

Test case catalog

Test Case Catalog NB-IoT Cloud Connector

Version: 1.0

Status: Final

File name: TestCaseCatalog\_NB-IoT Connector\_final.docx

For: Uday Patil

 P&I-MPP

 Deutsche Telekom AG

Imprint

|  |  |
| --- | --- |
| File name | Document title |
| TestCaseCatalog\_NB-IoT Connector\_final.docx | Test Case Catalog NB-IoT Cloud Connector  |
| Version | Status |
| 1.0 | Final |

Brief description

This document contains the test case catalog for the Cloud Connector for NB-IoT. This document focuses on the MQTT-SN Interface only.

History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modified by | Comment |
| 1.0 | 25th May 2017 | Fabian Knaul | Final version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Versioning Legend

|  |  |  |
| --- | --- | --- |
| Version | Comment | Status |
| 1.0 | Final document which is provided to the customer | released |
|  |  |  |

Authors

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Company | Phone no. | e-Mail |
| Fabian Knaul | Mm1 | +49 160 906 1 2011 | f.knaul@mm1.de |
|  |  |  |  |

Approval

|  |  |  |
| --- | --- | --- |
| Name | Role | Signature |
| Uday Patil | Customer |  |

Table of contents

[Chapter 1. Device registration 6](#_Toc483407974)

[1.1. Request device credentials 6](#_Toc483407975)

[1.2. Pre-provisioning of devices 10](#_Toc483407976)

[Chapter 2. Operation on device 12](#_Toc483408076)

[2.1. Downlink Message 12](#_Toc483408077)

[Chapter 3. Configuration update, OTA, recording of signal strength 14](#_Toc483408078)

[Chapter 4. Reliability – reconnection/send & receive data after connection loss (client/server) 15](#_Toc483408079)

[4.1. Device resends all buffered data after reconnect 15](#_Toc483408080)

[Chapter 5. Send an alarm 18](#_Toc483408081)

[5.1. Create an alarm from the device 18](#_Toc483408082)

[Chapter 6. Send an event – mapping correct/usage of sensor/device management library 21](#_Toc483408183)

[6.1. Send an event from the device 21](#_Toc483408184)

[Chapter 7. Send measurements 24](#_Toc483408185)

[7.1. Send SIMPLE measurement data 24](#_Toc483408186)

[7.2. Send COMPLEX measurement data 26](#_Toc483408187)

[Chapter 8. Session resumption 28](#_Toc483408188)

[8.1. Disconnect and reconnect device 28](#_Toc483408189)

[Chapter 9. Stability test (required for being added to portfolio) 31](#_Toc483408190)

[9.1. Long time test 31](#_Toc483408191)

[9.2. Device handling after power blackout 33](#_Toc483408192)

[Chapter 10. Traffic 36](#_Toc483408193)

[10.1. Check amount of transmitted data 36](#_Toc483408194)

[Chapter 11. Final test activities 40](#_Toc483408253)

[Chapter 12. Further specifications 41](#_Toc483408254)

# Device registration

Please note: This chapter contains the test cases being related to inventory activities as well as self-registration. Please make sure that either the self-registration or the pre-provisioning (or both) via CSV import is supported by your device.

## Request device credentials

| Description: | The test case checks if credentials (IMSI and Password) are generaterd correct and if the device connects correctly to the MQTT-SN Gateway. |
| --- | --- |
| Test data: | * IMSI
* password
 |
| Creation date: | 15th April 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * device is not registered yet
* IMSI is known
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Go to frontend for Cloud of Things: -> Device management -> Devices -> Registration | * Page “Device Registration” is shown and there is an input field called “Device ID”
 |
| Step 2 | Insert the IMSI as DeviceID in the field and click “Register Device” | * Message is shown “Device has been registered and is waiting for being connected!”
 |
| Verification Point 1 | Check if device from Step 2 is displayed in the registration listTodo for manufacturer:Please provide screenshot of Cloud of Things frontend (registration list). | * Unique device ID from step 2 is shown in the registration list.
* The status of the device is “Waiting for connection”
 |
| Step 3 | Device sends the MQTT-SN “CONNECT” Message consisting of IMSI and unique Password The NB-IoT Connector will send following message via REST to the Cloud of Things: https://tsiphoenix.ram.m2m.telekom.com/devicecontrol/deviceCredentials HTTP/1.1Accept-Encoding: gzip,deflateContent-Type: application/jsonContent-Length: <*content length*>Host: tsiphoenix.ram.m2m.telekom.comConnection: Keep-AliveUser-Agent: *<User-Agent>*Cookie: <*cookiedata*>Cookie2: <*cookiedata*>Authorization: <*Authorization data>*{ "id": "*<unique id for device>*"} | * Message is sent successfully
 |
| Verification Point 2 | Check if response for POST-request from step 3 contains an “HTTP 404 Not Found”-message and the text “Credentials for *<unique device id>* are not available. Device is in state PENDING\_ACCEPTANCE, (not ACCEPTED)).”Todo for manufacturer:1. Please provide CONNECT-message from step 3.
2. Please provide response of POST-request from step 3.

Please note: In case your device does not support the possibility to display the response, please provide a short explanation instead. | * Response contains an “HTTP 404 Not Found”-message and text “Credentials for *<unique device id>* are not available. Device is in state PENDING\_ACCEPTANCE, (not ACCEPTED)).”
 |
| Step 4 | Go to frontend for Cloud of Things: -> Device management -> Devices -> Registration | * Page “Device Registration” is shown
* A button “Accept” is displayed right beside the registered device from step 2.
 |
| Step 5 | Click button “Accept” for the registered device from step 2 | * Device status changes from “Pending Acceptance” to “Accepted”
 |
| Verification Point 3 | Check if device status changed from “Pending Acceptance” to “Accepted”.Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (registration list). | * Name of device from step 2 is displayed
* Status of the device is “Accepted”
 |
| Step 6 | Device sends the following POST-request:POSThttps://tsiphoenix.ram.m2m.telekom.com/devicecontrol/deviceCredentials HTTP/1.1Accept-Encoding: gzip,deflateContent-Type: application/jsonContent-Length: <*content length*>Host: tsiphoenix.ram.m2m.telekom.comConnection: Keep-AliveUser-Agent: *<User-Agent>*Cookie: <*cookiedata*>Cookie2: <*cookiedata*>Authorization: <*Authorization data>*{ "id": "*<unique id for device>*"} | * Message sent successfully
 |
| Verification Point 4 | Check if response for POST-request from step 6 contains an “HTTP 201 Created”-message and text "id":"*<unique device id>*", "password":"*<password>*", "self":"*<self url>*","tenantId":"*<tenant>*","username":"*<username>*"Todo for manufacturer:1. Please provide POST-request from step 6.
2. Please provide response of POST-request from step 6.

Please note: In case your device does not support the possibility to display the response, please provide a short explanation instead. | * Response contains an “HTTP 201 Created”-message and correct login credentials ("id":"*<unique device id>*", "password":"*<password>*", "self":"*<self url>*","tenantId":"*<tenant>*","username":"*<username>*")
 |
| Verification Point 5 | Check if the device from step 2 is not displayed in the registration list.Todo for manufacturer:Please provide a screenshot of the frontend for Cloud of Things (registration list). | * The registration list does not contain the device from step 2.
 |
| Step 7 | Go to frontend for Cloud of Things: -> Device management -> Management -> Device credentials | * List of created device logins is shown
 |
| Verification Point 6 | Check if the accepted device from step 2 is shown in the list of accepted devicesTodo for manufacturer:Please provide screenshot of Cloud of Things frontend (device credentials). | * Correct username of the device according to the response for the request from step 6 is displayed in the list of all devices
 |
| Step 8 | Click on the device username being created in step 6 | * Detail page of the device username of the response of step 6 is displayed.
 |
| Verification Point 7 | Check displayed information of detail page of the device of response from step 6Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (detail page of the device credentials). | The detail page of the device of the response of step 6 displays exactly the following information:* unique username
* activity status
* user group
* button to change password
* save button
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Screenshot of Cloud of Things frontend (registration list).
2. CONNECT-message from step 3.
3. Response of CONNECT message- from step 3.
4. Screenshot of Cloud of Things frontend (registration list) after device status has changed to “Accepted”
5. POST-request from step 6.
6. Response of POST-request from step 6.
7. Screenshot of the frontend for Cloud of Things (registration list) after the device has been removed from the registration list.
8. Screenshot of Cloud of Things frontend (device credentials).
9. Screenshot of Cloud of Things frontend (detail page of the device credentials).
 |  |

## Pre-provisioning of devices

| Description: | The test case checks if credentials for pre-provisioned devices can be created successfully via CSV import. At the moment the file has to be sent to martin.buch@t-systems.com for uploading. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device with at least IMSI
* CSV-File (IMSI, self selected password) with at least one dataset (one device)
 |
| Creation date: | 18th May 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * device is not registered yet
* unique id for device is known
* tenant is available
* valid login for Cloud of Things is available
* valid CSV file is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | send .csv File to martin.buch@t-systems.com | * File is uploaded
 |
| Step 2 | file upload is confirmed by martin.buch@t-systems.com |  |
| Step 3 | Go to frontend for Cloud of Things: -> Device management -> Management -> Device credentials | * List of device users is displayed.
 |
| Verification Point 1 | Check if the pre-provisioned device (device\_*<device\_id>*) from step 2 is shown in the list of “Device Users”Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (device credentials). | * Correct username of the device according to the csv file from step 2 is displayed in the list of “Device Users”.
 |
| Step 4 | Click on the device username being created in step 2 | * Detail page of the device username being created in step 2 is displayed.
 |
| Verification Point 2 | Check displayed information of detail page of the device of response from step 2Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (detail page of the device credentials). | The detail page of the device of the response of step 2 displays exactly the following information:* unique username
* activity status
* user group
* button to change password
 |
| Step 5 | Execute test case “Create device in inventory” with credentials from step 2 | * Test case is executed successfully
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Screenshot of Cloud of Things frontend (device credentials).
2. Screenshot of Cloud of Things frontend (detail page of the device credentials).
 |  |

# Operation on device

## Downlink Message

| Description: | The test case checks if an operation can be sent to the device successfully. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device with IMSI number
 |
| Creation date: | 29th April 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI number known
* device was successfully added to inventory in Cloud of Things
* device is able to receive Downlink Messages
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Go to frontend for Cloud of Things: -> Device management -> Devices -> All devicesSelect device from preconditions and go to menu item “Shell” | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all operations which have already been submitted.
* The button “Shell” is available
 |
| Step 2 | Enter an allowed downlink message in the text field. | * The message is shown in the CoT interface.
 |
| Verification Point 1 | Please provide screenshot of Cloud of Things frontend (showing the downlink message). | * Message from step 2 is displayed
 |
| Step 3 | Press the “EXECUTE” button. | * The status of the message is shown in the CoT interface, thje initial status is PENDING
 |
| Verification Point 2 | Check if operation which has been sent in step 3 is displayed in the list of all operationsTodo for manufacturer:Please provide screenshot of Cloud of Things frontend (list of operations from table device control). | * Status of operation from step 2 is displayed.
 |
| Step 4 | Click on name of operation from step 2 | * Info dialog for operation from step 2 is displayed.
 |
| Verification Point 3 | Check if all information which has been transmitted in step 2 are displayed in info dialog of the operationTodo for manufacturer:Please provide screenshot of Cloud of Things frontend (info dialog of operation). | Info dialog shows the following data from step 2:* description of operation
* status of operation
* creation date of operation
 |
| Step 5 | Wait for status change from PENDING to SUCESSFUL | * Device was successfully rebooted
 |
| Verification Point 4 | Check if Status of Operation changed from PENDING to SUCCESSFULTodo for manufacturer:Please provide screenshot of Cloud of Things frontend after Status change (info dialog of operation). | * Info dialog shows the status “SUCCESSFUL” for the operation.
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Screenshot of frontend with the complete downlink message
2. Screenshot of frontend (list of operations from table device control)
3. Screenshot of Cloud of Things frontend (info dialog of operation).
4. Please provide screenshot of Cloud of Things frontend after Status change (info dialog of operation).
 |  |

# Configuration update, OTA, recording of signal strength

N/A

# Reliability – reconnection/send & receive data after connection loss (client/server)

## Device resends all buffered data after reconnect

| Description: | The test case checks if a device is able to reconnect to Cloud of Things after a complete power failure and submit buffered measurements for the period of disconnection.Please note: In case your device is not able to buffer measurements during the time of being disconnected from Cloud of Things, please mark this test case as “n/a” and provide a short explanation. |
| --- | --- |
| Test data: | - login data for Cloud of Things- mobile device with active SIM and IMSI number- mobile device with the ability to buffer measurements for a period of time |
| Creation date: | 09th May 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory in Cloud of Things
* device with the possibility to buffer data
* Required interval for sending measurements is set to e.g. 5 min
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Disconnect the device from the internet to ensure connection lost to Cloud of Things (e.g. remove SIM card or LAN cable) | * Device is no longer connected to the internet or Cloud of Things respectively.
 |
| Step 2 | Go to frontend for Cloud of Things: -> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Step 3 | Go to details of device being disconnected in step 1 | * The info page of the device is displayed.
 |
| Verification Point 1 | Check if the connectivity status of the device is “Offline” after the required interval for sending measurements has passed.Todo for manufacturer:Please provide screenshot from info dialog after disconnection. | * The connectivity status of the device from preconditions is “Offline”.
 |
| Step 4 | Reconnect the device to the internet (e.g. put back SIM card or attach LAN cable) | * Device is automatically reconnected to the internet and to Cloud of Things
 |
| Verification Point 2 | Check device configuration or log for established connectionTodo for manufacturer:Please provide screenshot from device (e.g. logfile or status dialog) | * Connection is established
 |
| Step 5 | Go to frontend for Cloud of Things: -> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Step 6 | Go to details of device being reconnected in Step 5 | * The info page of the device is displayed.
 |
| Verification Point 3 | Check if the connectivity status of the device is “Online” after the required interval for sending measurements has passed.Todo for manufacturer:Please provide screenshot from info dialog after reconnection. | * The connectivity status of the device is “Online”.
 |
| Step 7 | Go to measurements of device being reconnected in step 5 | * The measurement page of the device is displayed
 |
| Verification Point 4 | Check resent measurement data are displayed after the required interval for sending measurements has passed.Todo for manufacturer:Please provide screenshot from measurement dialog. | * Resent measurements are displayed in Cloud of Things
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Screenshot from info dialog after disconnection of device.
2. Screenshot from device (e.g. logfile or status dialog)
3. Screenshot from info dialog after reconnection.
4. Screenshot from measurement dialog.
 |  |

# Send an alarm

## Create an alarm from the device

| Description: | The test case checks if the registered device sends an alarm in case of operational problems (e.g. battery low) correctly. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device IMSI number
 |
| Creation date: | 14th April 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory in Cloud of Things
* The device did not sent an alarm yet.
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Device sends an alarm in the following format:NBIoT/123456789298765/ALM/1The alarm MQTT-SN payload is defined as a String.The definition of the severity levels are as follows:Based on this, the following POST message is generated by the NB-IoT connector:POST-message:POST https://tsiphoenix.ram.m2m.telekom.com/alarm/alarms HTTP/1.1Accept-Encoding: gzip,deflateContent-Type: application/jsonAuthorization: *<Authorization data>*Content-Length: <*content length*>Host: tsiphoenix.ram.m2m.telekom.comConnection: Keep-AliveUser-Agent: *<User-Agent>*{ “source”: { “id”: “*<registered device id>*” }, “text”: “*<alarm message>*”, “time”: “*<timestamp of the alarm>*”, “type”: “*<alarm type>*”, “status”: “*<alarm status>*”, “severity”: “*<alarm severity>*”} | * Message is sent successfully
 |
| Verification Point 1 | Check if response for POST-request from step 1 contains an “HTTP 201 Created”-messageTodo for manufacturer:1. Please provide MQTT-SN request from step 1.
2. Please provide response of POST-request from step 1.

Please note: In case your device does not support the possibility to display the response, please provide a short explanation instead. | * Response contains an “HTTP 201 Created”-message
 |
| Step 2 | Go to frontend for Cloud of Things: -> Device management -> Overviews -> Alarms | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all alarms which have already been raised.
 |
| Verification Point 2 | Check if alarm which has been sent in step 1 is displayed in the list of all alarms with severity from the POST-message from step 1Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (Alarm table). | * Alarm message from POST-request of step 1 is displayed and contains the correct severity.
* Device id from preconditions is displayed below the alarm message.
 |
| Step 3 | Click on alarm message from step 1 | * Info dialog for alarm from step 1 is displayed.
 |
| Verification Point 3 | Check if all information which has been transmitted in POST-request from step 1 is displayed in info dialog of the alarm.Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (info dialog of alarm). | * Info dialog shows exactly the data from POST-request from step 1.
* The history of the alarm is displayed (creation date of alarm, alarm status etc.).
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. MQTT-SN-request from step 1
2. Response of POST-request from step 1
3. Screenshot of frontend (info dialog of alarm)
4. Screenshot of Cloud of Things frontend (info dialog of alarm).
 |  |

# Send an event – mapping correct/usage of sensor/device management library

## Send an event from the device

| Description: | The test case checks if the device is able to send an event correctly to the Cloud of Things (e.g. location update from a GPS sensor). |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device IMSI number
 |
| Creation date: | 29th April 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory in Cloud of Things
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Device sends an EVENT message in the following structure:Following Event IDs are defined:from this, the following POST-messagem is generated by the NB-IoT Connector:POST https://tsiphoenix.ram.m2m.telekom.com/event/events HTTP/1.1Accept-Encoding: gzip,deflateContent-Type: application/jsonAuthorization: *<authorization data>*Content-Length: *<content length>*Host: tsiphoenix.ram.m2m.telekom.comConnection: Keep-AliveUser-Agent: *<User-Agent>*{“source”: {“id”: “*<unique device id>*”},"text": "*<event message>*","time": "*<timestamp of event>",*"type": "*<event type>",* “*<event data>*” }} | * Message is sent successfully
 |
| Verification Point 1 | Check if response for POST-request from step 1 contains an “HTTP 201 Created”-messageTodo for manufacturer:1. Please provide MQTT-SN-request from step 1.
2. Please provide response of POST-request from step 1.

Please note: In case your device does not support the possibility to display the response, please provide a short explanation instead. | Response contains an “HTTP 201 Created”-message |
| Step 2 | Go to frontend for Cloud of Things: -> Device management -> Overviews -> Events | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all events which have already been sent.
 |
| Verification Point 2 | Check if event which has been sent in step 1 is displayed in the list of all events from the POST-message from step 1Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (event table). | * Event message from POST-message of step 1 is displayed.
* Device id from preconditions is displayed below the event message.
 |
| Step 3 | Click on event message from step 1 | * Info dialog for event from step 1 is displayed.
 |
| Verification Point 3 | Check if all information which has been transmitted in POST-request from step 1 is displayed in info dialog of the event.Todo for manufacturer:Please provide screenshot of Cloud of Things frontend (info dialog of event). | * Info dialog shows exactly the data from POST-request from step 1.
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. MQTT-SN-request from step
2. Response of POST-request from step 1
3. Screenshot of frontend (event table)
4. Screenshot of Cloud of Things frontend (info dialog of event).
 |  |

# Send measurements

## Send SIMPLE measurement data

| Description: | The test case checks if measurement data can be transmitted successfully from the to the Cloud of Things. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device IMSI number
 |
| Creation date: | 10th May 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory and is connected to Cloud of Things
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Device sends a simple measurement following this structure: From this the following POST-request is generaterd by the NB-IoT connector:POST https://tsiphoenix.ram.m2m.telekom.com/measurement/measurements HTTP/1.1Accept-Encoding: gzip,deflateContent-Type: application/jsonAuthorization: *<authorization data>*Content-Length: *<content length>*Host: tsiphoenix.ram.m2m.telekom.comConnection: Keep-AliveUser-Agent: *<User-Agent>*{ "source": { "id": "*<device id>*" }, "time": "*<current timestamp>*", "type": "*<device type>*", "c8y\_*<Measurementtype>*": { "*<Measurement>*": { "value": *<value>*, "unit": "*<Unit>*" } }} | * Message is sent successfully
 |
| Verification Point 1 | Check if response for POST-request from step 1 contains an “HTTP 201 Created”-messageTodo for manufacturer:1. Please provide MQTT-SN-request from step 1.
2. Please provide response of POST-request from step 1.

Please note: In case your device does not support the possibility to display the response, please provide a short explanation instead. | * Response contains an “HTTP 201 Created”-message and the correct externalId
 |
| Step 2 | Go to frontend for Cloud of Things: -> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Step 3 | Click on name of device used in step 1 | * Info dialog for device from step 1 is displayed.
 |
| Step 4 | Click on “Measurements” in the menu list | * Diagram “*<Measurementtype>*” is displayed
 |
| Verification Point 2 | Check if measurement created in step 1 is displayedTodo for manufacturer:Please provide screenshot of Cloud of Things frontend (diagram “*<Measurementtype>*”). | * Correct diagram “*<Measurementtype>*” of device from step 1 is displayed.
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. MQTT-SN-request from step 1
2. Response of POST-request from step 1
3. Screenshot of frontend (Diagram Measurement)
 |  |

## Send COMPLEX measurement data

| Description: | The test case checks if measurement data can be transmitted successfully from the to the Cloud of Things. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device IMSI number
 |
| Creation date: | 10th May 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory and is connected to Cloud of Things
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Device sends a complex measurement following this structure: and using Payload ID 186A1 or 186A2. From this the following POST-request is generaterd by the NB-IoT connector:Device sends the following POST-request:POST https://tsiphoenix.ram.m2m.telekom.com/measurement/measurements HTTP/1.1Accept-Encoding: gzip,deflateContent-Type: application/jsonAuthorization: *<authorization data>*Content-Length: *<content length>*Host: tsiphoenix.ram.m2m.telekom.comConnection: Keep-AliveUser-Agent: *<User-Agent>*{ "source": { "id": "*<device id>*" }, "time": "*<current timestamp>*", "type": "*<device type>*", "c8y\_*<Measurementtype>*": { "*<Measurement>*": { "value": *<value>*, "unit": "*<Unit>*" } }} | * Message is sent successfully
 |
| Verification Point 1 | Check if response for POST-request from step 1 contains an “HTTP 201 Created”-messageTodo for manufacturer:1. Please provide MQTT-SN-request from step 1.
2. Please provide response of POST-request from step 1.

Please note: In case your device does not support the possibility to display the response, please provide a short explanation instead. | * Response contains an “HTTP 201 Created”-message and the correct externalId
 |
| Step 2 | Go to frontend for Cloud of Things: -> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Step 3 | Click on name of device used in step 1 | * Info dialog for device from step 1 is displayed.
 |
| Step 4 | Click on “Measurements” in the menu list | * Diagram “*<Measurementtype>*” is displayed
 |
| Verification Point 2 | Check if measurement created in step 1 is displayedTodo for manufacturer:Please provide screenshot of Cloud of Things frontend (diagram “*<Measurementtype>*”). | * Correct diagram “*<Measurementtype>*” of device from step 1 is displayed.
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. MQTT-SN-request from step 1
2. Response of POST-request from step 1
3. Screenshot of frontend (Diagram Measurement)
 |  |

# Session resumption

## Disconnect and reconnect device

| Description: | The test case checks the behavior of a device that is first being disconnected and afterwards reconnected to the Cloud of Things. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device with IMSI number
 |
| Creation date: | 04th May 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory in Cloud of Things and is sending measurement messages according to the expected time interval for sending messages in Cloud of Things
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Go to frontend for Cloud of Things:-> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Verification Point 1 | Check if the device from preconditions is displayed in the list of all devices and if its connectivity status is “Online”Todo for manufacturer:1. Please provide the name of the device from preconditions.
2. Please provide the configured expected time interval for sending messages for the device from preconditions.
3. Please provide screenshot of frontend for Cloud of Things (list of all devices).
 | * Name of device from preconditions is displayed and its connectivity status is “Online”.
 |
| Step 2 | Disconnect the device from preconditions from the Cloud of Things and wait until the expected time interval for sending messages has elapsed. | Device from preconditions is disconnected. |
| Verification Point 2 | Check if the connectivity status of the device from preconditions is “Offline”.Todo for manufacturer:Please provide screenshot of frontend of Cloud of Things (list of all devices) during disconnect. | Connectivity status of the device is “Offline”. |
| Step 3 | Go to frontend for Cloud of Things: -> Device management -> Overviews -> Alarms | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all alarms which have already been raised.
 |
| VerificationPoint 3 | Check if any unavailability alarm with severity “Major” has been raised by the device from preconditions.Todo for the manufacturer:Please provide screenshot of frontend for Cloud of Things (list of all alarms) | The list of all alarms contains an unavailability alarm with severity “Major” from the device from preconditions.  |
| Step 4 | Go to-> Device management -> Devices -> All devicesSelect the device from preconditions and click on the menu item “Measurements”. | An overview of measurement data of the device from preconditions is displayed. |
| Verification Point 4 | Check if the diagrams in the overview of measurement data of the device from preconditions contain measurement data for the last time interval for sending.Todo for the manufacturer:Please provide screenshot of frontend for Cloud of Things (Measurements overview) during disconnect. | The diagrams does not contain measurement data for the last time interval for sending messages. |
| Step 5 | Reconnect the device from preconditions to the Cloud of Things. | Device from preconditions is reconnected. |
| Verification Point 5 | Check if the diagrams in the overview of measurement data of the device from preconditions contain measurement data for the last time interval for sending.Todo for the manufacturer:1. Please provide screenshot of frontend for Cloud of Things (Measurements overview) after reconnection.
2. Please describe the behavior of the device after it has been reconnected to the Cloud of Things (e.g. measurement data during disconnect are stored and sent after a connection has been established again, measurement data during disconnect is lost and not sent to the Cloud of Things after reconnect etc.)
 | The diagrams contain measurement data for the last time interval for sending messages. |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Name of the device from preconditions.
2. Configured expected time interval for sending messages for the device from preconditions.
3. Screenshot of frontend for Cloud of Things (list of all devices) before disconnecting the device.
4. Screenshot of frontend of Cloud of Things (list of all devices) during disconnect.
5. Screenshot of frontend for Cloud of Things (list of all alarms)
6. Screenshot of frontend for Cloud of Things (Measurements overview) during disconnect.
7. Screenshot of frontend for Cloud of Things (Measurements overview) after reconnection.
8. Description of the behavior of the device after it has been reconnected to the Cloud of Things (e.g. measurement data during disconnect are stored and sent after a connection has been established again, measurement data during disconnect is lost and not send to the Cloud of Things after reconnect etc.)
 |  |

# Stability test (required for being added to portfolio)

## Long time test

| Description: | The test case checks the connectivity of the device to the Cloud of Things during a period of 24 hours. |
| --- | --- |
| Test data: | * login data for Cloud of Things

- device IMSI* already registered device in the Cloud of Things
 |
| Creation date: | 21st April 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory in Cloud of Things
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
* an expected time interval for sending messages for the device has been configured in the Cloud of Things, time interval should be at least 5 minutes
* the device is already running for at least 24 hours and sending measurement messages according to the expected time interval for sending messages in Cloud of Things
 |  |
| Step 1 | Go to frontend for Cloud of Things:-> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Verification Point 1 | Check if the device from preconditions is displayed in the list of all devices and if its connectivity status is “Online”.Todo for manufacturer:1. Please provide the name of the device from preconditions.
2. Please provide the configured expected time interval for sending messages for the device from preconditions.
3. Please provide screenshot of frontend for Cloud of Things (list of all devices).
 | * Name of device from preconditions is displayed and its connectivity status is “Online”.
 |
| Step 2 | Go to  -> Device management -> Overviews -> Alarms | * A list is displayed containing all alarms which have already been raised.
 |
| VerificationPoint 2 | Check if any unavailability alarm with severity “Major” has been raised by the device from preconditions.Todo for the manufacturer:Please provide screenshot of frontend for Cloud of Things (list of all alarms) | * The list of all alarms does not contain an unavailability alarm with severity “Major” from the device from preconditions.
 |
| Step 3 | Go to-> Device management -> Devices -> All devicesSelect the device from preconditions and click on the menu item “Measurements”. | * An overview of measurement data of the device from preconditions is displayed.
 |
| Verification Point 3 | Check if the diagrams in the overview of measurement data of the device from preconditions contain measurement data for the last 24 hours according to the configured time interval for sending messages.Todo for the manufacturer:Please provide screenshot of frontend for Cloud of Things (Measurements overview) | * The diagrams contain valid measurement data for the last 24 hours according to the configured time interval for sending messages.
 |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Name of the device from preconditions
2. Configured expected time interval for sending messages for the device from preconditions
3. Screenshot of frontend for Cloud of Things (list of all devices).
4. Screenshot of frontend for Cloud of Things (list of all alarms)
5. Screenshot of frontend for Cloud of Things (Measurements overview)
 |  |

## Device handling after power blackout

| Description: | The test case checks the behavior of a device after a power blackout and the disconnection to Cloud of Things. |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device IMSI
 |
| Creation date: | 04th May 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device was successfully added to inventory in Cloud of Things and is sending measurement messages according to the expected time interval for sending messages in Cloud of Things
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
 |  |
| Step 1 | Go to frontend for Cloud of Things:-> Device management -> Devices -> All devices | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all devices which have already been created.
 |
| Verification Point 1 | Check if the device from preconditions is displayed in the list of all devices and if its connectivity status is “Online”Todo for manufacturer:1. Please provide the name of the device from preconditions.
2. Please provide the configured expected time interval for sending messages for the device from preconditions.
3. Please provide screenshot of frontend for Cloud of Things (list of all devices).
 | * Name of device from preconditions is displayed and its connectivity status is “Online”.
 |
| Step 2 | Disconnect the device from preconditions from power circuit and wait until the expected time interval for sending messages has elapsed. | Device from preconditions is powered down. |
| Verification Point 2 | Check if the connectivity status of the device from preconditions is “Offline”.Todo for manufacturer:Please provide screenshot of frontend of Cloud of Things (list of all devices) during disconnect. | Connectivity status of the device is “Offline”. |
| Step 3 | Go to frontend for Cloud of Things: -> Device management -> Overviews -> Alarms | * The frontend is displayed.
* The user is already logged in.
* A list is displayed containing all alarms which have already been raised.
 |
| VerificationPoint 3 | Check if any unavailability alarm with severity “Major” has been raised by the device from preconditions.Todo for the manufacturer:Please provide screenshot of frontend for Cloud of Things (list of all alarms) | The list of all alarms contains an unavailability alarm with severity “Major” from the device from preconditions.  |
| Step 4 | Go to-> Device management -> Devices -> All devicesSelect the device from preconditions and click on the menu item “Measurements”. | An overview of measurement data of the device from preconditions is displayed. |
| Verification Point 4 | Check if the diagrams in the overview of measurement data of the device from preconditions contain measurement data for the last time interval for sending.Todo for the manufacturer:Please provide screenshot of frontend for Cloud of Things (Measurements overview) during disconnect. | The diagrams does not contain measurement data for the last time interval for sending messages. |
| Step 5 | Reconnect the device from preconditions to the power circuit. | Device from preconditions is repowered. |
| Verification Point 5 | Check if the diagrams in the overview of measurement data of the device from preconditions contain measurement data for the last time interval for sending.Todo for the manufacturer:1. Please provide screenshot of frontend for Cloud of Things (Measurements overview) after reconnection.
2. Please describe the behavior of the device after it has been reconnected to the Cloud of Things (e.g. measurement data during disconnect are stored and sent after a connection has been established again, measurement data during disconnect is lost and not sent to the Cloud of Things after reconnect etc.)
 | The diagrams contain measurement data for the last time interval for sending messages. |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Name of the device from preconditions.
2. Configured expected time interval for sending messages for the device from preconditions.
3. Screenshot of frontend for Cloud of Things (list of all devices) before disconnecting the device.
4. Screenshot of frontend of Cloud of Things (list of all devices) during disconnect.
5. Screenshot of frontend for Cloud of Things (list of all alarms)
6. Screenshot of frontend for Cloud of Things (Measurements overview) during disconnect.
7. Screenshot of frontend for Cloud of Things (Measurements overview) after reconnection.
8. Description of the behavior of the device after it has been reconnected to the Cloud of Things (e.g. measurement data during disconnect are stored and sent after a connection has been established again, measurement data during disconnect is lost and not send to the Cloud of Things after reconnect etc.)
 |  |

# Traffic

## Check amount of transmitted data

| Description: | The test case checks and logs the amount of data being transmitted between Cloud of Things and the device during the following activities:* Request device credentials
* Send a downlink message
* Create an alarm from the device
* Send an event from the device
* Send a simple measurement
* Send a complex measurement
* Send an event from the device
 |
| --- | --- |
| Test data: | * login data for Cloud of Things
* device IMSI
* For determining the transmitted amount of data for “Recording of signal strength”, the device should be a mobile device with an active SIM card.
 |
| Creation date: | 26th April 2017 |
| Status: | Ready |
| Step Name | Description | Expected |
| Precondition | * IMSI is known
* device is not registered yet was successfully added to inventory in Cloud of Things
* For determining the transmitted amount of data for “Recording of signal strength”, the device should have an active SIM and is booked in a mobile network. If not, please skip this step in the test execution and provide a short explanation.
* tenant is available
* valid login for Cloud of Things is available
* user is already logged in to Cloud of Things
* Please make sure that there are no other activities on the device.
 |  |
| Step 1 | Start logging of transmitted data between Cloud of Things and the device from preconditionsNote: Use network statistics from the device. Please note that the measurement has to include the ACK message, if applicable. | Logging is started. |
| Step 2 | Execute test case “Request device credentials” for the device from preconditions. | Test case “Request device credentials” has been executed successfully. |
| Verification Point 1 | Check amount of data that has been transmitted during execution of test case “Request device credentials” for the device from preconditions.Todo for manufacturer:Please provide the amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Request device credentials”. |  |
| Step 3  | Execute test case “Send a downlink message” for the alarm from step 7 for the device from preconditions. | Test case “Send a downlink message” has been executed successfully. |
| Verification Point 2 | Check amount of data that has been transmitted during execution of test case “Send a downlink message” for the device from preconditions.Todo for manufacturer:Please provide the amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send a downlink message”. |  |
| Step 4 | Execute test case “Create an alarm from the device” for the device from preconditions. | Test case “Create an alarm from the device” has been executed successfully. |
| Verification Point 3 | Check amount of data that has been transmitted during execution of test case “Create an alarm from the device” for the device from preconditions.Todo for manufacturer:Please provide the amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Create an alarm from the device”. |  |
| Step 5 | Execute test case “Send an event from the device” for the device from preconditions. | Test case “Send an event from the device” has been executed successfully. |
| Verification Point 4 | Check amount of data that has been transmitted during execution of test case “Send an event from the device” for the device from preconditions.Todo for manufacturer:Please provide the amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send an event from the device”. |  |
| Step 6 | Execute test case “Send a simple measurement” for the device from preconditions. | Test case “Send a simple measurement” has been executed successfully. |
| Verification Point 5 | Check amount of data that has been transmitted during execution of test case “Send a simple measurement” for the device from preconditions.Todo for manufacturer:Please provide the amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send a simple measurement”. |  |
| Step 7 | Execute test case “Send a complex measurement” for the device from preconditions | Test case “Send a complex measurement” has been executed successfully. |
| Verification Point 6 | Check amount of data that has been transmitted during execution of test case “Send a complex measurement” for the device from preconditionsTodo for manufacturer:Please provide the amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send a complex measurement”. |  |
| Summary | As a result of this test case, the following deliverables have to be provided by the manufacturer:1. Amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Request device credentials”.
2. Amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send a downlink message”
3. Amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Create an alarm from the device”.
4. Amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send an event from the device”.
5. Amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send a simple measurement”.
6. Amount of data that has been transmitted between NB-IoT Connector and the device during execution of test case “Send a complex measurement”.
 |  |

# Final test activities

This chapter contains the activities which need to be done for cleaning up the environment after your test executions. Make sure that all test cases have been executed and that the results have been documented before executing the final test activities.

The Chapter will be updated as soon as the test case “Delete device in the inventory” is available.

# Further specifications

Please provide device specification and an overview which operations, events and measurements (type, measurement, units, etc.) are supported by your devices.