NUKI
THE SMART LOCK

MQTT API

V1.4

23.2.2023

Nuki Home Solutions GmbH
Münzgrabenstrasse 92/4, 8010 Graz

1. Introduction

The MQTT API offers the possibility to connect supported Nuki products to an MQTT server in order to allow basic control of them, similar to the functionality available via the Bridge HTTP-API such as retrieving the current lock state and performing lock operations.

Supported product is currently only the Smart Lock 3.0 Pro.

Check for the latest version of this document at our Developer Platform.

2. Provisioning & API Activation

2.1 Fixed Credentials in Firmware 3.5.11

If the Smart Lock 3.0 Pro runs firmware 3.5.11 and if the debug mode of the Smart Lock is enabled, a connection to a hardcoded MQTT server will be automatically established using the following login credentials:

Server: *mqtt.local* (mDNS) or *mqtt* (DNS). The Smart Lock first tries to resolve *mqtt.local* via mDNS. If this does not work, it tries to resolve *mqtt* via DNS. Connection attempts are made twice per hour and immediately after every successful WiFi reconnect.

Port: *1883*
Username: *nuki*

Password: SHA256 hash of the WiFi Password stored in the Smart Lock WiFi settings. The SHA256 hash has to be lowercase.

Example:
WIFI Password = *1234567890*
SHA256 = *c775e7b757ede630cd0aa1113bd102661ab38829ca52a6422ab782862f268646*

Source: https://emn178.github.io/online-tools/sha256.html

Remarks for the fixed credentials implementation:

- How to enable Debug mode:
  Tap 7x on the Settings > Features & Configuration > “NUKI SMART LOCK” headline.
- After enabling debug mode, it can take up to one minute until the Smart Lock

connects or disconnects from the MQTT server.
- When the debug mode of the Smart Lock and “LED signal on the Smart Lock” are both active, the red LED of the Smart Lock acts as a traffic indicator for incoming WIFI packets, similar to a traffic indicator of a network switch. Turn off “LED signal on the Smart Lock” to disable this diagnostic feature.
- Home Assistant Discovery and “Allow Locking” are per default activated and can not be deactivated.

2.2 Provisioning

Provisioning of an MQTT server is done via the Nuki App in the Device Administration > MQTT section.

It allows users to enable/disable the API, enter the credentials (username & password) and a hostname or IPv4 address for the MQTT server.

In addition Home Assistant Discovery (Default: on) and “Allow Locking” (Default: on) can be activated and deactivated.

2.3 Connection

Once the API has been activated and provisioned the Nuki device tries to establish a connection to the entered MQTT server on port 1883.

Connections will only be established to MQTT servers in the local LAN. I.e. either the hostname has to resolve to a local IP or a local IP is given as hostname (10.0.0.0 - 10.255.255.255, 172.16.0.0 - 172.31.255.255, 192.168.0.0 - 192.168.255.255).

The current status of the connection is visible inside the Nuki Apps Device Administration MQTT section.

The reconnect mechanism for the Smart Lock 3.0 WiFi connection is bound to a successful connection to the Nuki server. i.e. you can not isolate the Smart Lock from the internet as this will lead to reconnect attempts involving WiFi log off/ons with exponentially growing downtimes in between, which will also lead to disconnects of the MQTT connection. Likewise an unstable internet connection can lead to MQTT reconnects and downtimes.

2.4 Security

The MQTT API inside Smart Lock 3.0 Pro does not support encrypted connections because of memory constraints. Only connections to unencrypted MQTT servers are possible.

It is the sole responsibility of the user to secure the transportation layer (e.g. WiFi, LAN) against misuse. Do not use the MQTT API if you can not guarantee the integrity of your LAN.

The “Allow locking” setting in the provisioning flow allows to restrict the Smart Lock to only publish information. If this flag is disabled it does not subscribe to the topics needed for device control. I.e. it is not possible to send lock commands to the Smart Lock anymore in this mode.

3. Data Models

3.1 Device Types

ID	name
0	Smart Lock
2	Opener
3	Smart Door
4	Smart Lock 3.0 (Pro)

3.2 Modes

mode	smartlock	opener	Description
2	door mode	door mode	Operation mode after complete setup
3	-	continuous mode	Ring to Open permanently active

3.3 Lock States

ID	smartlock	opener
0	uncalibrated	untrained
1	locked	online
2	unlocking	-
3	unlocked	rto active
4	locking	-
5	unlatched	open
6	unlocked (lock ‘n’ go)	-
7	unlatching	opening
253	-	boot run
254	motor blocked	-
255	undefined	undefined

3.4 Lock Actions

ID	smartlock	opener
1	unlock	activate rto
2	lock	deactivate rto
3	unlatch	electric strike actuation
4	lock ‘n’ go	activate continuous mode
5	lock ‘n’ go with unlatch	deactivate continuous mode
6	full lock	
80	fob (without action)	fob (without action)
90	button (without action)	button (without action)

3.5 Simple Lock Actions

Possible outcome of a simple lock action (mapping handled in the firmware of the device):

action	smartlock / knob	smartlock / handle	opener
lock	lock	lock	deactivate rto and cm
unlock	unlatch	unlock	open

3.6 Doorsensor States

ID	name
1	deactivated
2	door closed
3	door opened
4	door state unknown
5	calibrating
16	Uncalibrated
240	Tampered
255	Unknown

3.7 Trigger

ID	name
0	system / bluetooth command
1	(reserved)
2	button
3	automatic (e.g. time control)
6	auto lock
171	HomeKit
172	MQTT

4. Topics

4.1 Topic Structure

Each Nuki device publishes to the same structure of topics:

nuki/nuki_id_in_hex/Topic

The Nuki ID in hexadecimal format is printed on the device itself and also shown in the device administration. e.g. 2BB28570.

4.2 Published Topics for Device States

The following topic structure is available per device and updated whenever an update to a device state occurs. In addition the “last updated” timestamp is changed with every update. The retain flag is activated with all topics and QOS = 0 is used.

Topic	Description	Example
deviceType	Nuki device type (see Device Types) Beta: Only device Type 4 = Smart Lock 3.0 Pro is supported	4
name	Name of the device	Home door
firmware	Current firmware version of the device	3.2.0
mode	ID of the lock mode (see Modes)	2
state	ID of the lock state (see Lock States)	1
batteryCritical	Flag indicating if the batteries of the	true

	Nuki device are at critical level	
batteryChargeState	Value representing the current charge status in %	18
batteryCharging	Flag indicating if the batteries of the Nuki device are charging at the moment	false
keypadBatteryCritical	Flag indicating if the batteries of the paired Nuki Keypad are at critical level	false
doorsensorState	ID of the door sensor state	2
doorsensorBatteryCritical	Flag indicating if the batteries of the paired Nuki Door Sensor are at critical level	false
ringactionTimestamp	Timestamp of the last ring-action. Only for Nuki Opener.	2018-10-03T06:49:00+00:00
serverConnected	Connection state to the Nuki server.	true
timestamp	Timestamp of the retrieval of the last update	2018-10-03T06:49:00+00:00
connected	Indicates if the device is currently connected to the MQTT server or not. Uses “false” as the last will message, which will be set by the mqtt server automatically if the device disconnects.	true

4.3 Published and Subscribed Topics for Device Control

The following topic structure allows to send commands to the device via a topic to which the device subscribes. For all messages QOS = 2 is used. The retain flag is not set.

If “Allow locking” is disabled during provisioning, the device does not subscribe to the lockAction, lock and unlock topics.

Topic	Description	Example
lockAction	ID of the desired Lock Action. Only actions 1-6 are supported.	1
lock	Set to “true” to execute the simple lock action “lock”	true
unlock	Set to “true” to execute the simple lock action “unlock”	true
commandResponse	The Nuki device publishes to this topic the return code of the last command it executed: 0 = Success 1-255 = Error code as described in the BLE API. Note: Nuki devices can only process one command at a time. If several commands are sent in parallel the commandResponses might overlap.	0
lockActionEvent	The Nuki device publishes to this topic a comma separated list whenever a lock action is about to be executed: ● LockAction ● Trigger ● Auth-ID: Auth-ID of the user ● Code-ID: ID of the Keypad code, 0 = unknown ● Auto-Unlock (0 or 1) or number of button presses (only button & fob actions) or Keypad source (0 = back key, 1 = code, 2 = fingerprint) ● Only lock actions that are attempted to be executed are reported. E.g. unsuccessful Keypad code entries or lock commands outside of a time window are not published.	Unlatch via Keypad with Auth-ID 54321 from Code-ID 12345: 3,0,54321,12345,1 Auto-Unlock via App from Auth-ID 54322: 1,0,54322,0,1 Lock’n Go via Button: 4,2,0,0,0 Button configured to “no action on double click” and pressed twice: 90,2,0,0,2 Fob with auth-id 54322 configured to “unlatch” on triple click and pressed 3x: 3,3,54322,0,3

5. Home Assistant Discovery

The Nuki MQTT API supports the MQTT Discovery of Home Assistant which enables Home Assistant to automatically add devices to its database once they are connected to the MQTT server. It can also be used from other integrations to automatically discover Nuki devices that connect to an MQTT server.

Home Assistant Discovery can be activated during provisioning.

When Home Assistant Discovery is activated the Nuki device publishes on each MQTT connect messages into the relevant topics with the prefix “homeassistant” which is hardcoded and can not be changed.

Those topics are retained by the MQTT server (i.e. the retain flag is set for the …/configure topic). If the Nuki device is removed from the MQTT server the retained topics will have to be manually deleted in order to remove the devices fully from Home Assistant. Disabling Home Assistant Auto Discovery or the MQTT API with the Nuki App while the Nuki device is connected to the MQTT server will also attempt to clear the retained topics once by sending empty messages to the retained …/configure topics.

5.1 Smart Lock

The following devices are published into the respective topic:
homeassistant/device_type/nuki_id_in_hex/nuki_name/config

Name	**Device Type**	**Description**
Smart Lock	Lock	Lock commands & states
Smart Lock Battery	Sensor	Smart Lock battery in %
Smart Lock Battery Critical	Binary_sensor	Smart Lock battery critical flag
Smart Lock Battery Charging	Binary_sensor	Smart Lock battery is charging flag
Smart Lock Unlatch	Button	Button for unlatch command

Smart Lock Lock’n’Go	Button	Button for lock’n’go action
Smart Lock Lock’n’Go with Unlatch	Button	Button for lock’n’go with unlatch
Smart Lock Door Sensor	Binary_sensor	Door state. Only if a Nuki Door Sensor is paired
Smart Lock Door Sensor Battery Critical	Binary_sensor	Door sensor battery critical flag. Only if a Nuki Door Sensor is paired.
Smart Lock Keypad Battery Critical	Binary_sensor	Keypad battery critical. Only if a Nuki Keypad is paired.

6. Changelog

1.4 - 23.2.2023

- Added HomeKit and MQTT to triggers. Lock actions originated via HomeKit and MQTT now also trigger a lockActionEvent
- Added provisioning & security remarks
- Added “Allow locking” flag
- Added Home Assistant Discovery
- Changed fixed credentials mode from beta to release firmware 3.5.11

1.3 - 27.10.2022

- Added lockActionEvent
- Added Triggers
- Added Button & Fob (without action) to Lock Actions
- Document clean up

1.2 - 27.9.2022

- Changed hostname for dns resolution
- Added remarks for beta implementation

1.1 - 26.7.2022

- Removed ringactionstate
- Removed that serverConnected is limited to SL3P. In case of a Nuki bridge holding the MQTT connection, serverConnected would mirror the connection state of the bridge.

1.0 - 19.7.2022

- Initial version