Swisscom LPN Portal

Application Development Guide

1

LPN Portal to AS tunnel interface

February 2017

NOTICE

Versions

Version	Date	Author	Details
1	31/08/16	Swisscom	Initial version
1.1	30/10/16	Swisscom	Revised initial version
1.2	14/02/17	Swisscom	New Layout

Reference documents

Item	Documents	Author
01	Swisscom LPN Portal Device Developer User Guide	Swisscom

Table of Contents

Definitions and acronyms

ABP	Activation-By-Personalization
ACK	Acknowledgement of an alarm
ADR	Adaptive Data Rate
AS	Application Server
BPM	Business Process Management
BSS	Billing Support Systems
CSP	Communication Service Provider
End Device	A sensor or actuator
ETSI	European Telecommunications Standards Institute
ESP	Estimated Signal Power
FCtrl	Frame Control
HAN	Home Area Network
HSM	Hardware Security Module
HTTPS	Hypertext Transfer Protocol Secure
IEC	International Electrotechnical Commission
IoT	Internet of Things
ISM	Industrial Scientific Medical
JSON	JavaScript Object Notation
LAT	Latitude
LC	Logical Channel
LON	Longitude
LoRaWAN™	Long Range Wide Area Network
LPWAN	Low Power Wide Area Network
LRC	Long Range Controller
LRR	Long Range Relay
M2M	Machine-to-Machine
MAC	Media Access Control
MIC	Message Integrity Code
NW	Network
OSS	Operations Support Systems
OTAA	Over-The-Air-Activation
PER	Packet Error Rate
REST	Representational State Transfer
RF	Radio Frequency
RIT	Receiver Initiated Transmit
RSSI	Received Signal Strength Indicator
Rx	Receiver
RX1	First receive window
RX2	Second receive window

SaaS	Software As A Service
SF	Spreading Factor
SMP	System Management Platform
SNR	Signal to Noise Ratio
SSO	Single Sign On
Tx	Transmitter
TWA	ThingPark Wireless Application
UNB	Ultra Narrow Band
URL	Uniform Resource Locator
XML	Extensible Markup Language

1. Scope

The scope of this document is to give development guidelines on the tunnel mode interface to applications developers.

2. Swisscom LPN Portal wireless tunnel mode interface

The Swisscom LPN Portal provides an interface for developers of applications combined with wireless sensors or actuators compatible with the LoRaWAN specification:

- Tunnel mode interface: A simple message passing interface between the Swisscom LPN PORTAL servers which implement the network MAC layer (LRC servers), and application servers. This interface forwards the uplink radio packets raw payload data and associated metadata (RSSI, SNR…) to one or more application servers associated to the network node MAC address. As the Swisscom LPN Portal supports bidirectional communications, application servers may also send requests to one of the LRC nodes to send downlink frames to a network node identified by its full format (64bits) MAC address

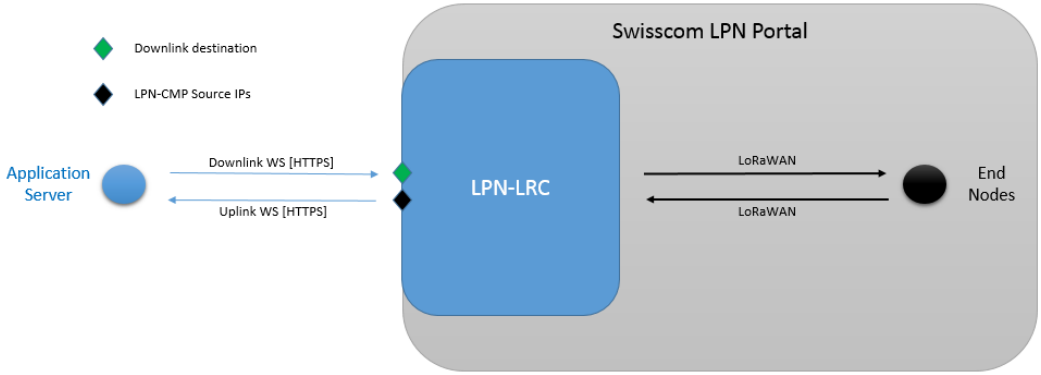
This document provides information on:

- Low level LRC configuration parameters required to associate one or more application servers with a given network node MAC address.
- Format of LRC to application server messages that encapsulate uplink payload data and associated metadata
- The format of application server to LRC messages that encapsulate downlink payload.

3. Connectivity

The tunneling interface is based on HTTPS for uplink & downlink packets flows.

The confidentiality of the uplink/downlink are managed by an HTTPS connection. The uplink HTTPS session is mounted between the LRC cluster and the Application Server. The downlink HTTPS session is mounted between the Application Server and the Reverse HTTP proxy in front of the LRC cluster.



The uplink and downlink packets may be secured by an authentication layer. The LRC will share an AS key with the Application Server.

The AS key will be used by the Application Server to generate a signature added to downlink packets. This signature will be used by the LRC to verify the identity and the authorization of the Application Server. If the identity or the authorization cannot be successfully verified, the packet will be dropped.

The AS key will be also used by the LRC to generate a signature added to uplink packets. The signature will be used by the AS to verify the identity of the LRC. If the identity cannot be successfully verified, the packet should be dropped by the AS.

- The downlink destination is as follows:

  https://proxy1.lpn.swisscom.ch/thingpark/lrc/rest/downlink/?DevEUI=<DEVEUI>&FPort=<FPORT>&Payload=<PAYLOAD>

- The LPN Portal source IPs for uplink messages which should be approved by your ASs firewall are the following:
  - 195.65.47.212
  - 195.65.47.242
  - 194.209.209.249

4. LRC frame tunneling

4.1. Parameters format

4.1.1 ISO 8601 timestamps

All ISO 8601 timestamps described in this section use the following convention:

`YYYY-MM-DDThh:mm:ss.s+|-hh:mm (e.g. 2016-08-01T09:06:06.0+02:00)`

Where:

- YYYY = four-digit year
- MM = two-digit month
- DD = two-digit day
- hh = two digits of hour (00 through 23)
- mm = two digits of minute (00 through 59)
- ss = two digits of second (00 through 59)
- s = one to three digits of millisecond (0 through 999)
- +|-hh:mm = timezone designator (+hh:mm or –hh:mm)

All timestamp parameters described above are mandatory.

4.1.2 Timestamp Encoding When Uplink/Downlink Security Is Activated

When the uplink/downlink security of the Application Server is activated, the timestamp parameter has to be encoded differently in the SHA256 and in the URL of the HTTP request:

Special Characters	In SHA256, use…	In URL, use…
+	+	%2B
-	–	-
:	:	%3A
.	.	.
Examples	2016-08-01T09:06:06.0+02:00 2016-11-28T09:06:06.0-04:00	2016-08-01T09%3A06%3A06.0%2B02%3A00 2016-11-28T09%3A06%3A06.0-04%3A00

4.2. Tunnel mode: LRC to application server interface

4.2.1 Uplink frame

Uplink destination URLs are defined for each device in the AS profile. The main criteria is the LoRa port numbers expressed as intervals, lists or single values. A default LoRa port destination can be declared.

There are two modes of distribution which are not exclusive:

- Blast/blind mode: packets are delivered to a set of destinations without acknowledgment control.
- Order mode: packets are delivered to a list of destinations until one of them confirms receipt (200OK).

Duplicate packets (same counter up and same payload) are not sent to application servers, as long as the multiple copies are received by the network infrastructure within a maximum delay of 250ms (configurable by the LPWA operator). The XML payload sent to the application servers still include the RF metadata corresponding to all receiving base stations.

The uplink frame is transmitted in a HTTP/POST request with query parameters and an XML payload.

Following query parameters are defined for uplink frames (lexicographic order):

AS_ID	Application Server ID (only reported when the authentication is activated) NOTE: AS_ID is not reported when the Uplink/downlink authentication is not activated in the AS profile.
LrnDevEui	Device DevEUI.
LrnFPort	LoRaWAN port number.
LrnInfos	Service profile name used to route the packet.
Time	ISO 8601 timestamp associated to generation of the HTTP request by the LRC (only reported when the authentication is activated) NOTE: Time is not reported when the Uplink/downlink authentication is not activated in the AS profile.
Token	Security token generated by the LRC (only reported when the authentication is activated) NOTE: Token is not reported when the Uplink/downlink authentication is not activated in the AS profile.

Following XML elements are defined for uplink frames (lexicographic order):

ACKbit	ACKBit set by the device. NOTE: ACKbit is not filled in the XML document if not set in the uplink frame.
ADRbit	ADRBit set by the device. NOTE: ADRbit is not filled in the XML document if not set in the uplink frame.
AppSKey	Encrypted Appskey with the ASkey.
Channel	LC used by the device.
CustomerData	ASCII customer data set by provisioning.

CustomerID	Customer ID associated to the Device Manager account.
DevAddr	Device DevAddr.
DevEUI	Device DevEUI.
DevLrrCnt	Number of LRRs which received this packet.
FCntDn	The last downlink counter to the device.
FCntUp	The uplink counter for this packet.
FPort	LoRaWAN FPort used by the device for this packet.
InstantPER	Instant PER (Packet Error Rate). NOTE: The instance PER is computed on the last 10 packets.
Late	Indicate if the packet was queued by the LRR. NOTE: Late is always filled. 0 means that the packet was not queued by the LRR, 1 means that the packet was queued (the LRR queues packets when the connection between the LRR and the LRC is temporarily broken).
Lrcid	ID of the LRC that processed the packet.
Lrrid	The ID of the LRR that received the packet with the best ESP. This LRR is flagged as "best LRR".
LrrLAT LrrLON	LAT and LON of the best LRR.
LrrRSSI	RSSI measured by the best LRR.
LrrSNR	SNR measured by the best LRR.
Lrrs/Lrr/Lrrid	LRR ID associated to this <Lrr> XML element.
Lrrs/Lrr/LrrESP	ESP measured by the LRR associated to this <Lrr> XML element.
Lrrs/Lrr/LrrRSSI	RSSI measured by the LRR associated to this <Lrr> XML element.
Lrrs/Lrr/LrrSNR	SNR measured by the LRR associated to this <Lrr> XML element.
MeanPER	Mean PER (Packet Error Rate). NOTE: The Mean PER is the average of the instantaneous PER of the last 20 packets.
mic_hex	MIC in hexadecimal ASCII format.
ModelCfg	ASCII ThingPark Cloud data set by provisioning.
MType	LoRaWAN MType of the packet.
payload_hex	LoRaWAN payload in hexadecimal ASCII format.
SpFact	SF used by the device.
SubBand	SUB-BAND used by the device.
Time	LRR Timestamp for the packet.

NOTE: <Lrr> XML elements is reported for max 3 LRRs which received the packet. If the packet was received by more than 3 LRRs, only the 3 LRRs with the best ESP are reported.

4.2.2 Sample of uplink frame HTTP request

In this sample, *<as-url>* is the destination URL configured in the AS profile:

NOTE: In an URL, the "+" and ":" characters must be escaped.

```
>> POST <as-
url>?LrnDevEui=000000000F1D8693&LrnFPort=2&LrnInfos=UPHTTP_LAB_LORA&AS_ID=app1.s
ample.com&Time=2016-01-
11T14%3A11%3A11.333%2B02%3A00&Token=fd0b0b00464aa798a59282d64eaa70813e33bff87682
880db49638569d096aad

<?xml version="1.0" encoding="UTF-8"?>
<DevEUI_uplink xmlns="http://uri.actility.com/lora">
      <Time>2016-10-26T15:09:59.680+02:00</Time>
      <DevEUI>FF000012FF000012</DevEUI>
      <FPort>1</FPort>
      <FCntUp>29</FCntUp>
      <ADRbit>1</ADRbit>
      <MType>2</MType>
      <FCntDn>5</FCntDn>
      <payload hex>000009f4</payload hex>
      <mic hex>1ab90d3d</mic hex>
      <Lrcid>00000401</Lrcid>
      <LrrRSSI>-114.000000</LrrRSSI>
      <LrrSNR>1.500000</LrrSNR>
      <SpFact>7</SpFact>
      <SubBand>G1</SubBand>
      <Channel>LC3</Channel>
      <DevLrrCnt>3</DevLrrCnt>
      <Lrrid>29000128</Lrrid>
      <Late>0</Late>
      <LrrLAT>47.379826</LrrLAT>
      <LrrLON>8.527115</LrrLON>
      <Lrrs>
                <Lrr>
                          <Lrrid>29000128</Lrrid>
                          <Chain>0</Chain>
                          <LrrRSSI>-114.000000</LrrRSSI>
                          <LrrSNR>1.500000</LrrSNR>
                          <LrrESP>-116.324738</LrrESP>
                </Lrr>
                <Lrr>
                          <Lrrid>29000150</Lrrid>
                          <Chain>0</Chain>
                          <LrrRSSI>-118.000000</LrrRSSI>
                          <LrrSNR>0.000000</LrrSNR>
                          <LrrESP>-121.010300</LrrESP>
                </Lrr>
                <Lrr>
                          <Lrrid>29000107</Lrrid>
                          <Chain>0</Chain>
                          <LrrRSSI>-121.000000</LrrRSSI>
                          <LrrSNR>-7.250000</LrrSNR>
                          <LrrESP>-128.999496</LrrESP>
                </Lrr>
      </Lrrs>
      <CustomerID>100000304</CustomerID>
      <CustomerData>{"alr":{"pro":"LORA/Generic","ver":"1"}}</CustomerData>
      <ModelCfg>0</ModelCfg>
      <AppSKey>a13a0a32570e67839a82ce3c06cc4b4e</AppSKey>
      <InstantPER>0.166667</InstantPER>
      <MeanPER>0.047665</MeanPER>
      <DevAddr>0981DACF</DevAddr>
</DevEUI_uplink>
```


The payload may be provided to the application server either encrypted or decrypted. The following rules apply:

ABP device	Payload is provided decrypted to the AS, if the AppSKey has been

	provisioned for the relevant FPort. Otherwise the Payload is provided encrypted to the AS.
OTAA device Embedded security server	Payload is provided decrypted to the AS
OTAA device HSM	The payload and AppSKey are provided encrypted to the AS.

4.2.3 Downlink frame sent

"Downlink frame sent" message destination URLs are defined for each device in the AS profile.

The main propose of the "downlink frame sent" is to report to the application server, the over the air delivery status of a downlink frame initiated by an application server. The delivery status of downlink frames initiated by the LRC itself are not reported to the application server.

Following query parameters are defined for "downlink frame sent" messages (lexicographic order):

AS_ID	Application Server ID (only reported when the authentication is activated) NOTE: AS_ID is not reported when the Uplink/downlink authentication is not activated in the AS profile.
LrnDevEui	Device DevEUI.
LrnFPort	LoRaWAN port number.
LrnInfos	Service profile name used to route the packet.
Time	ISO 8601 timestamp associated to generation of the HTTP request by the LRC (only reported when the authentication is activated) NOTE: Time is not reported when the Uplink/downlink authentication is not activated in the AS profile.
Token	Security token generated by the LRC (only reported when the authentication is activated) NOTE: Token is not reported when the Uplink/downlink authentication is not activated in the AS profile.

Following XML elements are defined for "downlink frame sent" messages (lexicographic order):

Channel	LC used by the device.
CustomerData	ASCII customer data set by provisioning.
CustomerID	Customer ID associated to the Device Manager account.
DeliveryStatus	The over the air delivery status: > 0: Downlink frame was sent over the air (either on RX1 or RX2). This means that the downlink frame was transmitted, over the air, by the LRR. But the downlink frame may not have been received by the device.

	> 1: Downlink frame was not sent over the air (neither on RX1 nor RX2). The downlink frame is not retransmitted by the network server. Accordingly the downlink frame will have to be reinitiated by the application server.
DeliveryFailedCause1	The over the air delivery error cause for RX1. Class A device: Transmission slot busy on RX1: > A0: "Radio stopped" > A1: "Downlink radio stopped" > A3: "Radio busy" > A4: "Listen before talk" Class A device: Received too late for RX1: > B0: "Too late for RX1" Class A device: LRC selects RX2: > C0: "LRC chooses RX2" Class A device: DC constraint on RX1: > D0: "Duty cycle constraint detected by LRR" > DA: "Duty cycle constraint detected by LRC" NOTE: DeliveryFailedCause1 is set to 00 (or is not filled) when no error occurs on RX1.
DeliveryFailedCause2	The over the air delivery error cause for RX2. Class A device: Transmission slot busy on RX2: > A0: "Radio stopped" > A1: "Downlink radio stopped" > A3: "Radio busy" > A4: "Listen before talk" Class A device: Received too late for RX2: > B0: "Too late for RX2" Class A device: DC constraint on RX2: > D0: "Duty cycle constraint detected by LRR" > DA: "Duty cycle constraint detected by LRC" Class C device: DC constraint on RX2: > E0: "Max delay for Class C" (60 seconds) NOTE: DeliveryFailedCause2 is set to 00 (or is not filled) when no error occurs on RX2.
DevEUI	Device DevEUI.
FCntDn	The downlink counter for this packet.
FCntUp	The last uplink counter from the device.
FPort	LoRaWAN FPort used by the device for this packet.
Lrcid	ID of the LRC that processed the packet.
Lrrid	ID of the LRR used to send the packet.
SpFact	RX1 SF as asked by the LRC.
SubBand	Sub-band used by the device.
Time	LRR Timestamp for the packet.

4.2.4 Sample of "downlink frame sent" HTTP request

In this sample, *<as-url>* is the destination URL configured in the AS profile:

```
>> POST <as-
url>?LrnDevEui=000000000F1D8693&LrnFPort=2&LrnInfos=UPHTTP_LAB_LORA&AS_ID=app1.s
ample.com&Time=2016-01-
11T14%3A22%3A22.333%2B02%3A00&Token=c85e44b8c386053962fd22be4b9728d770b4c767952f
1a6a741120be300776e7

<?xml version="1.0" encoding="UTF-8"?>
<DevEUI_downlink_Sent xmlns="http://uri.actility.com/lora">
   <Time>2015-01-27T10:00:46.336+01:00</Time>
   <DevEUI>000000000F1D8693</DevEUI>
   <FPort>2</FPort>
   <FCntUp>7022</FCntUp>
   <FCntDn>22</FCntDn>
   <Lrcid>00000000</Lrcid>
   <SpFact>7</SpFact>
   <SubBand>G1</SubBand>
   <Channel>LC3</Channel>
   <Lrrid>0000000a</Lrrid>
   <DeliveryStatus>1</DeliveryStatus>
   <DeliveryFailedCause1>A3</DeliveryFailedCause1>
   <DeliveryFailedCause2>00</DeliveryFailedCause2>
   <CustomerID>100000507</CustomerID>
   <CustomerData>…</CustomerData>
</DevEUI_downlink_Sent>
```

4.2.5 LRC authentication for uplink frame and downlink frame sent

Securing LRC to AS frame is implemented with the following principles:

- The LRC adds the AS ID and the generation time stamp in the message.
- Then, the LRC adds a security token to sign the message based on a pre-shared AS key.
- When the AS receives a message, the AS will re-compute the security token.
- If the re-computed security token matches the security token provided by the LRC, and if the time deviation (between the generation by the LRC and the reception by the AS) is acceptable (e.g. less than 10 seconds), the AS can trust the message and process it accordingly.

The AS ID / AS Key are part of the AS profile configuration associated to the device. The generation of the security token by the LRC can be deactivated by not setting an AS_ID and AS key in the AS Profile.

Token must be verified as following by the Application Server:

- The application server retrieves the <query-parameters> WITHOUT the Token QP (Query parameters include the AS_ID and the Time):

  For an uplink frame (based on the example provided section 4.2.2):

  ```
  e.g. <query-parameters> :=
  LrnDevEui=000000000F1D8693&LrnFPort=2&LrnInfos=UPHTTP_LAB_LORA&A
  S_ID=app1.sample.com&Time=2016-01-11T14:11:11.333+02:00
  ```

  For a downlink frame sent (based on the example provided section 4.2.4):

e.g. <query-parameters> :=
LrnDevEui=000000000F1D8693&LrnFPort=2&LrnInfos=UPHTTP_LAB_LORA&AS_ID=app1.sample.com&Time=2016-01-11T14:22:22.333+02:00

> The application server builds the <body-elements> as the concatenation, without separator, of the following values:

For an uplink frame (extract from the <DevEUI_uplink> body): CustomerID, DevEUI, FPort, FCntUp, payload_hex.
e.g. <body-elements> := 1000005070000000000F1D8693270110027bd00

For a downlink frame sent (extracted from the <DevEUI_downlink> body): CustomerID, DevEUI, FPort, FCntDown.
e.g. <body-elements> := 1000005070000000000F1D8693222

> The application server re-computes the <token> as:
SHA-256(<body-elements><query-parameters><AsKey>):

For an uplink frame:
e.g. <token> :=
SHA-256(1000005070000000000F1D8693270110027bd00LrnDevEui=000000000F1D8693&LrnFPort=2&LrnInfos=UPHTTP_LAB_LORA&AS_ID=app1.sample.com&Time=2016-01-11T14:11:11.333+02:0046ab678cd45df4a4e4b375Eacd096acc)

For a downlink frame sent
e.g. <token> :=
SHA-256(1000005070000000000F1D8693222LrnDevEui=000000000F1D8693&LrnFPort=2&LrnInfos=UPHTTP_LAB_LORA&AS_ID=app1.sample.com&Time=2016-01-11T14:22:22.333+02:0046ab678cd45df4a4e4b375Eacd096acc)

Where 46ab678cd45df4a4e4b375Eacd096acc is the 128 bits pre-shared key (lower case hex string representation) between the Application Server and the LRC as defined in the AS profile.

> The <token> is encoded as a hex string AND can be compared to the <token> provided by the LRC in the <query parameters> line.

For an uplink frame:
e.g. <encrypted-token> :=
fd0b0b00464aa798a59282d64eaa70813e33bff87682880db49638569d096aad

For a downlink frame sent:
e.g. <encrypted-token> :=
c85e44b8c386053962fd22be4b9728d770b4c767952f1a6a741120be300776e7

> Finally, if the token is valid, the application server can verify the deviation between the emission (as provided in the Time query parameter) and the reception by the application server.

4.2.6 XML or JSON encoding

Uplink frame and "downlink frame sent" HTTP request body may be encoded by the LRC has an XML payload or as a JSON payload. The default is set to XML.

This can be configured in the Application server associated to the device in the Device Manager.

Information elements in the XML document (as defined in *4.2.1 Uplink frame)* can be mapped one-to-one with information elements in the JSON document.

A 1-to-1 mapping must be assumed between information elements present in the XML document (as defined section 4.2.1 for uplink frame and section 4.2.3 for downlink frame sent) and information elements present in the JSON document.

4.2.7 Sample of uplink JSON payload


```
{
      "DevEUI_uplink": {
                "Time": "2016-10-26T14:52:00.331+02:00",
                "DevEUI": "FF000012FF000012","FPort": "1",
                "FCntUp": "20",
                "ADRbit": "1",
                "MType": "2",
                "FCntDn": "2",
                "payload_hex": "000009f4",
                "mic_hex": "987bda77",
                "Lrcid": "00000401",
                "LrrRSSI": "-114.000000",
                "LrrSNR": "-2.000000",
                "SpFact": "7",
                "SubBand": "G1",
                "Channel": "LC1",
                "DevLrrCnt": "2",
                "Lrrid": "29000150",
                "Late":"0",
                "LrrLAT": "47.374199",
                "LrrLON": "8.537522",
                "Lrrs": {
                          "Lrr": [
                             {
                                    "Lrrid": "29000150",
                                    "Chain": "0",
                                    "LrrRSSI": "-114.000000",
                                    "LrrSNR": "-2.000000",
                                    "LrrESP": "-118.124428"
                             },
                             {
                                    "Lrrid": "29000107",
                                    "Chain": "0",
                                    "LrrRSSI": "-119.000000",
                                    "LrrSNR": "-8.250000",
                                    "LrrESP": "-127.855560"
                             }
                            ]
                          },
                          "CustomerID": "100000304",
                          "CustomerData":
{"alr":{"pro":"LORA/Generic","ver":"1"}},
                          "ModelCfg": "0",
                          "AppSKey": "a13a0a32570e67839a82ce3c06cc4b4e",
                          "InstantPER": "0.090909",
                          "MeanPER": "0.004545",
                          "DevAddr": "0981DACF"
  }
}
```

4.3. Tunnel mode: Application server to LRC interface

4.3.1 Downlink frame

Depending on the device provisioning encryption/decryption can be performed by the LRC. The following rules apply:

NOTE: When the payload is encrypted, Wireless Logger cannot decrypt it as this application does not embed the associated decryption key.

ABP device	Payload can be provided not encrypted by the AS and will be encrypted by the LRC, if AppSKey has been provisioned for the relevant FPort and if the FCntDn parameter is absent. Otherwise the Payload must be provided encrypted by the AS and the FCntDn parameter must be present.
OTAA device Embedded security server	Payload must be provided not encrypted by the AS and will be encrypted by the LRC. The FCntDn parameter must be absent.
OTAA device HSM	The payload must be provided encrypted by the AS. The FCntDn parameter must be present.

The following HTTP/POST message format is used to tunnel the radio frame payload and associated metadata from the target application server to the LRC. The application server acts as a HTTP client and the reverse HTTP proxy (PROXY_HTTP server) acts as a HTTP server. Rerouting of the HTTP request to the primary LRC or the backup LRC is handled by the reverse HTTP proxy.

The LoRaWAN™ MAC message integrity code (MIC) is always computed by the LRC, as part of the MAC frame formatting. The MAC payload may be encrypted either by the application or by the LRC (see table above).

Such POST command may be generated easily by tools such as curl or POSTman.

```
curl -H "Content-type:application/x-www-form-urlencoded" -X POST
"https://proxy1.lpn.swisscom.ch/thingpark/lrc/rest/downlink?DevEUI=000000000F1D8
693&FPort=1&Payload=0102030405060708090A&FCntDn=1234"
```

Following query parameters are defined for downlink frame (lexicographic order):

DevEUI (Mandatory)	Target device IEEE EUI64 in hexadecimal form (representing 8 octets)
FPort (Mandatory)	Target port (in decimal format)
Payload (Mandatory)	Hexadecimal payload. The hexadecimal payload will be encrypted by the LRC cluster if FCntDn parameter is absent, and if the LRC has been configured with an AppSKey for the specified LoRaWAN port, otherwise the Payload must be encrypted by the Application Server according to the LoRaWAN specification and the FCntDn

	parameter must be present. The Application Server encryption uses the downlink counter, which is why the FCntDn query parameter is required in this case.
FCntDn (Optional)	LoRaWAN Downlink Counter value used to encrypt the payload. This query parameter is needed only if the Application server (not the LRC) encrypts the payload. If present, FCntDn will be copied in the LoRaWAN header field FCnt, and the encrypted payload will be copied as-is to the LoRaWAN downlink frame by the LRC.
Confirmed (Optional)	A value of Confirmed=0 requests transmission of an UNCONFIRMED downlink frame. A value of Confirmed=1 requests transmission of a CONFIRMED downlink frame. Default value Confirmed=0 (UNCONFIRMED).
AS_ID (Optional)	Application Server ID, as provisioned in the AS Profile. The Application server ID is mandatory if the Application server authentication has been activated in the AS Profile. In this case the LRC will check that the Application Server is authorized to send downlink command to the device.
Time (Optional)	ISO 8601 time of the request. The Time is mandatory when the Application server authentication has been activated in the AS Profile. In this case the LRC will verify the time deviation between the generation and the reception of the request. The deviation must be lower than "Max Time Deviation" as defined in the AS Profile. NOTE: In the URL of the HTTP request, use "%2B" ASCII code for the "+" character and the "%3A" ASCII code for the ":" character.
Token (Optional)	Security token to sign the downlink frame. The Token is mandatory when the Application server authentication has been activated in the AS Profile.

Sample of downlink frame HTTP request:

```
>> POST
https://proxy1.lpn.swisscom.ch/thingpark/lrc/rest/downlink?DevEUI=000000000F1D86
93&FPort=1&Payload=00&AS_ID=app1.sample.com&Time=2016-01-
11T14%3A28%3A00.333%2B02%3A00&Token=ea8f31d2299cbece8e180a3012766c4df15fe3cf2e14
2d9fdf4035b5894ec886
```

Confirmed frames are declared as a message in the LPN Portal, therefore a confirmation is subject to deduction of your up-/downlink budget.

LRC HTTP response codes:

- 200 "Request queued by LRC": request accepted and queued until the class A device opens Rx slots by sending an uplink. In the case of a class C device, the downlink command will be sent as soon as the LRR base station radio is available and the maximum regulatory Tx duty cycle allows transmission.
- 350 "Invalid DevEUI": An Invalid DevEUI was entered.
- 350 "Downlink counter value already used. Expected=1238": the downlink counter value was already used, for instance due to a race condition with another Application server.
- 350 "Downlink counter value increment too large. Expected=1001": the AS supplied downlink counter value is much larger than the expected downlink counter value and was rejected by the LRC.

- 350 "Confirmed downlink is not authorized for this device" : the request for transmission of a confirmed downlink packet was rejected by the LRC due to absence of "ackedDownlinkFrame" feature flag in the Connectivity plan associated to the device.
- 350 "Invalid LoRa port 0": sending on port 0 (port reserved for LoRaWan MAC commands) is unauthorized from the tunneling interface.
- 350 "Security Check. AS_ID is mandatory": speaking to this device needs AS_ID. The Application Server authorization has been activated for this device and the application must be identified.
- 350 "Security Check. missing timestamp/token": Time and Token query parameter are mandatory when application server authentication is activated.
- 350 "Security Check. bad AS_ID" : AS_ID is not declared in the database or is not authorized for the targeted device.
- 350 "Security Check. Server Decrypt Error": Missing or badly formatted security token.
- 350 "Security Check. malformed ISO8601 time": An ISO 8601 date/time must be used (YYYY-MM-DDThh:mm:ss.s+|-hh:mm) representing a local time with a time zone offset in hours and minutes.
- 350 "Security Check. Invalid downlink frame timestamp": the time deviation between the frame generation by the application server and the reception by the LRC exceeds the MAX deviation configured in AS profile.
- 350 "Security Check. bad token": Token was not accepted by the LRC

Sample CURL Command to turn ON the yellow LED on the mote with devEUI 000000000D177804

```
curl -H "Content-type:application/x-www-form-urlencoded" -X POST
"https://proxy1.lpn.swisscom.ch/thingpark/lrc/rest/downlink?DevEUI=000000000D177
804&FPort=1&Payload=01"
```

Use the same command with Payload=00 to turn off the LED.

NOTE: regarding queueing of several messages: The Swisscom LPN Portal Wireless network may queue up to 5 messages per device. The network uses the FPending flag defined in the LoRa WAN protocol to signal to the device that additional messages are queued. Messages will be sent, one at a time, in the receive window following the next uplink message received from the device.

4.3.2 Downlink confirmed Application server payload

Unconfirmed Downlink messages are not acknowledged at LoRaWAN level and therefore the network, and the tunnel mode Application server does not know whether they have been received or not.

Confirmed Downlink messages are acknowledged by the target device, but LORAWAN SECTION 4.3.1.2 "Message acknowledge bit and acknowledgement procedure (ACK in FCtrl)" lets the device free of sending delayed ACKs. Therefore, it is not possible to let the network manage retransmissions.

When the LRC receives a possibly empty (no payload) uplink message with ACK set in the FCtrl field, the LRC will add an "ACKbit" flag in the XML metadata of the uplink frame sent to the Application server. The retransmit policy is up to the application server.

4.3.3 Downlink multicast

At Phy level Multicast support in LoRaWAN class C amounts to having multiple End devices listen to the same network address. This is already supported as part of class C support in the Swisscom LPN Portal, an Application server just needs to send an unconfirmed downlink message to the target group address (configured as dummy device).

However, the LoRaWAN™ roadmap includes future work on multicast, including group membership and key management procedures, as well as large payload fragmentation transmission e.g. for firmware updates.

4.3.4 Application Server authentication for downlink frame

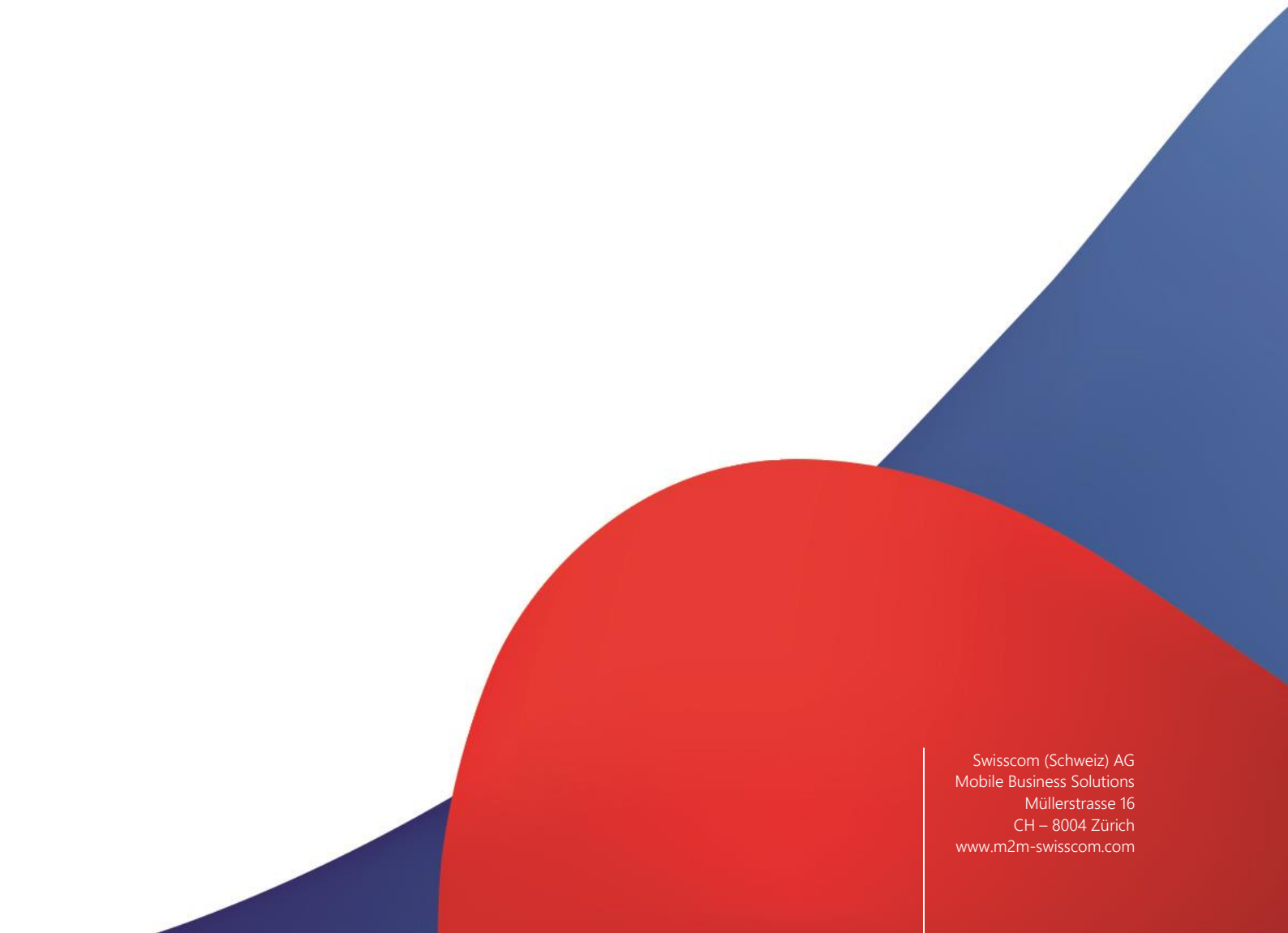
Securing downlink frame is implemented with the following principles.
The AS must not be able to send downlink POST if:

- The AS is not in possession of AS key.
- The AS has not been authorized to send downlink packet to the device
- The time between the generation of the request by the AS and the reception of the request by the LRC is too high.

The AS ID / AS Key and max time deviation are part of the AS profile configuration associated to the device. The Application Server authentication can be deactivated by not setting an AS_ID and AS key in the AS Profile.

Token must be computed as following by the Application Server:

- The downlink message <query-parameters> (Query parameters must include the AS_ID and the Time query parameters) are constructed WITHOUT the Token:
  `e.g. <query-parameters> := DevEUI=000000000F1D8693&FPort=1&Payload=00&AS_ID=app1.sample.com&Time=2016-01-11T14:28:00.333+02:00`
- The <token> is computed as SHA-256(<query-parameters><Askey>)
  `e.g. <token> := SHA-256(DevEUI=000000000F1D8693&FPort=1&Payload=00&AS_ID=app1.sample.com&Time=2016-01-11T14:28:00.333+02:0046ab678cd45df4a4e4b375Eacd096acc)`
  where `46ab678cd45df4a4e4b375Eacd096acc` is the 128 bits pre-shared key (lower case hex string representation) between the Application Server and the LRC as defined in the AS profile.
- The <token> is encoded as an hex string (e.g. `ea8f31d2299cbece8e180a3012766c4df15fe3cf2e142d9fdf4035b5894ec886`) AND added at the end of the query parameters line
  `e.g.` https://proxy1.lpn.swisscom.ch/thingpark/lrc/rest/downlink?DevEUI=000000000F1D8693&FPort=1&Payload=00&AS_ID=app1.sample.com&Time=2016-01-11T14%3A28%3A00.333%2B02%3B00&Token=ea8f31d2299cbece8e180a3012766c4df15fe3cf2e142d9fdf4035b5894ec886
  where: `2016-01-11T14%3A28%3A00.333%2B02%3A00` contains the "%2B" ASCII code for the "+" character and the %3A ASCII code for the ":" character.



Swisscom (Schweiz) AG
Mobile Business Solutions
Müllerstrasse 16
CH – 8004 Zürich
www.m2m-swisscom.com