

# FNV2-FCI Use Cases

Status	IN REVIEW
Completion (%)	100%
Version	0.1
Owner	Zhang, Derek (D.)
Reviewer	Risacher, Frederic (F.) Park, Terry (T.)
Review Date	
Approver	D'Souza, Stephen (S.)
Approve Date	

- Glossary
- 1. FNV2-FCI shall serve as the interface between the SDN and the FNV2-SOA middleware
- 2. MQTT protocol shall be used for messaging from the FNV2-FCI to the SDN
- 3. FNV2-FCI shall play both publisher and subscriber roles
- 4. Payload between FNV2-FCI and SDN is in the form of protobuf
- 5. FNV2-FCI shall support all the FTCP messages
- 6. Exactly one FNV2-FCI MQTT client for SDN
- 7. FNV2-FCI shall send an Connection Status Alert message to SDN
- 8. Unexpected disconnects handled
- 9. FNV2-FCI shall publish the MQTT connection status change to other ECG modules
- 10. FNV2-FCI shall send an Alert message with power mode change status to the SDN
- 11. Liability for indications to Power Management (TCU)
- 12. FNV2-FCI shall process the commands from the SDN in receiving order
- 13. Non-correlated Alert delivery
- 14. FNV2-FCI shall deliver the correlated Alert message in the receiving order
- 15. Retry mechanism for failed message delivery
- 16. Indicate successful message delivery if requested
- 17. SDN/MQTT session authentication
- 18. TLS shall be used to provide authentication, integrity and confidentiality for the messages exchanged between FNV2-FCI and SDN
- 19. SyncP used for FTCP encryption
- 20. FNV2-FCI shall handle message level security errors

## Glossary

Term	Definition
FNV2-FCI	FNV2Ford Cloud Interface
FNV2-SOA	FNV2 Service Oriented Architecture
SDN	Service Delivery Network
MQTT	Message Queue Telemetry Transport
FTCP	Ford Telematics Communication Protocol
GPB	Google Proto Buffer
GIVIS	Global in-Vehicle Information System Cloud
CCS	Customer Connectivity Settings
MQTT Delivery Token	The Delivery Token is an object. When a client publishes on a topic, a new delivery token is created to monitor the delivery of a publication. The normal use of the token is to check whether delivery is complete. The completion of deliveries is asynchronous and depends on the QoS associated publication.

## 1. FNV2-FCI shall serve as the interface between the SDN and the FNV2-SOA middleware

<b>Actors</b>	FNV2-FCI, SDN, FNV2-SOA middleware
<b>Pre-conditions</b>	ECG to SDN connection has been established
<b>Scenario Description</b>	<p>FNV2-FCI shall serve as the interface between the SDN and the FNV2-SOA Middleware.</p> <p>The FNV2-SOA middleware is the main communication backbone for all the other ECG software components and some of hardware modules. All those component rely on FNV2-SOA to exchange messages. FNV2-FCI is a FNV2-SOA client which is responsible to send messages to the SDN and receive messages from the SDN. FTCP is the chosen protocol between the SDN and the FNV2-FCI. And the core part of the FTCP is MQTT framework.</p>
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.01
<b>JIRA</b>	<a href="#">DAWN-638</a>

## 2. MQTT protocol shall be used for messaging from the FNV2-FCI to the SDN

<b>Actors</b>	FNV2-FCI, SDN
<b>Pre-conditions</b>	Routable network connection from FNV2-FCI to SDN exists.
<b>Scenario Description</b>	All types of FTCP messaging between SDN and FNV2-FCI is delivered over MQTT infrastructure. This interface terminates on one side at SDN MQTT broker and FNV2-FCI MQTT client on the other side.
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.02
<b>JIRA</b>	<a href="#">DAWN-658</a>

## 3. FNV2-FCI shall play both publisher and subscriber roles

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established

<b>Scenario Description</b>	FNV2-FCI will register itself as a MQTT client to MQTT broker as both subscriber and publisher. The SDN will register to the MQTT broker as another client who can publish and subscribe. Publish/Subscribe decouples a publisher from a subscriber. In order to decide where each message goes, MQTT uses topics. A topic is a hierarchical structured string, which is used for message filtering and routing. The MQTT broker is the heart of the publish/subscribe protocol and it is responsible for receiving all messages from all its subscribers, and filtering them then sending them to all subscribers.
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.03
<b>JIRA</b>	<a href="#">DAWN-659</a>

#### 4. Payload between FNV2-FCI and SDN is in the form of protobuf

<b>Actors</b>	FNV2-FCI, SDN
<b>Pre-conditions</b>	MQTT message exchange between FNV2-FCI and SDN exists.
<b>Scenario Description</b>	FNV2-FCI deserializes the payload of MQTT messages from SDN. FNV2-FCI serializes the payload of MQTT messages to SDN. Google Protocol Buffer (a.k.a. protobuf) is the mechanism for serializing payload data.
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.04
<b>JIRA</b>	<a href="#">DAWN-660</a>

#### 5. FNV2-FCI shall support all the FTCP messages

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established

<b>Scenario Description</b>	<p>FNV2-FCI shall support the following FTCP v4.0.4 message exchange patterns and the correlations among the associated messages with the appropriate QoS.</p> <p>Messages:</p> <ul style="list-style-type: none"> <li>• Command-CommandResponse</li> <li>• Command-CommandResponse-Alert</li> <li>• Alert</li> <li>• Query-QueryResponse</li> <li>• Query-QueryResponse-Command-CommandResponse</li> </ul> <p>Here are the assigned QoS values for the corresponding FTCP topics:</p> <ul style="list-style-type: none"> <li>• TCU Command: this is a generic topic that is used to send all FTCP commands to the vehicle. All messages to this topic shall be published and subscribed to with MQTT QoS of 2.</li> <li>• TCU_CONNECTION_STATUS_ALERT: this topic is used to communicate the connection status of the TCU to the MQTT broker. Messages are published to this topic with QoS of 1.</li> <li>• COMMAND_RESPONSE: the TCU shall publish responses to the commands published by the SDN to this topic. Messages are published to this topic with QoS of 1.</li> <li>• SDN_QUERY: the TCU publishes Query messages to this topic and will be subscribed by the SDN. All messages are published to this topic with QoS of 1.</li> <li>• SDN_QUERY_RESPONSE: the SDN shall publish Query Response messages to this topic. Messages are published to this topic with QoS of 1.</li> </ul>
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.05
<b>JIRA</b>	<a href="#">DAWN-661</a>

## 6. Exactly one FNV2-FCI MQTT client for SDN

<b>Actors</b>	FNV2-FCI, SDN
<b>Pre-conditions</b>	System reboot or network connection failure occurred.
<b>Scenario Description</b>	FNV2-FCI establishes session to SDN/MQTT as a client for both publish and subscribe activities. FNV2-FCI establishes exactly one such connection to SDN/MQTT.
<b>Post-conditions</b>	One MQTT message session between FNV2-FCI and SDN exists.
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.06
<b>JIRA</b>	<a href="#">DAWN-662</a>

## 7. FNV2-FCI shall send an Connection Status Alert message to SDN

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established.

<b>Scenario Description</b>	FNV2-FCI shall send the connection status alert every time it establishes a new MQTT session or when it gracefully terminates one. If FNV2-FCI receives an MQTT delivery token after sending its Connection Status Alert, it shall continue maintaining its MQTT client connection. When FNV2-FCI first establishes the connection, it shall report the connection status as "Connected"; and when FNV2-FCI gracefully disconnects from MQTT broker, it shall report the connection status as "Disconnected". Once FNV2-FCI receives an MQTT token after sending its "Disconnect" alert, it shall gracefully tear down the MQTT connection followed by the cellular data connection.
<b>Post-conditions</b>	ECG to SDN session stays connected.
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.07
<b>JIRA</b>	<a href="#">DAWN-663</a>

## 8. Unexpected disconnects handled

<b>Actors</b>	FNV2-FCI, SDN
<b>Pre-conditions</b>	System reboot or ungraceful network disconnection occurred.
<b>Scenario Description</b>	FNV2-FCI shall implement a connectionLost callback method for connection retries. FNV2-FCI shall also registered itself as a subscriber who is interested in the current connection state for ECG to the internet. If the cellular connection is temporary lost due to network change or out of coverage, FNV2-FCI shall trigger the connectionLoss callback function to reestablish MQTT connection once the cellular connection comes back. If the cellular connection has been switched to WiFi connection, FNV2-FCI shall be able to trigger the connectionLoss callback function to reestablish the MQTT connection through WiFi.
<b>Post-conditions</b>	One MQTT message session between FNV2-FCI and SDN exists.
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.08
<b>JIRA</b>	<a href="#">DAWN-664</a>

## 9. FNV2-FCI shall publish the MQTT connection status change to other ECG modules

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established
<b>Scenario Description</b>	MQTT connection to SDN MQTT broker is established over either the cellular or wifi connection. Due to coverage or network change, the TCP connection may drop and this will cause the MQTT connection to drop temporally. FNV2-FCI shall publish the MQTT connection status change to the interested other ECG components (subscribers).
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A

<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.09
<b>JIRA</b>	<a href="#">DAWN-665</a>

## 10. FNV2-FCI shall send an Alert message with power mode change status to the SDN

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established
<b>Scenario Description</b>	<p>FNV2-FCI shall send the sleep entry alert every time following power mode transitions are performed. These decision for the transitions shall be received by FNV2-FCI from Power Management.</p> <p>The power mode transitions are:</p> <ul style="list-style-type: none"> <li>• Entering Low Power Registered Mode</li> <li>• Entering Sleep Power Mode (SLP-High-Rate scan)</li> <li>• Entering Sleep Power Mode (SLP-Low-Rate scan)</li> <li>• Entering Full Power IGN ON Mode</li> <li>• Entering Full Power Charge Mode</li> <li>• Entering Deep Sleep Mode</li> <li>• Entering DRX Wakeup Mode</li> </ul>
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.10
<b>JIRA</b>	<a href="#">DAWN-666</a>

## 11. Liability for indications to Power Management (TCU)

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	FNV2-FCI to SDN connection has been established
<b>Scenario Description</b>	<p>Query + Response: If a query is being sent to SDN, indicate ongoing FTCP session to the PM.</p> <p>If an Alert or a Command response has been sent and a delivery token is expected, indicate ongoing FTCP session to the PM.</p>
<b>Post-conditions</b>	PM will prevent sleep mode entry if a response to a correlated message is expected from SDN.
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.11
<b>JIRA</b>	<a href="#">DAWN-667</a>

## 12. FNV2-FCI shall process the commands from the SDN in receiving order

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established
<b>Scenario Description</b>	<p>FNV2-FCI shall process the commands it receives from the SDN in the order they were received. The FNV2-FCI should have an internal queue of messages and execute them in a FIFO order.</p> <p>The SDN follows a request-response model to exchange the Commands and Command Response messages with the TCU; every TCU Command Response is correlated with its SDN Command by applying the defined CorrelationID Pattern.</p> <p>Here are the list of Commands and Responses:</p> <ul style="list-style-type: none"> <li>• VehicleStatusUpdateCommand&lt;-&gt;VehicleStatusUpdateCommandResponse</li> <li>• LockCommand&lt;-&gt;LockCommandResponse</li> <li>• UnlockCommand&lt;-&gt;UnlockCommandResponse</li> <li>• ClearUserSettingsCommand&lt;-&gt;ClearUserSettingsCommandResponse</li> <li>• ProvisioningDataRequestCommand&lt;-&gt;ProvisioningDataRequestCommandResponse</li> <li>• SubscriptionStateNotificationCommand&lt;-&gt;SubscriptionStateNotificationCommandResponse</li> <li>• GetUserSettingsCommand&lt;-&gt;GetUserSettingsCommandResponse</li> <li>• AuthorizationStatusChangeCommand&lt;-&gt;AuthorizationStatusChangeCommandResponse</li> <li>• ChangeHomeURLCommand&lt;-&gt;ChangeHomeURLCommandResponse</li> <li>• WifiDataPlanNotificationCommand&lt;-&gt;WifiDataPlanNotificationCommandResponse</li> <li>• MapUpdateNotificationCommand&lt;-&gt;MapUpdateNotificationCommandResponse</li> </ul>
<b>Post-conditions</b>	The Commands are sent by SDN to the vehicle and the Command Responses are received by SDN from the vehicle.
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.12
<b>JIRA</b>	<a href="#">DAWN-668</a>

## 13. Non-correlated Alert delivery

<b>Actors</b>	FNV2-FCI, SDN, FNV2-SOA
<b>Pre-conditions</b>	FNV2-FCI to SDN connection has been established
<b>Scenario Description</b>	<p>FNV2-FCI shall deliver alerts originating on the vehicle in order they were most recently reported (via FNV2-SOA). If the rate of alerts being reported is higher than the rate of alerts being sent to SDN, FNV2-FCI will queue the alerts in a stack like (LIFO) buffer, up to a set max limit.</p> <p>It is understood that SDN will interpret and correlate any and all alerts that might be related. For example:</p> <ul style="list-style-type: none"> <li>• 'Condition X occurred' alert is is passed to FNV2-FCI for SDN delivery, immediately followed by 'Condition X cleared' alert</li> <li>• FNV2-FCI will queue the 'Condition X occurred' and publish the 'Condition X cleared' to SDN first</li> <li>• FNV2-FCI will next retrieve the 'Condition X occurred' from the (LIFO) queue and publish to SDN</li> <li>• SDN will correlate and ignore the latter alert, as it is now obsolete</li> </ul>
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.13

JIRA	<a href="#">DAWN-669</a>
------	--------------------------

## 14. FNV2-FCI shall deliver the correlated Alert message in the receiving order

<b>Actors</b>	FNV2-FCI, SDN, TCU
<b>Pre-conditions</b>	ECG to SDN connection has been established
<b>Scenario Description</b>	<p>FNV2-FCI shall deliver the correlated alerts (alerts triggered as a result of a command from SDN) in the order they were received. This means that the ECG should have an internal queue and shall execute the alerts from the queue in a FIFO manner.</p> <p>List of correlated Alert messages:</p> <ul style="list-style-type: none"> <li>• RemoteStartBeginAlert</li> <li>• RemoteStartEndAlert</li> <li>• CustomerRequestedSOCReachedAlert</li> <li>• UserAuthorizationResponseAlert</li> <li>• SchedulerDriveConditioningSyncAlert</li> <li>• FavoriteLocationDataSyncAlert</li> <li>• ChargeScheduleStatusChangeSyncAlert</li> <li>• ScheduledChargeNotOccurringAlert</li> <li>• ChargedAndDriveConditionCompleteAlert</li> <li>• VehicleDiagnosticDataResponseAlert</li> <li>• VehicleConfigChangeResponseAlert</li> <li>• FirmwareUpgradeStatusAlert</li> <li>• FleetVehicleDiagnosticDataResponseAlert</li> <li>• CCSUpdateAlert</li> <li>• CCSUserPromptResponseAlert</li> </ul>
<b>Post-conditions</b>	Alert messages are sent by ECG and received by SDN.
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.14
<b>JIRA</b>	<a href="#">DAWN-670</a>

## 15. Retry mechanism for failed message delivery

<b>Actors</b>	FNV2-FCI, SDN
<b>Pre-conditions</b>	FNV2-FCI is connected to SDN via MQTT
<b>Scenario Description</b>	<p>FNV2-FCI sends an Alert or a Command Response to SDN and will await delivery confirmation in form of a 'delivery token'.</p> <p>In the event of FNV2-FCI not receiving a delivery token from SDN within the time period specified, FNV2-FCI will send the same Alert or Command Response anew, with a different message ID. This retry will be repeated until either a delivery token is received or defined number of 'max retries' is reached. If max retries is reached, FNV2-FCI will cease sending the same Alert or Command Response, but will queue them for later retry to occur when a new MQTT session with SDN has been established.</p>
<b>Post-conditions</b>	Alert or a Command Response to SDN delivery is confirmed by a 'delivery token'
<b>List of Exception Use Cases</b>	N/A



<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.15
<b>JIRA</b>	<a href="#">DAWN-671</a>

## 16. Indicate successful message delivery if requested

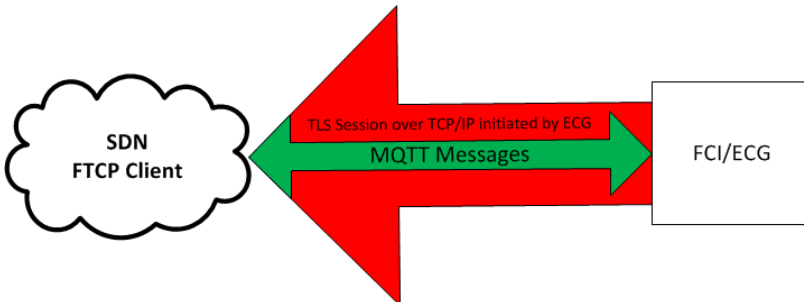
<b>Actors</b>	FNV2-FCI, FNV2-SOA
<b>Pre-conditions</b>	FNV2-SOA client (ECG) originates an Alert or Command Response to SDN with delivery confirmation requested
<b>Scenario Description</b>	FNV2-FCI will publish the Alert or Command Response to SDN, using retry mechanism if required, and receive a delivery token. FNV2-FCI will publish a message back to the origination FNV2-SOA client, indicating successful delivery of the Alert or Command Response.
<b>Post-conditions</b>	FNV2-SOA client (ECG) is notified of successful delivery of the Alert or Command Response
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_GEN.16
<b>JIRA</b>	<a href="#">DAWN-672</a>

## 17. SDN/MQTT session authentication

<b>Actors</b>	FNV2-FCI, SDN, ECG
<b>Pre-conditions</b>	MQTT session between FNV2-FCI and SDN requires authentication in order to proceed
<b>Scenario Description</b>	FNV2-FCI will retrieve a unique user id and password from the ECG. The user id is the ESN and the password is some random string. Both, ESN and password/string are flashed in the ECG at the time of manufacture and information about them is maintained in a repository on the Ford backend, which the SDN/MQTT broker can leverage. FNV2-FCI then authenticates itself to the Ford SDN MQTT message broker using the user id and password.
<b>Post-conditions</b>	MQTT session between FNV2-FCI and SDN has been successfully authenticated
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_SEC.01
<b>JIRA</b>	<a href="#">DAWN-673</a>

## 18. TLS shall be used to provide authentication, integrity and confidentiality for the messages exchanged between FNV2-FCI and SDN

<b>Actors</b>	FNV2-FCI, SDN, FNV2-SOA middleware
<b>Pre-conditions</b>	ECG to SDN connection has been established

<b>Scenario Description</b>	<p>The TLS shall be used to provide authentication of senders and recipients, and integrity and confidentiality of the data delivered to and from FNV2-FCI over TCP/IP. Once a secure TLS connection to SDN is established, FNV2-FCI shall authenticate itself to the backend server. It does this using PSK scheme; FNV2-FCI shall have a unique username and password which must be presented to the MQTT broker to connect. The ESN of the TCU shall be used as the MQTT username during authentication. The password is generated by using a prescribed mechanism (A51t).</p> <p>The minimum revision of the TLS protocol implemented shall be TLS1.2. TLS is an application layer protocol based on the top of TCP/IP. Once the client and server have agreed to use TLS, they negotiate a stateful connection by using a handshaking procedure. The protocols use a handshake with an asymmetric cipher to establish cipher settings and a shared key for a session; the rest of the communication is encrypted using a symmetric cipher and the session key.</p> 
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_SEC.02
<b>JIRA</b>	<a href="#">DAWN-674</a>

## 19. SyncP used for FTCP encryption

<b>Actors</b>	FNV2-FCI, SDN
<b>Pre-conditions</b>	MQTT session between FNV2-FCI and SDN is up
<b>Scenario Description</b>	FNV2-FCI will retrieve a unique key for message signing and encryption from the ECG. The unique key is flashed in the ECG at the time of manufacture and information about it is maintained in a repository on the Ford backend, which the SDN/MQTT broker can leverage. FNV2-FCI then signs and encrypts each FTCP message to SDN using SyncP.
<b>Post-conditions</b>	FTCP messages between FNV2-FCI and SDN are SyncP encrypted
<b>List of Exception Use Cases</b>	FOTA
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_SEC.03
<b>JIRA</b>	<a href="#">DAWN-675</a>

## 20. FNV2-FCI shall handle message level security errors

<b>Actors</b>	FNV2-FCI, SDN, FNV2-SOA middleware
---------------	------------------------------------

<b>Pre-conditions</b>	ECG to SDN connection has been established
<b>Scenario Description</b>	<p>FNV2-FCI and SDN shall subscribe to SECURITY/MODULE_ERROR and SECURITY/SERVER_ERROR topics to handle the message level security errors. SECURITY/SERVER_ERROR topic will be used to deliver out of sequence error message from SDN to FNV2-FCI and the SECURITY/MODULE_ERROR topic will be used by FNV2-FCI to report SyncP errors to SDN.</p> <p>FNV2-FCI shall be able to publish errors by using the FTCP ModuleSecurityError message and it shall be able to publish those messages to MQTT broker on the SECURITY/MODULE_ERROR topic. Those messages may be posted with QoS 0. The FNV2-FCI shall be able to receive and process messages from the MQTT broker on the SECURITY/SERVER_ERROR topic.</p> <p>Here are the errors that FNV2-FCI can report to SDN or the SDN may report to FNV2-FCI:</p> <ul style="list-style-type: none"> <li>• INVALID_SECURITY_TYPE</li> <li>• MESSAGE_SIZE_OUT_OF_BOUNDS</li> <li>• INVALID_DATA</li> <li>• GENERIC_SECURITY</li> </ul>
<b>Post-conditions</b>	
<b>List of Exception Use Cases</b>	N/A
<b>Interfaces</b>	
<b>Requirement</b>	FNV2_FCI_SEC.04
<b>JIRA</b>	<a href="#">DAWN-676</a>