

# Nuki Bridge API

V1.8

06.03.2019

Nuki Home Solutions GmbH  
Münzgrabenstrasse 92/4, 8010 Graz

1. Introduction
2. Calling URL
  - 2.1 Example
3. Bridge discovery & API activation
  - 3.1 Example
  - 3.2 Token
4. Lock states
  - 4.1 Lock actions
5. Endpoints
  - /auth
  - /configAuth
  - /list
  - /lockState
  - /lockAction
  - /unpair
  - /info
  - /callback
    - /callback/add
    - /callback/list
    - /callback/remove
6. Maintenance endpoints
  - /log
  - /clearlog
  - /fwupdate
  - /reboot
  - /factoryReset
7. Changelog

# 1. Introduction

The REST API on the Nuki Bridge offers simple endpoints to list all available Nuki Smart Locks, retrieve their current lock state and perform lock operations.

When using the Nuki Software Bridge, all configuration is done inside the Nuki Bridge App instead of the Nuki App.

## 2. Calling URL

This is the address used to call the available services of the internal webserver.

The IP address is shown in the bridge settings within the Nuki App or can be retrieved from the bridge discovery URL.

The server is listening for incoming requests either on default port 8080 or the configured one if it has been modified within the Nuki App.

### 2.1 Example

The following base url will be used in upcoming examples:

<http://192.168.1.50:8080/>

## 3. Bridge discovery & API activation

Calling the URL <https://api.nuki.io/discover/bridges> returns a JSON array with all bridges which have been connected to the Nuki Servers through the same IP address than the one calling the URL within the last 30 days. The array contains the local IP address, port, the ID of each bridge and the date of the last change of the entry in the JSON array.

### 3.1 Example

```
{
  "bridges": [
    {
      "bridgeId": "2117604523", "ip": "192.168.1.50", "port": 8080, "dateUpdated": "2017-06-14T06:53:44Z"
    }
  ],
  "errorCode": 0
}
```

Once a bridge has been discovered on the LAN the API can be activated and the [API token](#) retrieved by calling the `/auth` command. The user has to confirm this request by pressing the button on the bridge. For more details see the description of the `/auth` command. Alternatively you can activate the API and set the token by managing the bridge in the Nuki App.

If discovery is disabled via `/configAuth` or through the Nuki App, the IP is 0.0.0.0 and the port 0. In this case the `/auth` command fails with HTTP error 403.

## 3.2 Token

We offer two ways of verifying calls to endpoints with a token:

Method	Usage
Plain token	You can use the plain token for testing and in private, secured WIFIs or VLANs.
Hashed token	Use if you do not want to send the plain token within your API-calls.  <b>Note:</b> Only available for the hardware bridge for now.

Parameters:

Name	Parameter	Values	Example
Plain token	token	uint8[20]	123456
Timestamp	ts	YYYY-MM-DDTHH:MM:SSZ	2019-03-05T01:06:53Z
Random Number	rnr	number from 0 to 65535	4711
Hash	hash	sha256("ts,rnr,token")	f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6

```
sha256("2019-03-05T01:06:53Z,4711,123456") =
f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6
```

Example-Calls:

**Plain token:**

<http://192.168.1.50:8080/info?token=123456>

**Hashed token:**

<http://192.168.1.50:8080/info?ts=2019-03-05T01:06:53Z&rnr=4711&hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6>

A hashed token will only be valid with a sufficiently current timestamp and can not be reused, to prevent replay attacks. So making two calls with the exact same timestamp will only work with different random numbers.

To debug problems with non synchronous times you can check the current time on the bridge via [bridge discovery](#).

## 4. Lock states

Possible lock states (used in [Endpoints](#) below):

ID	name
0	uncalibrated
1	locked
2	unlocking
3	unlocked
4	locking
5	unlatched
6	unlocked (lock 'n' go)
7	unlatching
254	motor blocked
255	undefined

## 4.1 Lock actions

Possible lock actions (used in [Endpoints](#) below):

ID	name
1	unlock
2	lock
3	unlatch
4	lock 'n' go
5	lock 'n' go with unlatch

## 5. Endpoints

/auth

<b>URL</b>	<a href="http://192.168.1.50:8080/auth">http://192.168.1.50:8080/auth</a>	
<b>Usage</b>	Enables the api (if not yet enabled) and returns the api token. If no api token has yet been set, a new (random) one is generated.  When issuing this API-call the bridge turns on its LED for 30 seconds. The button of the bridge has to be pressed within this timeframe. Otherwise the bridge returns a negative success and no token.	
<b>Response</b>	JSON list containing the success of the authorization	
	<b>token</b>	The api token
	<b>success</b>	Flag indicating the success of the authorization
<b>Errors</b>	<b>HTTP 403</b>	Returned if the authentication is disabled

<b>Example-Call</b>	<a href="http://192.168.1.50:8080/auth">http://192.168.1.50:8080/auth</a>
<b>Example-Response</b>	{ "token": "token123", "success": true }

## /configAuth

<b>URL</b>	<a href="http://192.168.1.50:8080/configAuth">http://192.168.1.50:8080/configAuth</a>	
<b>Usage</b>	Enables or disables the authorization via <a href="#">/auth</a> and the publication of the local IP and port to the discovery URL ( <a href="https://api.nuki.io/discover/bridges">https://api.nuki.io/discover/bridges</a> ).	
<b>URL-Parameters</b>	<b>enable</b>	Flag (0 or 1) indicating whether or not the authorization should be enabled
	<b>token</b> or <b>hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON list containing the success of the operation	
	<b>success</b>	Flag indicating the success of the authorization
<b>Errors</b>	<b>HTTP 400</b>	Returned if the given value for <b>enable</b> is invalid (neither 0 nor 1)
	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/configAuth?enable=0&amp;token=123456">http://192.168.1.50:8080/configAuth?enable=0&amp;token=123456</a> <a href="http://192.168.1.50:8080/configAuth?enable=0&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/configAuth?enable=0&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	{ "success": true }	

## /list

<b>URL</b>	<a href="http://192.168.1.50:8080/list">http://192.168.1.50:8080/list</a>		
<b>Usage</b>	Returns a list of all paired Smart Locks		
<b>URL-Parameters</b>	<b>token</b> or <b>hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.	
<b>Response</b>	JSON array. One item of the following per Smart Lock		
	<b>nukild</b>	ID of the Smart Lock	
	<b>name</b>	Name of the Smart Lock	
	<b>lastKnownState</b>	JSON list containing the last known lock state of the Smart Lock	
		<b>state</b>	ID of the lock state (see <a href="#">Lock states</a> )
		<b>stateName</b>	Name of the lock state (see <a href="#">Lock states</a> )
		<b>batteryCritical</b>	Flag indicating if the batteries of the Smart Lock are at critical level
<b>timestamp</b>		Timestamp of the retrieval of this lock state	
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.	
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/list?token=123456">http://192.168.1.50:8080/list?token=123456</a> <a href="http://192.168.1.50:8080/list?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/list?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>		
<b>Example-Response</b>	<pre>[{   "nukild": 1,   "name": "Home",   "lastKnownState": {     "state": 1,     "stateName": "locked",     "batteryCritical": false,     "timestamp": "2016-10-03T06:49:00+00:00" }   },{   "nukild": 2,</pre>		



	<pre> "name": "Grandma", "lastKnownState": {   "state": 3,   "stateName": "unlocked",   "batteryCritical": false,   "timestamp": "2016-10-03T06:49:00+00:00" } } </pre>
--	---

## /lockState

<b>URL</b>	<a href="http://192.168.1.50:8080/lockState">http://192.168.1.50:8080/lockState</a>	
<b>Usage</b>	Retrieves and returns the current lock state of a given Smart Lock	
<b>URL-Parameters</b>	<b>nukild</b>	The ID of the Smart Lock from which the lock state should be retrieved
	<b>token or hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON list containing the retrieved lock state	
	<b>state</b>	ID of the lock state (see <a href="#">Lock states</a> )
	<b>stateName</b>	Name of the lock state (see <a href="#">Lock states</a> )
	<b>batteryCritical</b>	Flag indicating if the batteries of the Smart Lock are at critical level
	<b>success</b>	Flag indicating if the lock state retrieval has been successful
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
	<b>HTTP 404</b>	Returned if the given Smart Lock is unknown
	<b>HTTP 503</b>	Returned if the given Smart Lock is offline
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/lockState?nukild=1&amp;token=123456">http://192.168.1.50:8080/lockState?nukild=1&amp;token=123456</a> <a href="http://192.168.1.50:8080/lockState?nukild=1&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762">http://192.168.1.50:8080/lockState?nukild=1&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762</a>	

	<a href="#">b806ad7d0d01cb6</a>
<b>Example-Response</b>	<pre>{   "state": 1,   "stateName": "locked",   "batteryCritical": false,   "success": true }</pre>

## /lockAction

<b>URL</b>	<a href="http://192.168.1.50:8080/lockAction">http://192.168.1.50:8080/lockAction</a>	
<b>Usage</b>	Performs a lock operation on the given Smart Lock	
<b>URL-Parameters</b>	<b>nukild</b>	The ID of the Smart Lock which should execute the lock action
	<b>action</b>	The desired lock action (see <a href="#">Lock states</a> )
	<b>nowait</b>	Flag (0 or 1) indicating whether or not to wait for the lock action to complete and return its result ( <i>optional; defaults to 0</i> )
	<b>token</b> or <b>hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON list containing the result of the lock action	
	<b>batteryCritical</b>	Flag indicating if the batteries of the Smart Lock are at critical level
	<b>success</b>	Flag indicating if the lock action has been executed successful
<b>Errors</b>	<b>HTTP 400</b>	Returned if the given <b>action</b> is invalid
	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
	<b>HTTP 404</b>	Returned if the given Smart Lock is unknown
	<b>HTTP 503</b>	Returned if the given Smart Lock is offline

<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/lockAction?nukild=1&amp;action=1&amp;token=123456">http://192.168.1.50:8080/lockAction?nukild=1&amp;action=1&amp;token=123456</a> <a href="http://192.168.1.50:8080/lockAction?nukild=1&amp;action=1&amp;ts=2019-03-05T01:06:53Z&amp;nr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/lockAction?nukild=1&amp;action=1&amp;ts=2019-03-05T01:06:53Z&amp;nr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>
<b>Example-Response</b>	{ "success": true, "batteryCritical": false }

## /unpair

not available on the software bridge

<b>URL</b>	<a href="http://192.168.1.50:8080/unpair">http://192.168.1.50:8080/unpair</a>	
<b>Usage</b>	Removes the pairing with a given Smart Lock	
<b>URL-Parameters</b>	<b>nukild</b>	The ID of the Smart Lock which should be unpaired
	<b>token</b> or <b>hash, nr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON list containing the result of the operation	
	<b>success</b>	Flag indicating if the lock action has been executed successful
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
	<b>HTTP 404</b>	Returned if the given Smart Lock is unknown
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/unpair?nukild=1&amp;token=123456">http://192.168.1.50:8080/unpair?nukild=1&amp;token=123456</a> <a href="http://192.168.1.50:8080/unpair?nukild=1&amp;ts=2019-03-05T01:06:53Z&amp;nr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/unpair?nukild=1&amp;ts=2019-03-05T01:06:53Z&amp;nr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	{ "success": true }	

/info

<b>URL</b>	<a href="http://192.168.1.50:8080/info">http://192.168.1.50:8080/info</a>		
<b>Usage</b>	Returns all Smart Locks in range and some device information of the bridge itself		
<b>URL-Parameters</b>	<b>token</b> or <b>hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.	
<b>Response</b>	JSON list with the result		
	<b>bridgeType</b>	<ul style="list-style-type: none"> <li>• 1 =&gt; Hardware bridge</li> <li>• 2 =&gt; Software bridge</li> </ul>	
	<b>ids</b>	JSON list containing the ids of the bridge	
		<b>hardwareId</b>	Hardware ID ( <i>hardware bridge only</i> )
		<b>serverId</b>	Server ID
	<b>versions</b>	JSON list containing the versions of bridge	
		<b>firmwareVersion</b>	Version of the bridges firmware ( <i>hardware bridge only</i> )
		<b>wifiFirmwareVersion</b>	Version of the WiFi modules firmware ( <i>hardware bridge only</i> )
		<b>appVersion</b>	Version of the bridge appsoftware bridge only
	<b>uptime</b>	Uptime of the bridge in seconds	
	<b>currentTime</b>	Current timestamp	
	<b>serverConnected</b>	Flag indicating whether or not the bridge is connected to the Nuki server	
<b>scanResults</b>	JSON Array. One item of the following per Smart Lock		

		<b>nukild</b>	Smart Lock ID
		<b>name</b>	BLE-Name of the Smart Lock
		<b>rsssi</b>	RSSI value
		<b>paired</b>	Flag indicating whether or not a pairing with this Smart Lock has already been established
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.	
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/info?token=123456">http://192.168.1.50:8080/info?token=123456</a> <a href="http://192.168.1.50:8080/info?ts=2019-03-05T01:06:53Z&amp;nr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/info?ts=2019-03-05T01:06:53Z&amp;nr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>		
<b>Example-Response</b>	<pre>{   "bridgeType": 1,   "ids": {"hardwareId": 12345678, "serverId": 12345678},   "versions": {"firmwareVersion": "0.1.0", "wifiFirmwareVersion": "0.2.0"},   "uptime": 120,   "currentTime": "2016-04-01T12:10:11Z",   "serverConnected": true,   "scanResults": [ { "nukild": 10, "name": "Nuki_00000010", "rsssi": -87, "paired": true }, { "nukild": 11, "name": "Nuki_00000011", "rsssi": -93, "paired": false } ] }</pre>		

## /callback

The following endpoints provides methods to register up to 3 http (no https) url callbacks, which will be triggered once the lock state of one of the known Smart Locks changes.

The new lock state will be sent to the callback url by executing a POST request and posting a JSON list in the following format:

```
{"nukild": 11, "state": 1, "stateName": "locked", "batteryCritical": false}
```

/callback/add

<b>URL</b>	<a href="http://192.168.1.50:8080/callback/add">http://192.168.1.50:8080/callback/add</a>	
<b>Usage</b>	Registers a new callback url	
<b>URL-Parameters</b>	<b>url</b>	The callback url to be added (no https, url encoded, max. 254 chars)
	<b>token or hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON list containing the result	
	<b>success</b>	Flag indicating if the url has been added successfully
	<b>message</b>	Contains the reason for the failure if <b>success</b> is false
<b>Errors</b>	<b>HTTP 400</b>	Returned if the given <b>URL</b> is invalid or too long
	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/callback/add?url=http%3A%2F%2F192.168.0.20%3A8000%2Fnuki&amp;token=123456">http://192.168.1.50:8080/callback/add?url=http%3A%2F%2F192.168.0.20%3A8000%2Fnuki&amp;token=123456</a> <a href="http://192.168.1.50:8080/callback/add?url=http%3A%2F%2F192.168.0.20%3A8000%2Fnuki&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/callback/add?url=http%3A%2F%2F192.168.0.20%3A8000%2Fnuki&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	<pre>{   "success": true }</pre>	

/callback/list

<b>URL</b>	<a href="http://192.168.1.50:8080/callback/list">http://192.168.1.50:8080/callback/list</a>	
<b>Usage</b>	Returns all registered url callbacks	
<b>URL-Parameters</b>	<b>token or hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.

<b>Response</b>	JSON list with the result		
	<b>callbacks</b>	JSON array. One item of the following per callback	
		<b>id</b>	ID of the callback
		<b>url</b>	URL of the callback
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.	
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/callback/list?token=123456">http://192.168.1.50:8080/callback/list?token=123456</a> <a href="http://192.168.1.50:8080/callback/list?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/callback/list?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>		
<b>Example-Response</b>	<pre>{   "callbacks": [     {       "id": 0,       "url": "http://192.168.0.20:8000/nuki"     },     {       "id": 1,       "url": "http://192.168.0.21/test"     }   ] }</pre>		

/callback/remove

<b>URL</b>	<a href="http://192.168.1.50:8080/callback/remove">http://192.168.1.50:8080/callback/remove</a>	
<b>Usage</b>	Removes a previously added callback	
<b>URL-Parameters</b>	<b>id</b>	The id of the callback to be removed
	<b>token or hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON list containing the result	
	<b>success</b>	Flag indicating if the url has been added successfully
	<b>message</b>	Contains the reason for the failure if <b>success</b> is false

<b>Errors</b>	<b>HTTP 400</b>	Returned if the given <b>url</b> is invalid or too long
	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/callback/remove?id=0&amp;token=123456">http://192.168.1.50:8080/callback/remove?id=0&amp;token=123456</a> <a href="http://192.168.1.50:8080/callback/remove?id=0&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/callback/remove?id=0&amp;ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	<pre>{   "success": true }</pre>	

## 6. Maintenance endpoints

The following endpoints are available for maintenance purposes of the hardware bridge. Therefore they are not available on the software bridge.

/log

<b>URL</b>	<a href="http://192.168.1.50:8080/log">http://192.168.1.50:8080/log</a>	
<b>Usage</b>	Retrieves the log of the bridge	
<b>URL-Parameters</b>	<b>offset</b>	Offset position where to start retrieving log entries ( <i>optional; defaults to 0</i> )
	<b>count</b>	How many log entries to retrieve ( <i>optional; defaults to 100</i> )
	<b>token or hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	JSON array. One item of the following per log entry	
	<b>timestamp</b>	Timestamp of the log entry
	<b>type</b>	Type of the log entry
	some more optional parameters	



<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/log?token=123456">http://192.168.1.50:8080/log?token=123456</a> <a href="http://192.168.1.50:8080/log?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/log?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	<pre>[   {"timestamp": "2016-10-06T16:46:05+00:00", "type": "..."},   {"timestamp": "2016-10-06T16:46:05+00:00", "type": "..."}, ... ]</pre>	

## /clearlog

<b>URL</b>	<a href="http://192.168.1.50:8080/clearlog">http://192.168.1.50:8080/clearlog</a>	
<b>Usage</b>	Clears the log of the Bridge	
<b>URL-Parameters</b>	<b>token</b> or <b>hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	No response	
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/clearlog?token=123456">http://192.168.1.50:8080/clearlog?token=123456</a> <a href="http://192.168.1.50:8080/clearlog?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/clearlog?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	None	

## /fwupdate

<b>URL</b>	<a href="http://192.168.1.50:8080/fwupdate">http://192.168.1.50:8080/fwupdate</a>
<b>Usage</b>	Immediately checks for a new firmware update and installs it

<b>URL-Parameters</b>	<b>token</b> or <b>hash</b> , <b>rnr</b> , <b>ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	No response	
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/fwupdate?token=123456">http://192.168.1.50:8080/fwupdate?token=123456</a> <a href="http://192.168.1.50:8080/fwupdate?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/fwupdate?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	None	

## /reboot

<b>URL</b>	<a href="http://192.168.1.50:8080/reboot">http://192.168.1.50:8080/reboot</a>	
<b>Usage</b>	Reboots the bridge	
<b>URL-Parameters</b>	<b>token</b> or <b>hash</b> , <b>rnr</b> , <b>ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	No response	
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/reboot?token=123456">http://192.168.1.50:8080/reboot?token=123456</a> <a href="http://192.168.1.50:8080/reboot?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/reboot?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	None	

## /factoryReset

<b>URL</b>	<a href="http://192.168.1.50:8080/factoryReset">http://192.168.1.50:8080/factoryReset</a>
------------	---

<b>Usage</b>	Performs a factory reset	
<b>URL-Parameters</b>	<b>token</b> or <b>hash, rnr, ts</b>	The <a href="#">api token</a> configured via the Nuki app when enabling the API as plain token or hashed token.
<b>Response</b>	No response	
<b>Errors</b>	<b>HTTP 401</b>	Returned if the given <b>token</b> is invalid or a <b>hashed token</b> parameter is missing.
<b>Example-Calls</b>	<a href="http://192.168.1.50:8080/factoryReset?token=123456">http://192.168.1.50:8080/factoryReset?token=123456</a> <a href="http://192.168.1.50:8080/factoryReset?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6">http://192.168.1.50:8080/factoryReset?ts=2019-03-05T01:06:53Z&amp;rnr=4711&amp;hash=f52eb5ce382e356c4239f8fb4d0a87402bb95b7b3124f0762b806ad7d0d01cb6</a>	
<b>Example-Response</b>	None	

## 7. Changelog

### Changelog v 1.8

07.03.2019

- Introducing the hashed [token](#) as a more secure alternative to send the plain token

### Changelog v 1.7

30.03.2018

- Small changes in bridge discovery information

### Changelog v 1.6

21.06.2017

- Added Bridge discovery