Edge C SDK
Release Notes

Version 2.2.1
February 2019

ThingWorx Edge C SDK Release Notes

The following sections list and describe the new features and the issues fixed in the releases of the ThingWorx Edge C SDK in chronological order, from the current release to the oldest release.

To download the latest distribution bundle for your platform, visit the Software Downloads page of the PTC eSupport Portal, https://support.ptc.com/appserver/cs/software_update/swupdate.jsp.

C SDK v.2.2.1

This release of the C SDK includes the software fixes from v.2.2.0 plus, as part of PTC's continuous efforts to improve security, a security enhancement. This release provides the following enhancement:

ID (SFID)	Applies To	Description
CSDK-1315 (14593755)	C SDK v.2.2.1	Release 2.2.1 of the C SDK includes non-FIPS OpenSSL 32– and 64–bit libraries, version 1.0.2q, which, on Windows platforms is based on Visual Studio 2015 runtime library. The FIPS OpenSSL libraries of this SDK are still based on Visual Studio 2012 and therefore still are version 1.0.2l of the OpenSSL libraries. In addition, as of this release, the axTLS library is no longer provided in the distribution bundles of the C SDK.

C SDK v.2.2.0

This release of the C SDK includes the following enhancements and software fixes::

ID (SFID)	Applies to	Description
Enhancements		
CSDK-	C SDK 2.2.0 and .NET SDK 5.8.0	Security Enhancement For this release, the way that strings of type PASSWORD are provided to the SDK has changed. This includes Application Keys, digests, passwords for certificates (passphrase for keystore), and passwords for proxy server users. For details about this enhancement, see the *ThingWorx Edge C SDK Developer's Guide* that accompanies the C SDK or

ID (SFID)	Applies to	Description
		the C SDK section of theThingWorx Edge SDKs and ThingWorx WebSocket-based Edge MicroServer Help Center.
CSDK-1160 and C SDK 1302	C SDK 2.2.0	The C SDK has been updated to support Microsoft Visual Studio 2017 in addition to its support for 2012 and 2013.
Issues Fixed in This Release		
CSDK-1165 (13968996)	C SDK 2.2.0, .NET SDK 5.8.0, WS EMS 5.4.5	Sometimes when starting the C SDK, a linking error occurs (0x18) An intermittent linking error occurs when starting the C SDK. CAUSE: The **twCSdk.dll** loads consistently at the address that OpenSSL wants to occupy when this error is encountered. Since OpenSSL is a dependency of the C SDK, which is a dependency of the top level executable, Windows always loads the C SDK dll before it loads OpenSSL. You can see this in an application that allows you to view the dependency order as a 'tree'. WORKAROUND: Make the following code and configuration changes to force `libeay32.dll` be to loaded before the C SDK in your final application: 1. In CMake build files for the C SDK, .NET SDK, and any other application that depends on OpenSSL, `libeay32.dll` should be specified first in the linker order. (Windows only) 2. In the **main()** function of your C or .NET application, you need to make a call to some function that lies in `libeay32.dll`. It can just be a call to get the version, as shown below: `const char * version_test = SSLeay_ version(SSLEAY_VERSION);` These changes force the executable to depend directly on `libeay32.dll`, which makes Windows load it before it loads the C SDK.
CSDK-1175	C SDK 2.2.0 and WS EMS 5.4.5	**"twMap_Add**: parse function returned null" is logged whenever starting WS EMS v.5.4.0 This issue is fixed in this release of the C SDK.

ID (SFID)	Applies to	Description
CSDK-1290	C SDK 2.2.0	C SDK build depends on relative path. This issue is fixed in this release.
CSDK-1291	C SDK 2.2.0	`snprintf` redefinition This issue is fixed in this release. `snprintf`'s are now wrapped in `#ifndef`'s.
CSDK-1319	C SDK 2.1.2 through 2.2.0, WS EMS 5.4.5	**twApi_Connect()** can block and never return in some cases. If no receive timeout value is set on the underlying socket, the **twApi_Connect()** function can sometimes block and never return. The C SDK uses blocking sockets where the edge device must receive a TCP three-way handshake before this function will return. By default, the TCP receive timeout is set to 0, which means that it will never time out. In cases where the remote host never sends any packages to the edge device, the **twApi_ Connect()** function will never time out and block forever. RESOLUTION: Set a receive timeout value on the socket, using the **setsocketopts()** call. For example: ```DWORD timeout = 60000; // 60 seconds in milliseconds setsockopt (t->connection->sock, SOL_ SOCKET, SO_RCVTIMEO, (char *)&timeout, sizeof (timeout));```
CDK-1340	C SDK 2.1.2 through 2.2.0, WS EMS 5.4.5	This issue has been resolved as part of PTC's continued investment in helping customers reduce risks associated with security threats.

C SDK v.2.1.5

This release includes software fixes for the following issue:

ID (SFID)	Applies to	Description
CSDK-1294 (14545563)	C SDK 2.1.6, WS EMS 5.4.5	WS EMS 5.4.1 - **GetDirectoryStructure** returns wrong result when sub-folder is named with 1-digit number and deep sub-folders are not returned. This issue is fixed in this release. **GetDirectoryStructure** will now return the correct result when a sub-folder has a 1-digit or 1-character name.

C SDK v.2.1.4

This release includes software fixes for the following issue:

ID (SFID)	Applies to	Description
CSDK-1278 (14215962 and 14231681)	C SDK 2.1.4	SCM file transfer fails on large files and remains *.part* at destination. This issue is fixed this release. **📝 Note** The `messageChunkSize` must be less than (or equal to) the value of `frameSize`.

C SDK Version 2.1.3

With this release, the C SDK supports the separately available ThingWorx Software Content Management (SCM) Edge Extension (v1.0.0). See the release notes for the SCM Edge Extension for details. To use ThingWorx SCM, you also need the SCM extension for ThingWorx platform (formerly part of ThingWorx Utilities).

ID (SFID)	Applies To	Description
Issues Fixed in This Release		
CSDK-80	C SDK 2.1.3	`messageChunkSize` and `frameSize` (`twApi_Initialize` and `twWs_Create`) have conflicting error logic. This issue is fixed in this release.
CSDK-787	C SDK 2.1.3	`twMap` should not produce an error when a duplicate key is added. The log message, `twMap_Add: key already exists, deleting new entry`, now logs at the DEBUG level of log messages because this behavior is acceptable.

ID (SFID)	Applies To	Description
CSDK-1104	C SDK 2.1.3	CSDK 2.0.0 added an **init** callback that is fired when the SDK is initialized by **twApi_Initialize**. The registration function **twApi_ RegisterInitCallback** for this callback allows the user to provide a `void*` 'user data' pointer, which is cached off in a data structured owned by **twcfg**. However, the user data is not actually passed into the callback when it is invoked. This issue is fixed in this release.
CSDK-1257	C SDK 2.1.3	Circular `include` dependency in C SDK. The `enum LogLevel` definition has been moved from the `twLogger.h` file to the `twDefinitions.h` file. In addition the `#include twLogger.h` line has been removed from the `twOSPort.h` file.

C SDK Version 2.1.2

Issue ID (SFID)	Applies To	Description
Enhancements		
CSDK-1170	C SDK 2.1.2	Improve logging for staging directory failures. A file transfer failed to complete when the staging directory was on a separate partition (RAMDisk) from the final destination. The resolution of this issue was additional logging around the failure (a DEBUG build) to help resolve a file transfer failure in `twListEntities()`.
CSDK-1195	C SDK 2.1.2	Add support to the ThingWorx C SDK to customize what cipher suites are used at runtime. The C SDK now allows developers to customize the cipher suites to use at run time by specifying a string that contains the cipher suite configuration in a new parameter, `cipher_set`, in the `twcfg` structure. For more information, see the section, *Support for Cipher Suites*, in the C SDK section of the ThingWorx Edge SDKs and WS EMS Help Center or in the *ThingWorx Edge C SDK Developer's Guide, v.2.1.2*, which is available in the distribution bundle or through the Reference Documents page of the PTC Support site.

Issue ID (SFID)	Applies To	Description
Issues Fixed		
CSDK-1185	CSDK 2.1.2	Race condition prevents FIPS mode from being used on WebSocket connection. Due to a race condition, calls to `twApi_EnableFipsMode()` would not enable FIPS mode on the WebSocket connection to the ThingWorx platform.
CSDK-1196	C SDK 2.1.2	WebSocket compression header message is formatted incorrectly The WebSocket compression header message was formatted incorrectly, preventing compression from being enabled. This issue is fixed in this release.
CSDK-1217	C SDK 2.1.2	C SDK mishandles incoming fragmented WebSocket messages (continuation frames). The C SDK did not correctly handle incoming WebSocket messages that were fragmented into multiple frames. Instead of concatenating the frames into a single message before attempting to deserialize the AlwaysOn payload, the SDK instead treated the first and all subsequent fragments as whole messages, parsing each of their payloads as individual AlwaysOn messages This issue is fixed in this release. Incoming WebSocket messages that are fragmented into multiple frames are now concatenated into a single message before the SDK attempts to deserialize the payload.
CSDK-1225	C SDK 2.1.2	Compression race condition allows corrupt messages to be sent to the platform. With this release of the C SDK, binary data added to the sliding window compression context is sent out the WebSocket in the exact same order.
CSDK-1226	C SDK 2.1.2	C SDK Tunnel Manager calls incorrect API when setting client CA list. In `twTunnel_Create()` the tunnel manager incorrectly calls `twTlsClient_SetCertificateChain()` when setting the

Issue ID (SFID)	Applies To	Description
		CA Certificate list. This issue prevented the WebSocket tunnel from working correctly when certificate validation was enabled. The tunnel manager now correctly calls `twTlsClient_SetClientCaList()`.
CSDK-1227	C SDK 2.1.2	C SDK fails to send messages when the message serializes as a fragmented WebSocket message. Similar to CSDK-1217, the CSDK would not generate fragmented WebSockets (that is, WebSockets whose message size is greater than the message chunk size, or 8k by default). Rather than generate the fragmented message, it failed to send the whole message. This issue occurred when compression was enabled and file transfers of random or compressed data were being performed from the edge up to the platform. This issue is fixed in this release. File transfers from the edge up to the platform should now complete successfully when compression is enabled and the files contain random or compressed data.
CSDK-1231	C SDK 2.1.2	Manage remote property bindings - fails with large property count with long property names. When doing remote property binding in ThingWorx Composer, no values were displayed for a large property count, where the property names were long. This issue is fixed in this release. The maximum string length (set in `max_string_prop_length`) for property and response messages has been adjusted from 16384 bytes to 1MB, and this value is now configurable.

C SDK Version 2.1.1

Issue ID (SFID)	Applies To	Description
CSDK-1169 (13968996)	C SDK 2.0.5	Offline Message Store Does Not Transfer Multiple Chunks to the Platform The offline message store does not transfer any property value information when the offline message store file size is larger than 8 KB. This issue occurred with C SDK v.2.0.5 and ThingWorx platform v.8.0.3. CAUSE: When creating the large offline messages (>8 KB), the C SDK breaks them up into multi-part messages. When flushing the offline message store, the C SDK is generating new request ID's for each message. The platform cannot sync up the multi-part request because it expects the related messages to have the same request ID. This issue is fixed in this release.

C SDK Version 2.1.0

Issue ID	Applies To	Description
New in This Release		
CSDK-976 / EDGE 2274	C SDK 2.1.0	Enable WebSocket Compression: C SDK See the *ThingWorx Edge C SDK Develoepr's Guide* for more information about compression.
Issues Fixed in This Release		

Issue ID	Applies To	Description
CSDK-1119	C SDK 2.1.0	`twGetFileChecksum` can access freed memory. The `twGetFileChecksum` function, which is called by the platform after a file transfer completes, can access memory that has been freed by another thread, crashing the hosting application. This occasionally happens in normal circumstances when a file transfer is canceled (when it is nearly complete). This issue is fixed in this release.
CSDK-1094	C SDK 2.1.0	Missing `init` Function in Windows Extension Library throws Exception. When an extension library is renamed or built incorrectly, its `init` function can be missing. RESOLUTION: With this release, if this situation occurs, an error message is presented and the call to load the library is terminated gracefully.

C SDK Version 2.0.5

Issue ID	Applies To	Description
CSDK-1148	C SDK 2.0.4, .NET SDK 5.7.4	C SDK fix for .NET SDK 5.7.4. The .NET SDK was observed to not work with OpenSSL, occasionally hanging, and initializaing very slowly. `twApi_ RegisterInitCallback()` as allocating the incorrect memory size from the heap and overwriting neighboring heap space. This callback function now allocates the correct memory size.

C SDK Version 2.0.4

Issue ID	Applies To	Description
CSDK-1146	C SDK 2.0.3	The twMap utility was leaking in v.2.0.3. This issue is fixed in this release.

C SDK Version 2.0.3

Issue ID	Applies to	Description
CSDK-1141	C SDK 2.0.2	axTLS `TW_NEW_SERVER` implementation ignores `use_default_certificate` switch. The `TW_NEW_SERVER` function in twAxTls.h does not support the flag, `use_default_certificate` in the same way that `twOpenSSL.h` does.. This issue is fixed in this release.
CSDK-1125	C SDK 2.0.0	Incorrect comparison operation prevents TLS library information from being printe3d at startup This issue is fixed in this release.

C SDK Version 2.0.2

Issue ID	Applies to	Description
CSDK-1112	C SDK 2.0.0	Causing a disconnect during asynchronous publish prevents value from being published to the platform. A new C SDK API function clears the cached SPM list of `currentValues`on disconnect. This new function is intended to ensure values are published to the platform as expected.
CSDK-850	C SDK 2.0.0	Only one timed-out message is reaped per iteration (every 50 seconds). The function, **twMessageHandler_CleanupOldMessages**, is expected to clean up all messages that have timed out, but it only cleaned up the first message. This issue is fixed in this release. The function, **twMessageHandler_CleanupOldMessages**, now works as expected. All timed-out messages are cleaned up when this function is called.
CSDK-197	C SDK 2.0.0	C SDK leaks memory in `twMap_Create()`. The `twMap_Create()` function allocates memory that was never freed. This issue is fixed in this release.

C SDK Version 2.0.1

Issue ID	Applies to	Description
CSDK-1070	C SDK 2.0.0	Race condition causes the queue size of the `SubscribedPropertyManager` (SPM) to underflow and reject all data updates. The `queueSize` counter was exposed to concurrent access, which caused it to be updated incorrectly. Afterwards, as updates were flushed out of the SPM, the unsigned value underflowed, leading the SDK to reject all incoming property updates with a message that the maximum value for the queue size would be exceeded if the property specified in the error message were added. This issue is fixed in this release.

C SDK Version 2.00

Issue ID (SFID)	Applies to	Description
New in This Release		
CSDK-989	C SDK 2.0.0	To prevent integer overflows when defining connect retries, the **twApi_Connect()** method signature has changed to accept "int16_t retries" instead of "int32_t retries".
CSDK-953	C SDK 2.0.0	Print Version Info on Startup When the C SDK is started it prints the following version information: • Version number of the SDK • SSL library being used — OpenSSL, OpenSSL-FIPS, NoTls, or axTLS • Version number of the SSL library being used. • Whether FIPS is enabled
CSDK-952	C SDK 2.0.0	Create SSL Compatibility Matrix The document, *ThingWorx Edge Requirements and Compatibility Matrix*, for the Edge SDKs, WS EMS, and Connection Services has been updated to include a matrix for C SDK releases and the compatible axTLS and OpenSSL versions.
CSDK-938	C SDK 2.0.0	Support Loading OpenSSL from system libs To load OpenSSL from system libs, set the following environment variables to your OpenSSL `include/lib` directories and compile: • `THINGWORX_CSDK_OPENSSL_INCLUDE_DIR` • `THINGWORX_CSDK_OPENSSL_LIB_DIR`
CSDK-932	C SDK 2.0.0	Distribute OpenSSL in the C SDK Starting with this release, the C SDK distribution bundles for the supported platforms include the appropriate OpenSSL binaries, v. 1.0.2k of OpenSSL.

Issue ID (SFID)	Applies to	Description
		📝 **Note** The C SDK build process will now default to an OpenSSL-based build instead of an axTLS-based build. axTLS has been removed due to issues with SSL/TLS certificate configurations and known bugs related to accepted cipher suites in Tomcat.
Issues Fixed in This Release		
CSDK-911 (13591031)	C SDK 1.3.4	SDKClient Code Does Not Receive the Actual Exception Message Thrown from ThingWorx Service. CAUSE: The SDK client returned a Generic Exception instead of the actual exception Reason thrown in a ThingWorx local service. The SDK should return the actual Reason so that a developer can take action on the error. RESOLUTION: The SDK will now return the actual exception message as thrown in the service. The C SDK function `twApi_InvokeServiceAsync()` invokes a service asynchronously and can be passed a callback. Upon service completion, the the "Reason" field of the response, is passed to the callback, providing a way for an application to react to the exception.
CSDK-891 (13647664)	C SDK 1.3.2.636	C SDK Crashes with Null Pointer Exception in `twList_Create()` Function. If the allocation of the mutex failed, the list was deleted and the list pointer was set to null (0). Then the list->delete_function assignment statement was executed. The list became NULL, causing a crash. This issue is fixed in this release.
CSDK-888 (13338641)	C SDK	C SDK can reuse request ids, which can lead to unexpected behavior. The C SDK was observed sending two requests with the same request id. The first request timed out, while the second request appeared to succeed. Even though the second request was successful, the session id received from ThingWorx platform for the authentication request was invalid (–1). All follow-on communications failed with an invalid session error. CAUSE: When a persisted message is resent, it should get a new request id to avoid collisions. Instead of requesting a new id, the persisted message took an id from an active message and incremented that message's id. So, when a new message requested a new id, a duplicate id was issued. RESOLUTION: The resolution is an interface for all messages to use to get a request id, so that collisions can no longer occur.
CSDK-781	C SDK, v.1.3.0,	Numeric Truncation Error

Issue ID (SFID)	Applies to	Description
	1.4.0, and 1.5.1	CAUSE: *connect_retries* was the same type everywhere EXCEPT for the input to **twApi_Connect()**. RESOLUTION: The **twApi_Connect()** method signature has changed so that it is an `int16_t`. This change will prevent integer overflows when defining connect retries.
CSDK-32	C SDK 1.3.2	`SetGatewayType` and `SetGatewayName` C SDK API Functions Do Not Check If the API Has Been Initialized RESOLUTION: these two functions do check whether the API has been initialized. (that is, they make this check: `if (!tw_api) return TW_NULL_API_SINGLETON`).
Known Issue		
CSDK-1100	C SDK 2.0.0	Linux-ARM cross compilation issue. During testing of C SDK 2.0.0, it was discovered that cross compiling to ARM on systems without hardware floating point was having problems locating the shared libraries used in its tool chain. This issue affects devices that do not support or do not have the resources to do on-board compilations. This issue is a configuration issue with CMake. CMake was introduced in version 1.4.0 of the C SDK.

C SDK Version 1.5.3

Issue ID (SFID)	Applies to	Description
Issues Fixed in This Release		
CSDK-1123	C SDK	Memory leak discovered in C SDK application. This issue is fixed in this release.
Known Issues		
CSDK-916	C SDK	C SDK v.1.4.x and 1.5.x releases use axTLS v.2.1.2. These versions of the C SDK were unable to connect to SSL Tomcat servers that did not specify `pathLenConstraint` in their root certificate (which include PTC Cloud Services production instances of ThingWorx platform). A partial fix for this issue was provided in release 1.5.0 (see CSDK-933 in the notes for that release) for the problem with the root certificate in a chain, where the chain consists of the root, one intermediate certificate (with or without `pathLenConstraint`), and a server certificate (which is not a CA so constraints are irrelevant). However, the problem can still occur if a certificate chain has more than one intermediate certificate that does not have `pathLenConstraint`. For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9.. IMPORTANT! This issue is present only when using axTLS builds of the C SDK v.1.4.x and 1.5.x. OpenSSL-based C SDK builds do not experience this issue.
CSDK-911 (13591031)	C SDK	SDK client code does not receive the actual exception message thrown from ThingWorx service

Issue ID (SFID)	Applies to	Description
		Invoking a service from ThingWorx SDK client returns a Generic Exception and not the actual exception thrown in ThingWorx local service. Without the actual exception, a developer cannot take action based on the exception.
CSDK-891 (13647664)	C SDK	C SDK crashes with null pointer exception in `twList_Create` function. If the allocation of the mutex fails, the list is deleted and the 'list' pointer is set to null (0). The list delete_function assignment statement is then executed. The 'list' will be NULL at this time, causing a crash.
CSDK-888 (13338641)	C SDK	C SDK can reuse request ids, which can lead to unexpected behavior. The C SDK was observed sending two requests with the same request id. Request 1 timed out, and request 2 appeared to succeed. However, the session id received from the platform for the authentication request was invalid, causing all follow-on communication to fail with an invalid session error.
CSDK-115 (13373562)	C SDK	Size of the file limited by `twcfg.offline_msg_queue_size` is not honored. When using the subscribed property manager and a device is offline when calling `twApi_PushSubscribedProperties()`, the property values are written to a persisted file named `subscribed_props.bin`. The size of the file should be limited by the variable `twcfg.offline_msg_queue_size`. However, the actual file will grow considerably larger than the value specified by `twcfg.offline_msg_queue_size`.

C SDK Version 1.5.2

Issue ID (SFID)	Applies to	Description
Issues Fixed in This Release		
CSDK-978	C SDK	NUMBER updates dropped when the previous update's value was 0. For properties of type NUMBER, the C SDK always used `{{0}}` as the incoming value when doing VALUE and DEADBAND delta comparisons. Therefore, if the previous value of a property was `{{0}}`, the next incoming value would not be sent to the platform, resulting in an error message. NOTES: • This issue only affected properties of type NUMBER. • This issue did not affect push type ALWAYS. • For DEADBAND properties, *every update* after the '0' update was ignored

Issue ID (SFID)	Applies to	Description
		This issue is fixed in this release. Updates of DEADBAND properties are no longer ignored after the '0' update.
CSDK-968	C SDK	Removing property with push type ALWAYS causes null pointer de-reference. Unregistering a property callback for a property with push type ALWAYS on a Thing with other properties caused a null pointer de-reference in the stack, crashing an application or corrupting SDK structures. This issue is fixed in this release. The last pointer is now checked, preventing the null pointer de-reference.
Known Issues		
CSDK-916	C SDK	C SDK v.1.4.x and 1.5.x releases use axTLS v.2.1.2. These versions of the C SDK were unable to connect to SSL Tomcat servers that did not specify `pathLenConstraint` in their root certificate (which include PTC Cloud Services production instances of ThingWorx platform). A partial fix for this issue was provided in release 1.5.0 (see CSDK-933 in the notes for that release) for the problem with the root certificate in a chain, where the chain consists of the root, one intermediate certificate (with or without `pathLenConstraint`), and a server certificate (which is not a CA so constraints are irrelevant). However, the problem can still occur if a certificate chain has more than one intermediate certificate that does not have `pathLenConstraint`. For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9.. IMPORTANT! This issue is present only when using axTLS builds of the C SDK v.1.4.x and 1.5.x. OpenSSL-based C SDK builds do not experience this issue.
CSDK-911 (13591031)	C SDK	SDK client code does not receive the actual exception message thrown from ThingWorx service Invoking a service from ThingWorx SDK client returns a Generic Exception and not the actual exception thrown in ThingWorx local service. Without the actual exception, a developer cannot take action based on the exception.
CSDK-891 (13647664)	C SDK	C SDK crashes with null pointer exception in `twList_Create` function. If the allocation of the mutex fails, the list is deleted and the 'list' pointer is set to null (0). The list delete_function assignment statement is then executed. The 'list' will be NULL at this time, causing a crash.

Issue ID (SFID)	Applies to	Description
CSDK-888 (13338641)	C SDK	C SDK can reuse request ids, which can lead to unexpected behavior. The C SDK was observed sending two requests with the same request id. Request 1 timed out, and request 2 appeared to succeed. However, the session id received from the platform for the authentication request was invalid, causing all follow-on communication to fail with an invalid session error.
CSDK-115 (13373562)	C SDK	Size of the file limited by `twcfg.offline_msg_queue_size` is not honored. When using the subscribed property manager and a device is offline when calling `twApi_PushSubscribedProperties()`, the property values are written to a persisted file named `subscribed_props.bin`. The size of the file should be limited by the variable `twcfg.offline_msg_queue_size`. However, the actual file will grow considerably larger than the value specified by `twcfg.offline_msg_queue_size`.

C SDK Version 1.5.1

Issue ID (SFID)	Applies to	Description
Issues Fixed in This Release		
CSDK-944	C SDK	Poor performance publishing to entities with high property counts. With changes made for CSDK-743 and CSDK-837, the `SetPropertyVTQ` function was entering into a linear search through a list of properties for the relevant entity. This linear traversal led to poor performance on entities that had very high property counts (several hundred or more). To resolve this issue, the list used to implement the most-recent value cache has been replaced with a dictionary, which allows lookups to happen without adversely affecting performance (CPU usage). It is important to note that for implementations that have far fewer properties, this issue should not occur with releases 1.4.x or 1.5.0 (up to 100 Things, each with 100 properties, for example).
Known Issues		
CSDK-916	C SDK	C SDK v.1.4.x and 1.5.x releases use axTLS v.2.1.2. These versions of the C SDK were unable to connect to SSL Tomcat servers that did not specify `pathLenConstraint` in their root certificate (which include PTC Cloud Services production instances of ThingWorx platform). A partial fix for this issue was provided in release 1.5.0 (see CSDK-933 in the notes for that release) for the problem with the root certificate in a chain, where the chain consists of the root, one intermediate certificate (with or without `pathLenConstraint`), and a server certificate (which is not a CA so constraints are irrelevant). However, the problem can still occur if a certificate chain has more than one intermediate certificate that does not have `pathLenConstraint`. For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9..

Issue ID (SFID)	Applies to	Description
		IMPORTANT! This issue is present only when using axTLS builds of the C SDK v.1.4.x and 1.5.x. OpenSSL-based C SDK builds do not experience this issue.
CSDK-911 (13591031)	C SDK	SDK client code does not receive the actual exception message thrown from ThingWorx service Invoking a service from ThingWorx SDK client returns a Generic Exception and not the actual exception thrown in ThingWorx local service. Without the actual exception, a developer cannot take action based on the exception.
CSDK-891 (13647664)	C SDK	C SDK crashes with null pointer exception in `twList_Create` function. If the allocation of the mutex fails, the list is deleted and the 'list' pointer is set to null (0). The list delete_function assignment statement is then executed. The 'list' will be NULL at this time, causing a crash.
CSDK-888 (13338641)	C SDK	C SDK can reuse request ids, which can lead to unexpected behavior. The C SDK was observed sending two requests with the same request id. Request 1 timed out, and request 2 appeared to succeed. However, the session id received from the platform for the authentication request was invalid, causing all follow-on communication to fail with an invalid session error.
CSDK-115 (13373562)	C SDK	Size of the file limited by `twcfg.offline_msg_queue_size` is not honored. When using the subscribed property manager and a device is offline when calling `twApi_PushSubscribedProperties()`, the property values are written to a persisted file named `subscribed_props.bin`. The size of the file should be limited by the variable `twcfg.offline_msg_queue_size`. However, the actual file will grow considerably larger than the value specified by `twcfg.offline_msg_queue_size`.

C SDK Version 1.5.0

Issue ID (SFID)	Applies to	Description
New in This Release		
CSDK-894	C SDK	Implement Idle Transfer Monitor in `twApi_TaskerFunction` File transfer timeouts are now evaluated independent of the FileManager singleton. The new idel file transfer monitor is in `twApi.c` (1776) within the `twApi_TaskerFunction`.
CSDK-864	C SDK	Deadband Push Type

Issue ID (SFID)	Applies to	Description
		To support KEPServer users, the C SDK supports a new push type that implements a cumulative Threshold to provide deadband support.. The deadband push type is implemented as absolute (no percentages). It follows the existing threshold field limits.
Issues Fixed in This Release		
CSDK-933	C SDK, WS EMS	axTLS should ignore `pathLenConstraint` on root certificates axTLS in the C SDK now accepts null `pathLenConstraint` for certificates. With this change, even if the root certificate of a certificate chain does not have `pathLenConstraint`, the C SDK can still connect to a platform. If one (and only one) intermediate certificate in the chain does not have `pathLenConstraint`, the C SDK can still connect. However, if the certificate chain has additional intermediate certificates that do not have `pathLenConstraint`, the C SDK cannot connect. This change applies only to root certificates in the chain. For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9.. IMPORTANT! See also the Known Issue, CSDK–916.
CSDK-837 (13573872)	C SDK	Queue size in **SetPropertyVTQ** is incorrect when enabling fold The offline message storage queue size was incorrect with folding enabled. The **QueueValueForSending** function in the file,`twSubscribedProps.c`, judged based on the *fold* parameter. If the value of the *fold* parameter is true, it needs to find and replace the property or add it if it does not exist. So, if the property exists in the offline message store fle, it will be replaced instead of added as a new property. The *>queueSize* should not change when an existing property is replaced, but in fact it did change. This issue is resolved in this release.
CSDK-883	C SDK	Quality Only Changes on VALUE Properties Are Not Propagated. If a property was using a push type of `VALUE`, and (only) the quality for that property changed, that update was not propagated to ThingWorx platform. A `VALUE` push type is now handled such that an update is pushed based on either a value or quality change.
Known Issues		
CSDK-916	C SDK	C SDK v.1.4.x and 1.5.0 releases use axTLS v.2.1.2. These versions of the C SDK were unable to connect to SSL Tomcat servers that did not specify `pathLenConstraint` in their root certificate (which include PTC Cloud Services production instances of ThingWorx platform). A partial fix for this issue is provided in release 1.5.0 (see CSDK-933 above) for the problem with the root certificate in a chain, where the chain consists of the root, one intermediate certificate (with or without `pathLenConstraint`), and a server

Issue ID (SFID)	Applies to	Description
		certificate (which is not a CA so constraints are irrelevant). However, the problem can still occur if a certificate chain has more than one intermediate certificate that does not have `pathLenConstraint`. For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9.. IMPORTANT! This issue is present only when using axTLS builds of the C SDK v.1.4.x and 1.5.0. OpenSSL-based C SDK builds do not experience this issue.
CSDK-911 (13591031)	C SDK	SDK client code does not receive the actual exception message thrown from ThingWorx service Invoking a service from ThingWorx SDK client returns a Generic Exception and not the actual exception thrown in ThingWorx local service. Without the actual exception, a developer cannot take action based on the exception.
CSDK-891 (13647664)	C SDK	C SDK crashes with null pointer exception in `twList_Create` function. If the allocation of the mutex fails, the list is deleted and the 'list' pointer is set to null (0). The list delete_function assignment statement is then executed. The 'list' will be NULL at this time, causing a crash.
CSDK-888 (13338641)	C SDK	C SDK can reuse request ids, which can lead to unexpected behavior. The C SDK was observed sending two requests with the same request id. Request 1 timed out, and request 2 appeared to succeed. However, the session id received from the platform for the authentication request was invalid, causing all follow-on communication to fail with an invalid session error.
CSDK-115 (13373562)	C SDK	Size of the file limited by `twcfg.offline_msg_queue_size` is not honored. When using the subscribed property manager and a device is offline when calling `twApi_PushSubscribedProperties()`, the property values are written to a persisted file named `subscribed_props.bin`. The size of the file should be limited by the variable `twcfg.offline_msg_queue_size`. However, the actual file will grow considerably larger than the value specified by `twcfg.offline_msg_queue_size`.

C SDK Version 1.4.1

Issue	Applies to	Description
Issues Fixed in This Release		
CSDK-743	C SDK, WS EMS 5.3.3	SPM's most recent value cache never removes deleted properties. The Subscribed Property Manager (SPM) retains the most recent value for all properties that it has seen whose push type is set to `VALUE`. This cache always retained the most recent VTQ for every

Issue	Applies to	Description
		property processed by the SPM, even after those properties were removed from the Thing on ThingWorx platform or callbacks for the property were unregistered. This behavior continually increased the size of the cache. This issue is fixed in this release.
CSDK-802	C SDK, WS EMS 5.3.3	Multiple open tunnels can cause tunnel disconnections. This release resolves this issue.
CSDK-856	C SDK, WS EMS 5.3.3	Add note to C SDK developer's guide about tick resolution parameter for tunneling. The C SDK developer's guide has updates concerning tick resolution and running the C SDK on Windows-based operating systems.
Known Issue		
CSDK-916	C SDK 1.4.x	C SDK v.1.4.x releases use axTLS v.2.1.2. These versions of the C SDK are unable to connect to SSL Tomcat servers that do not specify `pathLenConstraint` in their installed server certificate (which include PTC Cloud Services production instances of ThingWorx platform). For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9.. IMPORTANT! This issue is present only when using axTLS builds of the C SDK v.1.4.x. OpenSSL-based C SDK builds do not experience this issue.
CSDK-115 (13373562)	C SDK	Size of the file limited by `twcfg.offline_msg_queue_size` is not honored. When using the subscribed property manager and a device is offline when calling `twApi_PushSubscribedProperties()`, the property values are written to a persisted file named `subscribed_props.bin`. The size of the file should be limited by the variable `twcfg.offline_msg_queue_size`. However, the actual file will grow considerably larger than the value specified by `twcfg.offline_msg_queue_size`.

C SDK Version 1.4.0

Issue	Applies tò	Description
New in This Release		
CSDK-1	C SDK	Replace all InfoTable iterations using **twList_Next()** within the Subscribed Property Manager (SPM) with **twList_forEach()**. To improve performance of the Subscribed Property Manager (SPM), the iterations over infotables that previously used the **twList_Next()** function now use `twList_foreach()`.
CSDK-245	C SDK	The C SDK now ships with CMake. The distribution bundle includes CMake files so that developers can use CMake to build their applications for their desired platforms. The *ThingWorx Edge C SDK Developer's Guide* has been updated for this release to provide guidance for CMake.

Issue	Applies tò	Description
		NOTE: While the old makefiles are still in the distribution bundle, they are deprecated with this release.
CSDK-256 (formerly EDGE-2185)	C SDK	Implement Synchronized State Handler within the C SDK. The C SDK now provides the ability to install handlers to notify developers when the `getPropertySubscriptions()` is being called on ThingWorx platform in response to a `notifyPropertyUpdates` message. This mechanism works in a similar manner to the bind event handler.
CSDK-321	CSDK-	Add public **twApi** functions to get/set all configurable properties in the `twcfg struct`. When compiling dynamic .exe's/dll's on Windows, the `twcfg struct` is accessible onlly to the core API. Therefore, external libraries could not access certain critical configuration members, such as offline message size. All members of `twConfig.twcfg` can be written by public **twApi** calls. Note that `twApi` has NOT changed.
CDSK-362 (formerly EDGE-506)	C SDK	Upgrade C SDK to use axTLS 2.1.2. The updates to the most recent release of axTLS include security fixes. The C SDK distribution bundle includes this latest version. The C SDK has been tested to connect to an instance of ThingWorx platform using the default cipher suites used by Java 8 and Tomcat 8.0.36+. For more information about axTLS 2.1.2, visit http://axtls.sourceforge.net/.
CSDK-789	C SDK	Modify the push behavior of the Subscribed Property Manager (SPM) to be multithreaded. To significantly improve performance the **twApi_ PushSubscribedPropertiesAsync** function and its callback handling have been implemented. It is now possible for multiple calls to the **UpdateSubscribedPropertyValues** service to be outstanding at the same time. The new API is **twApi_ PushSubscribedPropertiesAsync(char * entityName, char forceConnect,PushSubscribedPropertiesAsyncCallback cb, void* userdata)**. For more information, see the Doxygen documentation for the C SDK.
Issues Fixed in This Release		
CSDK-13	C SDK	Column order in DataShape generated via C API is not maintained in data sent to ThingWorx. Column order is nOT guaranteed to be preserved in InfoTables. The Data Shape used to specify the InfoTable supports an ordinal aspect on fields. This aspect can be used at display time to render the fields in the correct order.

Issue	Applies tò	Description
		Note too that InforTables store field values by key, not index. See the *ThingWorx Edge C SDK Developer's Guide* for more information.
CSDK-14 (formerly EDGE-1964)	C SDK and .NET SDK (until other SDKs and WS EMS are built with C SDK 1.4.1)	The C SDK could not establish a secure WebSocket connection (WSS, SSL) to a ThingWorx server. when attempting to connect. This issue has applied to the C SDK 1.3.2 through 1.3.5, .NET SDK 5.6.2 through .5.6.5, WS EMS 5.3.2.x, and iOS SDK 1.0. CAUSE: Versions of Apache Tomcat 8.0.34 and above have disabled RSA-based ciphers by default due to forward secrecy concerns. (see https://tomcat.apache.org/tomcat-8.0-doc/changelog.html for 8.0.35.). The axTLS libraries used by WS EMS, C SDK, .NET SDK, and iOS SDK supported two encryption ciphers: TLS_RSA_WITH_AES_256_CBC_SHA and TLS_RSA_WITH_AES_128_CBC_SHA. Any application that used SSL for Edge connections if the Tomcat server was upgraded to 8.0.35 or later may have been affected by this change to Tomcat. With the upgrade to axTLS 2.1.2, this issue is resolved for the C SDK in this release. ⚠️ **Caution** The WS EMS, .NET SDK, and iOS SDK have not been rebuilt with C SDK 1.4.0 at this time. To avoid the issue, make sure that you are using the C SDK 1.4.0. Otherwise, refer to the release notes for WS EMS, .NET SDK, or iOS SDK for workarounds.
CSDK-55	C SDK	C SDK rejecting multipart messages with more than 127 chunks. The C SDK message handling code incorrectly parsed the chunk ID and chunk count from incoming multipart messages, causing the message handling code to reject the chunks and ultimately ignore the message. This issue is fixed in this release.
CSDK-739	C SDK	Possible Deadlock in `twList_foreach()` Handler. To avoid a possible deadlock in Linux, do NOT call `twList_Remove` from within a `foreach` handler.
CSDK-746	C SDK	Improper return code caused Subscribed Property Manager (SPM) to buffer infinite records while offline. An improperly defined return code caused the SPM to report pushes that failed to either be published or persisted as "persisted", resetting the enqueued count and allowing the SPM's VQT buffer to grow indefinitely. This issue is fixed in this release.
CSDK-748	C SDK	Subscribed Property Manager (SPM) — most recent value cache only compared property name.

Issue	Applies tò	Description
		The SPM retains a list of the most recent value of every property with the push type "VALUE" that it has seen so that it can compare the next value it receives against the last. When doing lookups in this cache, the SPM compares only the property names (remote edge names), not the entity names. If two bound Things have properties with the same name, they share a most recent value cache.
		This issue is fixed in this release. The SPM compares the Thing Name as well as the property names when doing lookups in the cache.
CSDK-749	C SDK	Subscribed Property Manager (SPM) — most recent value cache iteration uses `twList_Next`.
		The SPM uses a single `twList` containing references to the most recent value for every property of push type "VALUE" that the SDK has seen since initialization. Previously, as each VQT came in, a linear search was run on this cache for a property whose property name matched the incoming VQT's property name (see also CSDK-748). This iteration was performed using **twList_Next**, which was quadratic in complexity. This list can grow to contain many properties at runtime. This search method adversely affected performance when a list grew.
		This release resolves this issue. The function **twList_foreach** is now used instead of **twList_Next** to search for matches. In addition, the search also compares the Thing Name associated with the incoming property name to the Thing Name associated with the property in the cache..
CSDK-750	C SDK	`SetPropertyVTQ` returns `CHANGE_BUFFER_FULL` if property does not change by configured value.
		The function `twSubscribedPropsMgr_SetPropertyVTQ` returns `TW_PROPERTY_CHANGE_BUFFER_FULL` if the passed VTQ is for a property with push type VALUE, and the new value in the VTQ does not exceed (oldValue + threshold). The same happens if the property is set to a push type of NEVER. The result was that the linking application determined that the SPM's buffer was full and likely caused it to force a push before trying to re-enqueue the same value, which continued to fail in the same way.
		This issue is fixed in this release. The SPM compares the Thing Name as well as the property names when doing lookups in the cache.
CSDK-780	CSDK-	`subscribedPropertyUpdateTaskForEachHandler` invokes listeners with bad arguments.
		This issue is fixed in this release.
Known Issues		

Issue	Applies tò	Description
CSDK-916	C SDK	C SDK v.1.4.x releases use axTLS v.2.1.2. These versions of the C SDK are unable to connect to SSL Tomcat servers that do not specify `pathLenConstraint` in their installed server certificate (which include PTC Cloud Services production instances of ThingWorx platform). For information about the `pathLenConstraint` parameter, see RFC 5280, section 4.2.1.9.. IMPORTANT! This issue is present only when using axTLS builds of the C SDK v.1.4.x. OpenSSL-based C SDK builds do not experience this issue.
CSDK-812	C SDK, .NET SDK	SPM is incompatible with RAM backed offline message store. If the offline message store is not written to files, the SPM initialization fails to initialize the SPM structure's `persistedValues` stream. When attempting to push properties when not connected, this issue leads to a persistence failure in the SPM. WORKAROUND: Use the file type of offline message store.
CSDK-115 (13373562)	C SDK	Size of the file limited by `twcfg.offline_msg_queue_size` is not honored. When using the subscribed property manager and a device is offline when calling `twApi_PushSubscribedProperties()`, the property values are written to a persisted file named `subscribed_props.bin`. The size of the file should be limited by the variable `twcfg.offline_msg_queue_size`. However, the actual file will grow considerably larger than the value specified by `twcfg.offline_msg_queue_size`.

C SDK Version 1.3.6

Issue	Applies to	Description
Issue Fixed in This Release		
CSDK-785	C SDK, .NET SDK	Determine cause of incoming message queue filling up, which caused message to be discarded. The SDK now cleans up registered properties and services if a BIND fails.
Known Issues		

C SDK Version 1.3.6 (continued)

Issue	Applies to	Description
CSDK-812	C SDK, .NET SDK	SPM is incompatible with RAM backed offline message store. If the offline message store is not written to files, the SPM initialization fails to initialize the SPM structure's `persistedValues` stream. When attempting to push properties when not connected, this issue leads to a persistence failure in the SPM. WORKAROUND: Use the file type of offline message store.
EDGE-1964	C SDK, .NET SDK, iOS SDK, and WS EMS	The Edge device (C SDK) cannot establish a secure WebSocket connection (WSS, SSL) to a ThingWorx server. The error appears as `Error 0`, `Error initializing SSL connection`, `twWs_ Connect: Error restarting socket. Error 0`, and/or `No compatible ciphers` when a C SDK device attempts to connect. This issue applies to the C SDK 1.3.2 through 1.3.6, .NET SDK 5.6.2 through .5.6.5, WS EMS 5.3.2.x, and iOS SDK 1.0. CAUSE: Versions of Apache Tomcat 8.0.35 and above have disabled RSA-based ciphers by default due to forward secrecy concerns. (see https://tomcat.apache.org/tomcat-8.0-doc/changelog.html for 8.0.34.). The axTLS libraries used by WS EMS, C SDK, .NET SDK, and iOS SDK support two encryption ciphers: TLS_RSA_WITH_AES_256_CBC_SHA and TLS_RSA_WITH_AES_128_CBC_SHA. Any application that uses SSL for Edge connections if the Tomcat server is upgraded to 8.0.35 or later may be affected by this change to Tomcat. WORKAROUNDS: • Downgrade to a version of Tomcat version 8.0.33 or lower. • In the `server.xml` configuration file of Tomcat, explicitly define a list of ciphers that includes the axTLS ciphers. For an example with a list of ciphers supported in Tomcat version 8.0.36, see https://support.ptc.com/appserver/cs/view/solution.jsp?n=CS245522. ⚠ **Caution** Make sure that any ciphers you use have been validated with any internal security requirements before implementing this workaround in production environment

C SDK Version 1.3.5

Issue	Applies to	Description
Issue Fixed in This Release		
CSDK-755	C SDK and WS EMS	`twTlsClient_Create` could cause a segfault if multiple threads each tried to create a TLS client at nearly the same time. This issue was seen in ThingWorx WebSocket—based Edge MicroServer (WS

C SDK Version 1.3.5 (continued)

Issue	Applies to	Description
		EMS), where communication between the WS EMS and Lua Script Resource could result in the creation of multiple HttpClient objects, each of which ran on its own thread and tried to create a `twTlsClient`. When two threads try to create a new `twTlsClient`, they could both call into `SSL_library_init()`, which caused heap corruption and a crash inside OpenSSL.
CSDK-129	C SDK and WS EMS	If a multipart response was received by the edge device, and that response arrived after its corresponding request had timed out, the final chunk of that multipart response is freed twice and the reconstituted message is leaked.
Enhancement in This Release		
CSDK-745	C SDK and WS EMS	The C SDK now exposes the **tunnelServiceCallback** function so that clients can use the tunnel services that are implemented in `twTunnelManager.c`.
Known Issues		

C SDK Version 1.3.5 (continued)

Issue	Applies to	Description
CSDK-812	C SDK, .NET SDK	SPM is incompatible with RAM backed offline message store. If the offline message store is not written to files, the SPM initialization fails to initialize the SPM structure's `persistedValues` stream. When attempting to push properties when not connected, this issue leads to a persistence failure in the SPM. WORKAROUND: Use the file type of offline message store.
EDGE -1964 / CSDK-14	C SDK, .NET SDK, iOS SDK, and WS EMS	The Edge device (C SDK) cannot establish a secure WebSocket connection (WSS, SSL) to a ThingWorx server. The error appears as `Error 0`, `Error initializing SSL connection`, `twWs_Connect: Error restarting socket. Error 0`, and/or `No compatible ciphers` when a C SDK device attempts to connect. This issue applies to the C SDK 1.3.2 through 1.3.5, .NET SDK 5.6.2 through .5.6.4, WS EMS 5.3.2.x, and iOS SDK 1.0. CAUSE: Versions of Apache Tomcat 8.0.35 and above have disabled RSA-based ciphers by default due to forward secrecy concerns. (see https://tomcat.apache.org/tomcat-8.0-doc/changelog.html for 8.0.34.). The axTLS libraries used by WS EMS, C SDK, .NET SDK, and iOS SDK support two encryption ciphers: TLS_RSA_WITH_AES_256_CBC_SHA and TLS_RSA_WITH_AES_128_CBC_SHA. Any application that uses SSL for Edge connections if the Tomcat server is upgraded to 8.0.35 or later may be affected by this change to Tomcat. WORKAROUNDS: • Downgrade to a version of Tomcat version 8.0.33 or lower. • In the `server.xml` configuration file of Tomcat, explicitly define a list of ciphers that includes the axTLS ciphers. For an example with a list of ciphers supported in Tomcat version 8.0.36, see https://support.ptc.com/appserver/cs/view/solution.jsp?n=CS245522. ⚠️ **Caution** Make sure that any ciphers you use have been validated with any internal security requirements before implementing this workaround in production environment

C SDK Version 1.3.4

Issue	Applies to	Description
Issues Fixed in This Release		
EDGE-2127	C SDK 1.3.4	After pulling out the network cable from a device, C SDK 1.3.2 using OpenSSL hangs in `SSL_write` (for OpenSSL connection) until the cable is plugged back in. (64–bit Debian Linux)

C SDK Version 1.3.4 (continued)

Issue	Applies to	Description
		This hang occurred consistently when sending the second chunk of a multipart message. This issue is fixed in this release.
EDGE-2137	C SDK 1.3.4 and .NET SDK 5.6.4	When the C SDK tries to send `offline_msgs.bin`, no properties are logged on ThingWorx platform. This issue has been fixed in this release.
EDGE-2138	C SDK1.3.4	In C SDK 1.3.3, the C SDK leaks memory in `twMap_Create()`. This issue appeared when calling `twMapGetDeleteFunctionMap` and `twMapGetParseFunctionMap` in items that were being created in `twMap`. This issue is fixed in this release.
EDGE-2140	C SDK 1.3.4 and .NET SDK 5.6.4	The SteamSensor example segfaults when yanking Ethernet cable. The fix for EDGE-2137 also fixed this issue in this release.
EDGE-2157	C SDK 1.3.4, .NET SDK 5.6.4	Remove registration of `NotifyPropertyUpdate` handler in `twApi_BindThingWithoutDefaultServices`. The .NET SDK uses the function, `twApi_BindThingWithoutDefaultServices`, internally. ⚠️ **Caution** Do NOT use `twApi_BindThingWithoutDefaultServices` for C SDK development. It will not register handlers for `GetMetadata` or `NotifyPropertyUpdate`. As a result, it will not properly bind Things to ThingWorx platform.
NSDK-22	C SDK 1.3.4 and .NET SDK 5.6.4	.NET SDK has an error and crashed without error message in log file The issue was in the C SDK that is the foundation for the .NET SDK. The C SDK has the fix for this issue, and the new C SDK libraries are included with the .NET SDK so that it too has the fix.

C SDK Version 1.3.4 (continued)

Issue	Applies to	Description
Known Issue		
EDGE -1964 / CSDK-14	C SDK, .NET SDK, iOS SDK, and WS EMS	The Edge device (C SDK) cannot establish a secure WebSocket connection (WSS, SSL) to a ThingWorx server. The error appears as `Error 0`, `Error initializing SSL connection`, `twWs_ Connect: Error restarting socket. Error 0`, and/or `No compatible ciphers` when a C SDK device attempts to connect. This issue applies to the C SDK 1.3.2 through 1.3.4, .NET SDK 5.6.2 through .5.6.4, WS EMS 5.3.2, and iOS SDK 1.0. CAUSE: Versions of Apache Tomcat 8.0.35 and above have disabled RSA-based ciphers by default due to forward secrecy concerns. (see https://tomcat.apache.org/tomcat-8.0-doc/changelog.html for 8.0.34.). The axTLS libraries used by WS EMS, C SDK, .NET SDK, and iOS SDK support two encryption ciphers: TLS_RSA_WITH_AES_256_ CBC_SHA and TLS_RSA_WITH_AES_128_CBC_SHA. Any application that uses SSL for Edge connections if the Tomcat server is upgraded to 8.0.35 or later may be affected by this change to Tomcat. WORKAROUNDS: • Downgrade to a version of Tomcat version 8.0.33 or lower. • In the `server.xml` configuration file of Tomcat, explicitly define a list of ciphers that includes the axTLS ciphers. For an example with a list of ciphers supported in Tomcat version 8.0.36, see https://support.ptc.com/appserver/cs/view/solution.jsp?n= CS245522. ⚠ **Caution** Make sure that any ciphers you use have been validated with any internal security requirements before implementing this workaround in production environment

C SDK Version 1.3.3

Issue	Applies to	Description
New in This Release		
EDGE-2025	C SDK, .NET SDK	Add bind message splitting. The bulk binding feature could cause the SDK to attempt to send a message that is larger than the configured maximum message size. When building the bind message, the SDK monitors the message size and returns an error when the message is full. When this error message is detected, the SDK sends the current message and starts a new one.
EDGE-1977	C SDK	Support `libcfu` as an implementation of `twMap`
EDGE-1900, EDGE-2044	C SDK	Add ForEach iterator function to `twList` API. To all list iterations to be performed faster and in a thread-safe manner, the `twList_Foreach` iterator function has been added to `twList` API for this release. The `twList_Next` significantly

C SDK Version 1.3.3 (continued)

Issue	Applies to	Description
		impacts performance and has been deprecated in favor or `twList_ Foreach`. It has been replaced in the following source files: `twApi.c, twInfoTable.c, twTunneling.c, twFileManager.c, twMessaging.c, twSubscribedProps.c, list.c, twFileTransferCallbacks.c, twHttpProxy.c,` and `logger.c`.
EDGE-2005,	C SDK	Replace twList with twMap in identified places. To improve the performance of the C SDK, certain uses of `twList` have been replaced with `twMap`.
EDGE-1758,	C SDK	Provide a List-backed `twMap` forC SDK. For this release, a list-backed `twMap` and `twDict` have been added. You can select the implementation you wan to use at compiled time. Note that map references should use a type of `twMap`.
EDGE-1653	C SDK, .NET SDK	To allow bulk binding in the C SDK and derived SDKs the **twApi_ BindThings** function has been added for this release.
Issues Fixed in This Release		
EDGE-1870	C SDK	Leak in `twSubscribedPropsMgr_SetPropertyVTQ` when the property is unknown. If a VTQ is passed to `twSubscribedPropsMgrVTQ` after the property for which it is intended has been removed, the function leaks a `twSubscribedProperty`. This issue is fixed in this release.
EDGE-1834	C SDK	O(n^2) performance degradation when retrieving parameters from tables with large row counts. When retrieving parameters from an infotable (for example, as input to a service), performance degrades exponentially as the number of rows in the table grows. To resolve this issue, the `twList_Next()` function has been replaced with direct locking and iteration (`twList_ GetByIndex()`), significantly improving perfonnance.
EDGE- 1737	C SDK	`twApi_SetSubscribedPropertyVTQ` returns success when the internal buffer of the SDK is full and the VTQ is going to be ignored. In `twSubscribedProps.c`, an error condition was logged and memory was cleaned up. However, the function returned `success`, which did not reflect the fact that the update exceeded the queue size and therefore was not actually sent. A return code has been added so that the queue can be flushed immediately (with `twApi_ PushSubscribedProperties`) and resubmitted.

C SDK Version 1.3.3 (continued)

Issue	Applies to	Description
New in This Release		
EDGE-1736	C SDK	Double free of a multi-part response chunk if the request times out. If a multi-part response was received by the edge device, and that response arrived after its corresponding request timed out, the final chunk of that response was freed twice, and the reconstituted message was leaked. This issue is fixed in this release.
EDGE-1735	C SDK	The function, `twDataShapeEntry_CreateFromStream()` can corrupt the stream if it encounters an entry with an aspect. This issue is fixed in this release.
EDGE-1690	C SDK	`twSubscribedProps.c` leaks memory in some situations. When the offline message store was compiled out, the edge application was not connected and was not forcing the SDK to attempt a connection. This issue is fixed in this release.
EDGE-1647	C SDK	C SDK leaks memory while handling services if there is no output Data Shape present. If a service was defined without an output shape (returns noThing), a memory leak could occur when creating a JSON object to store an aspect on `twApi`. When the output Data Shape exists, the associated aspect memory is cleaned up. However, when the shape does not exist, the aspect JSON object is created, never populated, and never cleaned up. This issue is fixed in this release.
EDGE-1568	C SDK WS EMS	`twInfoTable_CreateFromJson` leaks memory in certain cases. This switch block in `twInfoTable_CreateFromJson` was missing break statements, which caused primitives to be created and abandoned for certain base types. This issue is fixed in this release.
EDGE-630	C SDK	Data lost with offline storage when network disconnects. This issue is fixed in this release. Offline message store is automatically initialized by `twApi_Initialize()`, so you do not need to initialize it separately. However, if your application has complex requirements, you can use the offline message store singleton that was added for this release, **twOfflineMsgStore_Initialize()** to initialize it separately. The singleton enables or disables offline message storage, passes in the maximum size of the

C SDK Version 1.3.3 (continued)

Issue	Applies to	Description
New in This Release		
		store, and the path to the location of the offline message store. This function has no dependencies from **twApi** and **twMessage** in case **OFFLINE_MSG_STORE** is undefined. This release provides three additional functions for offline message storage: • **twOfflineMsgStore_SetDir()** • **twOfflineMsgStore_Delete()** • `twOfflineMsgStore_HandleRequest()` See the files of the C SDK installation, `/src/twUtils/twOfflineMsgStore.h`and `/src/twUtils/twOfflineMsgStore.c`, for definitions and implementation of these functions. The subscribed property manager has been decoupled from offline message store. Dependencies for Linux and Windows have been resolved.
Known Issue		

C SDK Version 1.3.3 (continued)

Issue	Applies to	Description
New in This Release		
EDGE -1964 / CSDK-14	C SDK, .NET SDK, iOS SDK, and WS EMS	The Edge device (C SDK) cannot establish a secure WebSocket connection (WSS, SSL) to a ThingWorx server. The error appears as `Error 0`, `Error initializing SSL connection`, `twWs_ Connect: Error restarting socket. Error 0`, and/or `No compatible ciphers` when a C SDK device attempts to connect. This issue applies to the C SDK 1.3.2 and 1.3.3, .NET SDK 5.6.2 and .5.6.3, WS EMS 5.3.2, and iOS SDK 1.0. CAUSE: Versions of Apache Tomcat 8.0.35 and above have disabled RSA-based ciphers by default due to forward secrecy concerns. (see https://tomcat.apache.org/tomcat-8.0-doc/changelog.html for 8.0.34.). The axTLS libraries used by WS EMS, C SDK, .NET SDK, and iOS SDK support two encryption ciphers: TLS_RSA_WITH_ AES_256_CBC_SHA and TLS_RSA_WITH_AES_128_CBC_SHA. Any application that uses SSL for Edge connections if the Tomcat server is upgraded to 8.0.35 or later may be affected by this change to Tomcat. WORKAROUNDS: • Downgrade to a version of Tomcat version 8.0.33 or lower. • In the `server.xml` configuration file of Tomcat, explicitly define a list of ciphers that includes the axTLS ciphers. For an example with a list of ciphers supported in Tomcat version 8.0.36, see https://support.ptc.com/appserver/cs/view/solution.jsp?n=CS245522. ⚠ **Caution** Make sure that any ciphers you use have been validated with any internal security requirements before implementing this workaround in production environment

C SDK Version 1.3.2

ID	Applies To	Description
Issues Fixed		
EDGE-758	C SDK and WS EMS	When calling the `GetRemoteMetadata` service from the ThingWorx platform via a Connection Server (v.6.5.11, 7.0, and 7.0.1), the Connection Server logs an error and the ThingWorx platform service times out. The WS EMS is successfully receiving the request and sending packets back to the Connection Server. This issue is fixed in this release.
EDGE-874	C SDK and WS EMS	The WS EMS was responding very slowly to requests, in comparison to v.5.2.2 and 5.3.0. This issue is fixed in this release.
EDGE-996	C SDK	`twFileManager_SendFile()` stopped sending files. This issue is resolved.

C SDK Version 1.3.2 (continued)

Issues Fixed		
ID	Applies To	Description
EDGE-1046	C SDK	File transfer that use identifiers were not working with the C SDK. This issue is fixed in this release.
EDGE-1074	C SDK	Some C SDK functions do not return `TW_OK` on success The documentation for the functions (Doxygen output bundled with the C SDK) has been updated for this release. The updated doc is also in the ThingWorx Edge Help Center.
EDGE-1087	C SDK	C SDK crashed when the SSL layer was disabled. This issue is fixed in this release.
EDGE-1125	C SDK	The C SDK code causes a crash at `twApi_Delete()` function and the application stops. This issue is fixed in this release.
EDGE-1297	C SDK	String property does not push changes if Composer `pushType` is not set to `Always`. This issue is fixed in this release.
EDGE-1432	C SDK	C SDK `twDirectory_MoveFile` should perform a copy and delete instead of a rename. RESOLUTION: The `twDirectory_MoveFile` method now attempts a rename. Then, if the rename fails, it copies the file to the destination and deletes the file from the source.
EDGE-1453	C SDK	Implement logic in `twFinishFileTransfer` to search the `openFile` list using the `adjustedPath` if the tid search fails. This issue is fixed in this release.
EDGE-1484	C SDK	File Transfers with Identifiers > 1 Block Size have the potential to not clean up a file (ThingName Based) This issue is fixed in this release.

C SDK Version 1.3.2 (continued)

Issues Fixed		
ID	**Applies To**	**Description**
Known Issue		
EDGE -1964 / CSDK-14	C SDK, .NET SDK, iOS SDK, and WS EMS	The Edge device (C SDK) cannot establish a secure Websocket connection (WSS, SSL) to a ThingWorx server. The error appears as `Error 0`, `Error initializing SSL connection`, `twWs_ Connect: Error restarting socket. Error 0`, and/or `No compatible ciphers` when a C SDK device attempts to connect. This issue applies to the C SDK 1.3.2, .NET SDK 5.6.2, WS EMS 5.3.2., and iOS SDK 1.0. CAUSE: Versions of Apache Tomcat 8.0.35 and above have disabled RSA-based ciphers by default due to forward secrecy concerns. (see https://tomcat.apache.org/tomcat-8.0-doc/changelog.html for 8.0.34.). The axTLS libraries used by WS EMS, C SDK, .NET SDK, and iOS SDK support two encryption ciphers: TLS_RSA_WITH_ AES_256_CBC_SHA and TLS_RSA_WITH_AES_128_CBC_SHA. Any application that uses SSL for Edge connections if the Tomcat server is upgraded to 8.0.35 or later may be affected by this change to Tomcat. WORKAROUNDS: • Downgrade to a version of Tomcat version 8.0.33 or lower. • In the `server.xml` configuration file of Tomcat, explicitly define a list of ciphers that includes the axTLS ciphers. For an example with a list of ciphers supported in Tomcat version 8.0.36, see https://support.ptc.com/appserver/cs/view/solution.jsp?n=CS245522. ⚠ **Caution** Make sure that any ciphers you use have been validated with any internal security requirements before implementing this workaround in production environment

C SDK Version 1.3.1

Issues Fixed		
ID	**Applies To**	**Description**
EDGE-842	C SDK only	The Steam Sensor with OpenSSL example for Linux that is provided in the zip archive for the C SDK did not work without changes to the Makefile. The OpenSSL libraries are now included in the distributed zip archive. The example should now compile "out of the box" from the distributed zip archive. If you have your own OpenSSL libraries, you can build against them as well.
EDGE-834	All SDKs and WS EMS	Pending requests to the ThingWorx platform would time out if the edge device received a request from the ThingWorx platform while it waited for the response.

C SDK Version 1.3.1 (continued)

Issues Fixed		
ID	**Applies To**	**Description**
		Workaround: Increasing the message timeout from the default value of 10 seconds to 60 seconds and changing the port configuration resolves this issue.
EDGE-829	C SDK, .NET SDK, and WS EMS	The ListFiles service gives different result for WS EMS 5.3 and 5.0. The ListFiles service in this release (5.3.1) now returns the path without the file name, as it did in release 5.0.
EDGE-827	C SDK	The Tunnel Manager was using "==" to compare strings. This release resolves this issue.
EDGE-823 Case 12819599	C SDK and WS EMS	The WS EMS running as a service on a Netbiter ec350 device failed to start up and displayed the following message: `Error creating BSD socket.` This release resolves this issue.
EDGE-683	All SDKs and WS EMS	An ADO application could not connect to the ThingWorx platform through an NTLM proxy server. This issue is fixed in this release. Tunneling through an NTLM proxy server is working.

C SDK Version 1.2.0

New Features
• Added support for subscribed properties.
• Added support for aspects on service definitions.
• Added functions for setting both the gateway name and the gateway type to the **twWs** struct.
• Updated developer guide.
• Added capability to clear proxy settings.
• Removed dependency on Joyent HTTP parser.
• SDK now has dependency on cJSON library (included in distribution).
• Changed default socket read timeout to 500ms.
Bug Fixes
The following issues are resolved in this release: • The `NoTls` option will not compile on the C SDK.
• All bound Things were registering with the same Thing Name.
• `GetDirectoryStructure` was not including a leading / character on paths.
• `GetDirectoryStructure` was not including virtual directory names as root directories in the result.
• Fixed deadlock in multi part message init.
• Timeout specified in `twcfg.socket_read_timeout` is now passed into the socket read call.
• Socket read timeouts no longer disconnect websocket.

C SDK Version 1.1.2

Version 1.1.2
New Features
• API reference documentation now included in both the header files and as html in the `doc/html` directory.
• New, updated developer guide.
• Added support for registering events in the C SDK. Events can be browsed from the ThingWorx platform.
• Fix for occasional 'selecting on socket' error reported in logs.
• OSX Makefile fixes.

C SDK Version 1.1.1

New Features
• Added ability to change offline message store directory using `twApi_SetOfflineMsgStoreDir`.
• Compile time settings are now readable in the **twcfg** structure.
• Added OSX build files back into distribution.
• Added **GetDirectoryStructure** service. This change is for backward compatibility only and is deprecated.
• Added a *SteamSensorWithMinimalFootprint* example.
• Modified *SteamSensorWithMinimalFootprint* to be single threaded.
• Changed Linux build of *SteamSensor* to have logging enabled.
Bug Fixes
• Fixed the incoming message handling that occurs within the **sendMessageBlocking** function to handle responses only. Avoids deadlocks in certain situations.
• Changed **twMessage_Send** to check to see if we are authenticated before we send.
• Changed code that sends offline messages to insert a new *RequestId* to remove any potential conflicts from a previous ID.
• Added Mutex protection in **twTlsClient_Reconnect** and `twTlsClient_ConnectSession`.
• Fixed `segfault` that occurs during non-persistent offline message storing.
• Updated copyright in documentation.
• Added registration of `Location` property so that it shows up when browsed.

C SDK Version 1.1.0

New Features
• Support for transferring files and directories that have multi-byte characters in their names.
• Support for browsing directories and files that have multi-byte characters in their names.
• Support for tunneling.
• Support for NTLM proxy servers.
• Addition of a TLS wrapper to support OpenSSL/FIPS.
• Portable multi-threading layer (use is optional).
• SHA-256 certificate signature support.
• Full certificate validation.
• Run-time overriding of default values.
• Wildcard property and service callback registration.
• Authentication event callbacks.
• Bind event callbacks.

Bug Fixes
• EDGE-318: C SDK disconnects.
• EDGE-291: Segfault on service call with empty input
• EDGE-221: C SDK continues to try to reconnect after number of retries is exhausted.
• EDGE-189: C SDK not checking for ping timeouts after sending a ping.
• EDGE-188: Cannot correctly import services from the C SDK to the ThingWorx platform.
• EDGE-169: C SDK does not reconnect after entire server shutdown.
• `CreateBinaryFile` will now open/write/close if it is called without `StartFileTransfer` being called previously.
• Fixed file transfer to support both 5.0 and later servers.