

```
ERROR RTI_Security_URIHelper_getStringBuiltin:!failed to read from file
```

When this scenario occurred in previous releases, the problem was that a memory leak of one byte accompanied this error. Repeatedly "touching" (i.e., updating the "last modified" timestamp of) this empty file resulted in repeated one-byte memory leaks.

The unbounded memory leak only affected versions 7.2.0, 7.3.0 to 7.3.0.2, and 7.4.0.

[RTI Issue ID SEC-2525]

5.5 What's New in 7.4.0

5.5.1 Improve isolation in large-scale secure systems using domain tags

Domain tags are now allowed in the rules of your Governance and Permissions Documents. Besides the existing numeric and mandatory `<id>` or `<id_range>` values, the *Security Plugins* now support the optional, string-based `<tag>` and `<tag_expression>` in the `<domains>` element of the Governance Document. The allow and deny rules of the Permissions Document can also have `<tag>` and `<tag_expression>` as part of their `<domains>`. A rule will only apply to a *DomainParticipant* if its domain ID and tag both match the rule.

5.5.2 Adapt permissions at run-time with new support for participant partitions

Starting in this release, a rule in the Permissions Document can match depending on the *DomainParticipant* partitions. Previously, the `<partitions>` tag could only appear within a `<publish>` or `<subscribe>` element. It indicated the *Publisher* or *Subscriber* partitions necessary for a match. Now, the `<partitions>` tag can be a direct child of the `<allow_rule>` and `<deny_rule>` elements. If so, it specifies the *DomainParticipant* partitions required for a match (instead of the *Publisher/Subscriber* partitions).

5.5.3 Simplify your permissions by matching a rule with a group of identities

The *Security Plugins* now support the `<subject_name_expression>` element in the Permissions Document. This tag is an alternative to the `<subject_name>` element. The new tag allows binding the permissions in a `<grant>` section with a *group* of identities, rather than a specific one.

The `<subject_name_expression>` element has the same attribute-value assertions as the `<subject_name>`, but the values can have special characters (such as '*' or '?') used to indicate patterns. For more information, see Rules for matching a subject name in the Permissions Document, in the *RTI Security Plugins User's Manual*.

5.5.4 Inactive Secure Key Exchange DataReader now implies inactive DomainParticipant

If your participant discovery and authentication channels were working but your Secure Key Exchange channel was not working for a given remote participant, you previously had to choose between two options:

- Set `DomainParticipantQos.discovery_config.secure_volatile_writer.max_heartbeat_retries` to a small number and risk entering an unrecoverable situation as described in What's New in 6.1.0.
- Set `DomainParticipantQos.discovery_config.secure_volatile_writer.max_heartbeat_retries` to unlimited and risk an unbounded memory growth on the Secure Key Exchange DataWriter due to samples never getting purged.

Starting in this release, you no longer have to choose between the two options. You can have the best of both options.

If a Secure Key Exchange *DataWriter* marks a Secure Key Exchange *DataReader* as inactive (due to a lack of response within the `max_heartbeat_retries` number of periodic heartbeats), and if `allow_unauthenticated_participants = FALSE`, then the *DataWriter's DomainParticipant* will remove the *DataReader's DomainParticipant* as if liveliness were lost, thus making the "unrecoverable situation" recoverable. If `allow_unauthenticated_participants = TRUE`, then the remote participant will be removed or treated as unauthenticated depending on the status of the Secure Key Exchange *DataReader*. See Configuring Reliability Protocol Settings of the Secure Key Exchange Topic, in the *Security Plugins User's Manual* for more information.

5.5.5 Removed risk of unbounded memory growth on Secure Key Exchange DataWriter

A *DataReader* is marked as *inactive* when it does not respond within the `max_heartbeat_retries` number of periodic heartbeats. This means the *DataWriter* will not wait for the *DataReader* to send an ACK/NACK before removing DDS samples.

The default value for `max_heartbeat_retries` has been decreased from UNLIMITED to 800 for the secure volatile channel. This change removes the risk of unbounded memory growth on the Secure Key Exchange DataWriter due to samples never being purged.

5.5.6 Authentication class_id now conforms to the value in the DDS Security 1.2 Specification

The *Builtin Security Plugins* now propagate the `DDS:Auth:PKI-DH:1.2` **class_id by** default, instead of the previous value of `DDS:Auth:PKI-DH:1.0`. The change is done to conform with the DDS Security 1.2 Specification. *DomainParticipants* still propagate the `DDS:Auth:PKI-DH:1.0` **class_id** to remote *DomainParticipants* from previous RTI releases (7.3.0 or older *Connext Professional* releases, and *Connext Micro* 3.0.3 release).

5.5.7 New standard property to configure shared secret algorithm, to comply with specification

This release of the *Security Plugins* introduces the `dds.sec.auth.key_establishment_algorithm` property, as defined in the DDS Security 1.2 specification. The new property deprecates the plugin-specific property `authentication.key_establishment_algorithm`.

Both properties configure the algorithm used to establish a shared secret during authentication. Their behavior and accepted values are the same. You should use the new standard property instead of the deprecated, plugin-specific one.

5.5.8 Deprecations and Removals

Deprecated means that the item is still supported in this release, but will be removed in a future release. *Removed* means that the item is discontinued or no longer supported.

Any deprecations or removals noted in RTI's documentation serve as notice under the Real-Time Innovations, Inc., Maintenance Policy #4220 and/or any other agreements by and between RTI and customer regarding maintenance and support of RTI's software. RTI's current standard terms and support and maintenance policies are available at https://www.rti.com/terms.

authentication.key_establishment_algorithm property deprecated

As described in *New standard property to configure shared secret algorithm, to comply with specification*, the plugin-specific property `authentication.key_establishment_algorithm` has been deprecated.

5.6 What's Fixed in 7.4.0

This section describes bugs fixed in Security Plugins 7.4.0. These are fixes made since 7.3.0 LTS.

Security Plugins 7.4.0 is an early access release. See the Connext Versions and Lifecycle page for more information on RTI's software release model.

For what's new and fixed in other products in the *Connext* suite, see those products' release notes on the RTI Community Portal or in your installation.

[Critical]: System-stopping issue, such as a crash or data loss. [Major]: Significant issue with no easy workaround. [Minor]: Issue that usually has a workaround. [Trivial]: Small issue, such as a typo in a log.

5.6.1 Discovery and Authentication

[Critical] Unable to communicate with DomainParticipants that propagate a different authentication class_id version

Secure *DomainParticipants* that had different versions in the **class_id** of their Authentication plugin failed to communicate.

For example, if a *DomainParticipant* following the 1.0 DDS Security specification discovered and tried to authenticate a remote *DomainParticipant* following version 1.2 of the specification, you would see a log message similar to the following:

```
RTI_Security_Authentication_begin_handshake_reply: received unexpected
handshake message, probably from a participant that this one lost liveliness
with before ongoing authentication completed. Once this participant sends an
authentication request, communication should be restored.
ERROR [[...]|RECEIVE SAMPLE|PROCESS HANDSHAKE|BEGIN HANDSHAKE␣
↪REPLY|LC:Security]
DDS_DomainParticipantTrustPlugins_forwardBeginHandshakeReply:FAILED TO␣
↪VALIDATE
| begin_handshake_reply returned VALIDATION_FAILED.
```

This release fixes the issue, so a difference in the minor version of the **class_id** doesn't prevent communication.

Note that the *Builtin Security Plugins* 7.4.0 (and later releases) propagate the older `DDS:Auth:PKI-DH:1.0` **class_id** to former RTI releases. This ensures interoperability between RTI releases, even if the older ones are affected by the **class_id** bug. Interoperability is only broken between releases affected by this bug and other vendors that propagate the newer `DDS:Auth:PKI-DH:1.2` **class_id**.

[RTI Issue ID SEC-2473]

5.6.2 Logging

[Minor] Setting logging.verbosity to NOTICE did not list provider information

Setting the `logging.verbosity` property to `NOTICE` failed to show the `RTI_Security_CryptoLibAdapter_listProviderInfo` output described in Verifying the Usage of Providers, in the RTI Security Plugins User's Manual.

One workaround was to first create and delete a dummy *DomainParticipant* with this property set to `NOTICE`. Creating subsequent *DomainParticipants* would log the expected output. This workaround is no longer necessary.

[RTI Issue ID SEC-2404]

[Trivial] Security category missing in some security-related log messages

Some messages related to security were missing the Security category in the Activity Context. For example, a message similar to the following would be logged after a failure due to the permissions of a remote endpoint:

```
ERROR [[...]|LC:Discovery] RTI_Security_AccessControl_check_create_
↪datawriter:[...]
```

Instead of:

```
ERROR [[...]|LC:Discovery,Security] RTI_Security_AccessControl_check_create_
↪datawriter:[...]
```

These security-related messages now have the Security category as part of the Activity Context.

[RTI Issue ID SEC-2387]

5.6.3 Access Control

[Trivial] Grant without a subject name silently ignored when parsing Permissions Document

The Permissions Document consists of a series of grants, and each one must have an associated subject name. Parsing the Permissions Document must fail if it has a grant without a subject name. Here is an example of a grant without a subject name:

```
<grant name="Participant_ecdsa01Peer01">
   <validity>
      <not_before>2020-06-30T13:00:00</not_before>
      <not_after>2030-06-30T13:00:00</not_after>
   </validity>
   <default>ALLOW</default>
</grant>
```

However, in previous releases, the *Security Plugins* simply ignored the grant. While the *Security Plugins* correctly prevented *DomainParticipant* creation or discovery from succeeding, they allowed the *DomainParticipant* creation or discovery process to proceed further than expected, which resulted in this error message incorrectly being printed:

```
XML file doesn't contain a grant for subject name
```

Now, if a grant is missing a subject name, *DomainParticipant* creation or discovery will fail with the following error message:

```
RTI_Security_PermissionsCfgFileParser_parse:!parse XML file.
Grant must contain <subject_name> or <subject_name_expression>
```

[RTI Issue ID SEC-2455]

5.6.4 Crashes

[Critical] Java: segmentation fault while enabling a DataWriter that protects payloads and enables batching

Consider the following scenario:

- You are using the Java API.
- The Governance Document has `<data_protection_kind>` set to a value other than `NONE` for a given topic.
- The `DataWriterQos` has `batch.enable` set to `true`.

Attempting to enable a *DataWriter* of that topic and with that QoS would fail with a segmentation fault in the internal function `PRESTypePluginDefaultEndpointData_calculateBatchBufferSize`. This problem only affected releases 7.0.0 to 7.3.0.

[RTI Issue ID SEC-2457]

[Critical] Parsing malformed protected RTPS packets may have caused bus error on some platforms

If `rtps_protection_kind` was set to a value other than `NONE` in the Governance Document, parsing malformed RTPS packets may have caused a bus error on some platforms, such as a QNX system running on an Arm v7 processor.

[RTI Issue ID SEC-2435]

[Critical] Invalid read when simultaneously changing identity certificate file and validating local DomainParticipant's identity status

Suppose you had set `files_poll_interval` to a value other than 0. If you changed the contents of your identity certificate file while validating the identity status of the local *DomainParticipant*, a race condition would have occurred because those two operations were not thread-safe with respect to each other.

The time to validate the local identity status depends on the `dds.participant.trust_plugins.certificate_expiration_advance_notice_duration.sec` property. If the property value is 0, validation takes place at the expiration date of the identity certificate. Otherwise, it occurs `dds.participant.trust_plugins.certificate_expiration_advance_notice_duration.sec` seconds in advance, and then once every `dds.participant.trust_plugins.certificate_expiration_advance_notice_reminder_period.sec` seconds (1 second by default).

This race condition led to the reading of invalid memory. A memory-checking tool such as Valgrind™ would have reported invalid reads in a function due to accessing an address freed by a different function.

Your application may have crashed with messages such as these:

```
[...]
ERROR [[...]|SET LISTENER] RTI_Security_Authentication_
↪validateCertificateChain:{[[...],{"plugin_class":"DDS:Auth:PKI-DH"},{
↪"plugin_method":"RTI_Security_Authentication_validateCertificateChain"}]}],
↪"m":"Identity verification failed. Make sure it was signed by the right␣
↪authority."}}
ERROR [[...]|SET LISTENER] RTI_Security_CertHelper_logMessageForEveryCa:{[[...
↪],{"plugin_class":"RTI:Common"},{"plugin_method":"RTI_Security_CertHelper_
↪logMessageForEveryCa"}]}],"m":"Failed to verify identity. Used authority: /
↪C=US/ST=CA/[...]"}}
../crypto/asn1/tasn_utl.c:101: OpenSSL internal error: refcount error
fish: Job 1, '[...]' terminated by signal SIGABRT (Abort)
```

[RTI Issue ID SEC-2407]

[Critical] Invalid read when simultaneously changing CRL file and validating remote DomainParticipant's identity

Suppose you had set `files_poll_interval` to a value other than 0. If you changed the contents of your CRL file while validating the identity of a remote DomainParticipant, a race condition would have occurred because those two operations were not thread-safe with respect to each other. This race condition led to the reading of invalid memory. A memory-checking tool such as Valgrind™ would have reported invalid reads in the function `RTI_Security_CertHelper_verifyCertificate()` due to accessing an address freed by a different function `RTI_Security_Authentication_updateCrl()`.

[RTI Issue ID SEC-2479]

[Major] Possible crash when using Python API due to collision with OpenSSL symbols when using certain Python releases

A Python application loading the *Security Plugins* may have crashed with a segmentation fault during *DomainParticipant* creation. You would have seen a message similar to:

```
Thread 1 "python3" received signal SIGSEGV, Segmentation fault.
0x00007fcbc6ff19d5 in OPENSSL_sk_num () from third_party/libcrypto.so.3
[...]
```

The crash was due to a conflict between the OpenSSL symbols in the library shipped with *Connext* and those embedded in the Python executable itself. The *Security Plugins* now look up OpenSSL symbols first within the library, before looking in the global namespace.

To see if your Python release is affected, look for OpenSSL symbols in the Python executable. For example, run the following command on a Linux system:

```
$> nm -D <python_exe> | grep UI_create_method
0000000000377fe0 T UI_create_method
```

If the output is not empty, it means the Python executable contains OpenSSL symbols and they will collide with the ones shipped with *Connext*.

[RTI Issue ID SEC-2376]

5.6.5 Hangs

[Critical] Potential uncontrolled CPU usage when parsing Governance or Permissions Documents due to a vulnerability in Expat

The *Security Plugins* rely on the Core Libraries to parse Governance and Permissions Documents. The Core Libraries, in turn, rely on Expat (third-party software) to parse XML documents. Expat was known to be affected by CVE-2023-52425. This bug has been fixed by upgrading Expat from version 2.4.8 to version 2.6.2. The impact on *Connext* applications of using the previous version was as follows:

- The attacker could exploit the bug by having the victim's Permissions CA sign a Governance or Permissions Document with a large token. This exploitation requires high privileges.
- To exploit a Governance Document, the attacker would need access to the victim's file system.
- To exploit a Permissions Document, the attacker would need access to the victim's network.
- The application could use an uncontrollable amount of CPU and fall victim to a denial-of-service attack.

[RTI Issue ID SEC-2413]

5.6.6 Vulnerabilities

[Critical] Heap buffer overflow while creating a DomainParticipant that uses a malformed Identity Certificate

The *Security Plugins* were affected by a heap buffer overflow vulnerability that occurred while creating a *DomainParticipant* that used a malformed Identity Certificate.

User Impact without Security

None.

User Impact with Security

The impact on *Security Plugins* applications of using the previous version was as follows:

- Exploitable by overwriting the Identity Certificate on the file system with a malicious Identity Certificate that does not even need to be properly signed by the Identity CA.
- The application could have experienced a heap buffer overwrite during creation of a new *DomainParticipant* that attempted to use the malicious Identity Certificate, impacting the integrity and availability of the application.
- This problem was much easier to exploit on a 32-bit architecture. On a 64-bit architecture, this problem affected only the *Security Plugins* for OpenSSL, not wolfSSL.

- The problem was only exploitable if the *authentication.propagate_simplified_identity_certificate* property was unset or set to *TRUE*.
- CVSS Base Score: 7.1 HIGH
- CVSS v3.1 Vector: AV:L/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:H

[RTI Issue ID SEC-2444]

Chapter 6

Known Issues

Note: For an updated list of critical known issues, see the Critical Issues List on the RTI Customer Portal at https://support.rti.com.

6.1 No Support for ECDSA-ECDH with Static OpenSSL Libraries and Certicom Security Builder

If you are using the Certicom® Security Builder® engine, you cannot use the ecdsa-ecdh shared secret algorithm together with static OpenSSL libraries. If you want to use ecdsa-ecdh with Certicom Security Builder, you must use dynamic OpenSSL libraries. Attempting to use ecdsa-ecdh with static OpenSSL libraries and Certicom Security Builder will cause the following errors during participant discovery:

```
Authentication_compute_sharedsecret:failed to provide remote DP public key

Authentication_process_handshake:key generation fail

Authentication_get_shared_secret:empty secret

PRESParticipant_authorizeRemoteParticipant:!security function get_shared_
↪secret
```

6.2 No Support for Writing >65kB Unfragmented Samples Using Metadata or RTPS Message Protection

The following use case is not supported:

- **metadata_protection_kind** = SIGN or ENCRYPT or **rtps_protection_kind** = SIGN or ENCRYPT
- **message_size_max** > 65536. This is possible when using the TCP transport.
- The user is writing unfragmented samples of size greater than 65kB but less than **message_size_max**.

In order to write the large sample, you must set **message_size_max** to be smaller than the message size, so the sample can be put in fragments smaller than 65 kB.

[RTI Issue ID SEC-768]

6.3 subscription_data and publication_data in check_local_datawriter_match / check_local_datareader_match are not Populated

When calling **check_local_datawriter_match / check_local_datareader_match**, *Connext* does not set the **subscription_data** and **publication_data** parameters. While this issue has no impact on the DDS Security builtin plugins, it could affect a custom plugin relying on those parameters.

[RTI Issue ID SEC-758]

6.4 relay_only parameter in check_remote_datareader is not Populated

When calling **check_remote_datareader**, *Connext* does not set the relay_only parameter. While this issue has no impact on the DDS Security builtin plugins, it could affect a custom plugin relying on this parameter.

[RTI Issue ID SEC-852]

6.5 'Allow Rule' Patterns Incorrectly do not Allow Subset Patterns in QoS

In the Permissions Document, an <allow_rule> that has a pattern partition other than * (e.g., P*) incorrectly does not allow creation of an entity whose PartitionQosPolicy contains a regular expression pattern that is a subset of that <allow_rule> (e.g., P1*). This problem only affects Security Plugins 6.1.0 and above.

The workaround is to change the <allow_rule>'s pattern partition to exactly match the pattern partition in the QoS (e.g., change P* to P1*).

[RTI Issue ID SEC-1242]

6.6 Lightweight Security Plugins do not support Distributed Logging

The *Lightweight Builtin Security Plugins* do not support logging messages over DDS. Configuring the `com.rti.serv.secure.logging.mode_mask` property to include the `SECURITY_TOPIC` value does not have any effect.

The Builtin Secure Logging Topic (`DDS:Security:LogTopicV2`) is a secure *Topic* that requires protecting both the data and the metadata of RTPS submessages. However, the *Lightweight Builtin Security Plugins*

only support protection at the RTPS level. A different, unprotected, *Topic* would be required for logging messages over DDS. The OMG DDS Security 1.2 specification doesn't describe how the *Lightweight Builtin Security Plugins* should handle distributed logging. This issue is not a bug in the *Lightweight Builtin Security Plugins*, but a result of unspecified requirements in the OMG DDS Security 1.2 specification.

[RTI Issue IDs SEC-2770, SEC-2780, and ADMINCONSOLE-1466]

Chapter 7

Copyrights and Notices

© 2017-2026 Real-Time Innovations, Inc. All rights reserved. April 2026

Trademarks

RTI, Real-Time Innovations, Connext, Connext Drive, NDDS, the RTI logo, 1RTI and the phrase, "Your Systems. Working as one." are registered trademarks, trademarks or service marks of Real-Time Innovations, Inc. All other trademarks belong to their respective owners.

Copy and Use Restrictions

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form (including electronic, mechanical, photocopy, and facsimile) without the prior written permission of Real-Time Innovations, Inc. The software described in this document is furnished solely under and subject to RTI's standard terms and conditions available at https://www.rti.com/terms and in accordance with your License Acknowledgement Certificate (LAC) and Maintenance and Support Certificate (MSC), except to the extent otherwise agreed to in writing by RTI.

Securing a distributed, embedded system is an exercise in user risk management. RTI expressly disclaims all security guarantees and/or warranties based on the names of its products, including Security Plugins and Security Plugins SDK. Visit https://www.rti.com/terms/ for complete product terms and an exclusive list of product warranties.

Third-Party Software

RTI software may contain independent, third-party software or code that are subject to third-party license terms and conditions, including open source license terms and conditions. Copies of applicable third-party licenses and notices are located at community.rti.com/documentation. IT IS YOUR RESPONSIBILITY TO ENSURE THAT YOUR USE OF THIRD-PARTY SOFTWARE COMPLIES WITH THE CORRESPONDING THIRD-PARTY LICENSE TERMS AND CONDITIONS.

Notices

Deprecations and Removals

Deprecated means that the item is still supported in this release, but will be removed in a future release. *Removed* means that the item is discontinued or no longer supported.

Any deprecations or removals noted in RTI's documentation serve as notice under the Real-Time Innovations, Inc., Maintenance Policy #4220 and/or any other agreements by and between RTI and customer regarding maintenance and support of RTI's software. RTI's current standard terms and support and maintenance policies are available at https://www.rti.com/terms.

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/