

Anybus[®] Wireless Bolt IoT[™]/Bolt LTE[™]

REST Commands

REFERENCE GUIDE

SCM-1202-156 1.1 en-US ENGLISH



Important User Information

Disclaimer

The information in this document is for informational purposes only. Please inform HMS Networks of any inaccuracies or omissions found in this document. HMS Networks disclaims any responsibility or liability for any errors that may appear in this document.

HMS Networks reserves the right to modify its products in line with its policy of continuous product development. The information in this document shall therefore not be construed as a commitment on the part of HMS Networks and is subject to change without notice. HMS Networks makes no commitment to update or keep current the information in this document.

The data, examples and illustrations found in this document are included for illustrative purposes and are only intended to help improve understanding of the functionality and handling of the product. In view of the wide range of possible applications of the product, and because of the many variables and requirements associated with any particular implementation, HMS Networks cannot assume responsibility or liability for actual use based on the data, examples or illustrations included in this document nor for any damages incurred during installation of the product. Those responsible for the use of the product must acquire sufficient knowledge in order to ensure that the product is used correctly in their specific application and that the application meets all performance and safety requirements including any applicable laws, regulations, codes and standards. Further, HMS Networks will under no circumstances assume liability or responsibility for any problems that may arise as a result from the use of undocumented features or functional side effects found outside the documented scope of the product. The effects caused by any direct or indirect use of such aspects of the product are undefined and may include e.g. compatibility issues and stability issues.

Table of Contents

Page

1	Preface	3
1.1	About This Document	3
1.2	Document History	3
1.3	Trademarks.....	3
2	Authentication.....	4
3	Public REST Commands.....	5
3.1	Read Configuration.....	5
3.2	Write Configuration.....	5
3.3	Backup Configuration.....	7
3.4	Restore Configuration	7
3.5	Factory Reset	8
3.6	Info	8
3.7	Log	10
3.8	SysLogInfo	11
3.9	Reboot.....	11
3.10	Ultra Low Power Mode	12
3.11	Firmware Update	12
3.12	DHCP Leases	13
3.13	Return Values.....	13
3.14	SIM Card	13
3.15	Return Values.....	14
3.16	Positioning.....	15
3.17	Return Values.....	16

This page intentionally left blank

1 Preface

1.1 About This Document

This document describes the available REST commands for Wireless Bolt IoT and Wireless Bolt LTE. The reader of this document is expected to be familiar with the product and have a good knowledge of wireless communication and network technology.

For additional related documentation, file downloads and technical support, please visit the Anybus support website at www.anybus.com/support.

1.2 Document History

Revision list		
Version	Date	Description
1.0	2020-04-23	First release
1.1	2020-07-05	New sections about SIM Card, returns status of SIM PIN and PUK and Positioning, return the position (GNSS) of the device.

1.3 Trademarks

Anybus® is a registered trademark. Wireless Bolt IoT™ and Wireless Bolt LTE™ are trademarks of HMS Networks AB. All other trademarks mentioned in this document are the property of their respective holders.

2 Authentication

All http requests to the REST API need to be authenticated using the "Basic HTTP Authentication Scheme" (RFC7617).

Basic auth is done by adding a header-parameter to the HTTP-request:

`Authorization: Basic <Base64 encoding of user-id:password>`

The user-id is always "admin" and the password is the web password currently in use, either the default password shipped with the product, or if it has been changed, the new password.

Example:

User-id: admin

Password: abc123

String to Base64-encode: admin:abc123

Completed header parameter:

`Authorization: Basic YWRtaW46YWJjMTIz`

3 Public REST Commands

3.1 Read Configuration

Read information from a configuration tree and present as JSON data.

3.1.1 Syntax

```
GET <target_ip>/cgi-bin/readconfigtree.cgi?path=<tree_path>
```

3.1.2 Query Parameters

Name	Description
target_ip	IP address of the Bolt IoT device, e.g. 192.168.0.98
tree_path	A path to the configuration tree or node to read

3.1.3 Return Values

A JSON structure with the requested data.

3.1.4 Examples

Read a single configuration node:

Query:

```
GET http://192.168.0.98/cgi-bin/readconfigtree.cgi?path=web:/settings/lock
```

Response:

```
{"name":"lock","type":"bool","value":true}
```

Read a configuration tree:

Query:

```
GET http://192.168.0.98/cgi-bin/readconfigtree.cgi?path=web:
```

Response:

```
{"name":"","type":"tree","children":  
[{"name":"settings","type":"stem","children":  
[{"name":"lock","type":"bool","value":true}]}]}
```

3.2 Write Configuration

Write information to a configuration tree.

3.2.1 Syntax

```
GET <target_ip>/cgi-bin/writeconfigtree.cgi?path=<tree_path>&value=<new_value>
```

3.2.2 Query Parameters

Name	Description
target_ip	IP address of the Bolt IoT device, e.g. 192.168.0.98
tree_path	A path to the tree and node to operate on
new_value	A value to write to the config tree

3.2.3 Configuration Paths

The following configuration paths can be modified by the user:

Path	Description
router:/settings/ipAddr	IP address
router:/settings/netMask	Subnet mask
router:/settings/dhcpEnable	Enable internal DHCP server: true or false
router:/settings/auth	Enable APN authentication: true or false
router:/settings/authUser	APN authentication: user
router:/settings/authPass	APN authentication: password
router:/settings/dhcpRangeStart	DHCP start address: 1..254
router:/settings/dhcpRangeStop	DHCP stop address: dhcpRangeStart..254
router:/settings/apn	Access Point Name (APN)
router:/settings/rat	Radio Access Technology (RAT): auto, lte-m1+gsm, lte-m1+lte-nb1, lte-nb1+gsm, lte-nb1+gsm, lte-m1, lte-nb1 or gsm
router:/settings/ratOrder	Modem Preset list, RAT search order: The six combinations of LTE-M1,LTE-NB1,GSM (IoT only)
router:/settings/natMode	NAT mode: disabled, nat11
router:/settings/natTarget	Target local IP for NAT1:1
router:/settings/natSourceFilter	NAT source filter: any, host, network, range
router:/settings/natSourceIp	NAT source IP(s): Comma separated list of up to 5 IPs (only applicable if filter is "host")
router:/settings/natSourceBaseIp	NAT source base IP (only applicable if filter is "network")
router:/settings/natSourceMask	NAT source netmask (only applicable if filter is "network")
router:/settings/natSourceRangeStart	NAT source range start IP (only applicable if filter is "range")
router:/settings/natSourceRangeStop	NAT source range stop IP (only applicable if filter is "range")
web:/settings/lock	Configuration lock: true or false
web:/settings/password	Web and REST API password (valid characters a-z, A-Z, 0-9, - and _)
position:/settings/enable	Positioning service enable: true or false

A reboot is needed for changes to be applied and used.

3.2.4 Return Values

```
{"success":<value>}
```

Value	Description
true	Request completed successfully.
false	Request failed or validation error or config locked.

3.2.5 Example

Query:

```
GET http://192.168.0.98/cgi-bin/writeconfigtree.cgi?path=router:/settings/apn&value=my_apn.company.com
```

Response:

```
{"success":true}
```


3.2.6 Example

Query:
http://192.168.0.99/cgi-bin/writeconfigtree.cgi?path=router:/settings/ratOrder&value=GSM,LTE-NB1,LTE-M1
Response:
{ "success":true }

3.2.7 Example

Query:
GET http://192.168.0.98/cgi-bin/writeconfigtree.cgi?path=router:/settings/natSourceFilterMode&value=host
Response:
{ "success":true }

Query:
GET http://192.168.0.98/cgi-bin/writeconfigtree.cgi?path=router:/settings/natSourceIp&value=10.10.0.1,10.20.0.1,10.30.0.1,10.40.0.1,10.50.0.1
Response:
{ "success":true }

3.3 Backup Configuration

Create and return a file with a backup of all settings in the device.

3.3.1 Syntax

GET <target_ip>/cgi-bin/backup.cgi

3.3.2 Query Parameters

None

3.3.3 Examples

Query:
GET http://192.168.0.98/cgi-bin/backup.cgi
Response:
<octet stream containing the backup>

3.4 Restore Configuration

Restore all settings in the device from a file.

3.4.1 Syntax

POST <target_ip>/cgi-bin/restore.cgi

3.4.2 Query Parameters

None

3.4.3 Post Data

The backup file contents.

3.4.4 Return Values

```
{"success":<value>}
```

Value	Description
true	Request completed successfully.
false	Request failed or validation error or config locked.

3.4.5 Examples

Query:

```
POST http://192.168.0.98/cgi-bin/restore.cgi
```

Response:

```
{"success":true}
```

3.5 Factory Reset

Restore all settings in the device to factory default values.

3.5.1 Syntax

```
GET <target_ip>/cgi-bin/factoryreset.cgi
```

3.5.2 Query Parameters

None

3.5.3 Return Values

```
{"success":<value>}
```

Value	Description
true	Request completed successfully.
false	Request failed or config locked.

3.5.4 Examples

Query:

```
GET http://192.168.0.98/cgi-bin/factoryreset.cgi
```

Response:

```
{"success":true}
```

3.6 Info

Get runtime information.

3.6.1 Syntax

```
GET <target_ip>/cgi-bin/info.cgi
```

3.6.2 Query Parameters

None

3.6.3 Return Values

A JSON structure with the following information.

Value	Description
uptime	Device uptime in seconds
time	System time (UTC)
radio_power	Radio module power state
	0: Off
	1: On
sim	SIM card status
signal_strength	Signal strength (0..5)
signal_strength_raw	Signal strength raw CSQ (0..31)
signal_quality	Signal quality (dBm)
cell_id	Serving Cell Identifier
lac	Location Area Code of the serving cell
tac	Tracking Area Code of the serving cell (LTE only)
rat	Currently used Radio Access Technology
	0: Unknown
	1: GSM
	2: UMTS
	4: LTE
operator	Currently used operator
status	Network registration status
	0: Not registered and not searching
	1: Registered to home network
	2: Searching for new operator
	3: Registration denied
	4: Registered to roaming network
	5: Unknown state
amplifier_temp	Amplifier temperature in degrees Celcius
controller_temp	Controller temperature in degrees Celcius
connection_state	Data session state
	0: Disconnected
	1: Authenticating
	2: Connected
	3: Suspending
	4: Incoming (MT-PDP context request)
voltage	Voltage (mV)
iotbolt_version	Bolt IoT version
modem_model	Modem model
modem_version	Modem FW version
pri	Carrier name and version
apn	Currently used APN
rat_specific	Access technology of the serving cell
	0: GSM
	1: GSM Compact
	2: UTRAN

Value	Description
	3: GSM w/EGPRS
	4: UTRAN w/HSDPA
	5: UTRAN w/HSUPA
	6: UTRAN w/HSDPA and HSUPA
	7: CAT-M1
	8: GSM-IoT
	9: NB-IoT
imsi	International Mobile Subscriber Identity
imei	International Mobile Equipment Identity
cellular_gateway	Cellular Gateway IP Address
cellular_ip	Cellular IP Address
iccid	SIM card ICCID
active_band	Used frequency band

3.6.4 Examples

Query:

```
GET http://192.168.0.98/cgi-bin/info.cgi
```

Response:

```
{ "uptime": "4287", "time": "2019-11-19 13:35:48", "radio_
power": "1", "sim": "2", "signal_strength": "4", "signal_strength_
raw": "23", "signal_quality": "1", "cell_
id": "26269452", "lac": "4294967295", "tac": "0x85", "rat": "4", "opera-
tor": "TELIA S", "status": "4", "amplifier_temp": "35", "controller_
temp": "35", "connection_state": "2", "voltage": "3807", "iotbolt_
version": "1.00.20-dev", "modem_model": "WP7607", "modem_
version": "SWI9X06Y_02.16.06.00", "pri": "GENERIC_001.028_
001", "apn": "", "rat_
specific": "9", "imsi": "238208700452254", "imei": "352653090129735", "-
cellular_
ip": "10.81.177.33", "iccid": "89450421170203490357", "active_
band": "B20" }
```

3.7 Log

Get system log.

3.7.1 Syntax

```
GET <target_ip>/cgi-bin/log.cgi
```

3.7.2 Query Parameters

None

3.7.3 Return Values

Plain text log file.

3.7.4 Examples

Query:

```
GET http://192.168.0.98/cgi-bin/log.cgi
Response:
Oct 9 10:03:18 Legato: INFO | Version: 1.00.18-dev
Oct 9 10:03:18 Legato: INFO | Boot reason: Power-on
Jan 6 00:00:23 Legato: INFO | eth0 IP address: 192.168.0.98 - netmask:
255.255.254.0
Jan 6 00:00:23 Legato: INFO | eth0 DHCP server: on
Jan 6 00:00:23 Legato: INFO | APN auth is: off
Jan 6 00:00:23 Legato: INFO | DHCP range start - stop: 192.168.0.100 -
192.168.0.200
Jan 6 00:00:24 Legato: INFO | IP address/netmask set to: 192.168.0.98/
255.255.254.0
Jan 6 00:00:24 Legato: INFO | Setting no auth for APN
Jan 6 00:00:24 Legato: INFO | Setting configured RAT: "gsm": 0x01
Jan 6 00:00:24 Legato: INFO | Data connection requested
Oct 17 12:02:43 Legato: INFO | Data connected, interface 'rmnet_data0'
Oct 17 12:02:43 Legato: INFO | Restarting DHCP/DNS services
```

3.8 SysLogInfo

Create and return an archive of all system logs.

3.8.1 Syntax

```
GET <target_ip>/cgi-bin/sysloginfo.cgi
```

3.8.2 Query Parameters

None

3.8.3 Examples

```
Query:
GET http://192.168.0.98/cgi-bin/sysloginfo.cgi
Response:
<octet stream containing the archive>
```

3.9 Reboot

Reboot the device.

3.9.1 Syntax

```
GET <target_ip>/cgi-bin/reboot.cgi
```

3.9.2 Query Parameters

None

3.9.3 Return Values

```
{"success":<value>}
```

Value	Description
true	Request completed successfully.
false	Request failed

3.9.4 Examples

Query:
GET http://192.168.0.98/cgi-bin/reboot.cgi
Response:
{ "success":true }

3.10 Ultra Low Power Mode

Put the device in Ultra Low Power Mode for a specified duration.
Note: Bolt LTE US (WP7610) does not support ULPM.

3.10.1 Syntax

GET <target_ip>/cgi-bin/ulpm.cgi?duration=<time in seconds>

3.10.2 Query Parameters

Name	Description
duration	Number of seconds (60-86400) to remain in ULPM before waking up again

3.10.3 Return Values

{ "success":<value>, "message":"<additional info>" }

Value	Description
true	Request completed successfully.
false	Request failed or validation error

3.10.4 Examples

Enter ULPM and sleep for 300 seconds (5 minutes):

Query:
GET http://192.168.0.98/cgi-bin/ulpm.cgi?duration=300
Response:
{ "success":true, "message":"sleeping for 300 s" }

3.11 Firmware Update

Update the device firmware.

3.11.1 Syntax

POST <target_ip>/cgi-bin/update.cgi

3.11.2 Query Parameters

None

3.11.3 Post Data

The contents of a firmware (.fws) file.

3.11.4 Return Values

Raw HTML output of the firmware update process. Searching this for the strings "SUCCESS" or "FAILED" will give the result.

3.11.5 Examples

```
Query:
POST http://192.168.0.98/cgi-bin/update.cgi
Response:
Trying to update...
Unpacking: 0%
FAILED: error code 2
```

3.12 DHCP Leases

Return a list of active DHCP leases.

3.12.1 Syntax

```
GET <target_ip>/cgi-bin/dhcpleases.cgi
```

3.12.2 Query Parameters

None

3.13 Return Values

Raw text output with DHCP leases in dnsmasq.leases format.

3.13.1 Examples

```
Query:
GET http://192.168.0.98/cgi-bin/dhcpleases.cgi
Response:
1573275080 00:e0:4c:34:92:e9 192.168.0.168 LT-5Q1XWT2 01:00:
e0:4c:34:92:e9
```

3.14 SIM Card

Returns status of SIM PIN and PUK, or outcome of a SIM PIN or SIM PUK entering attempt. Entered and correctly validated SIM PIN will be used on subsequent boots as long as the same sim card is inserted.

When PUK is needed, a new (any, as desired) PIN is also needed as input. The new PIN will be used on subsequent boots as long as the same sim card is inserted.

3.14.1 Syntax

Query Parameter	Description	
status	Request for status about PIN and PUK.	

Query Parameter	value	Description
pin	pin value	Request for validation with PIN.

Query Parameter	value	Description
puk	puk value	Request for unblocking SIM by PUK.

GET <target_ip>/cgi-bin/simcard.cgi?status=

GET <target_ip>/cgi-bin/simcard.cgi?pin=12345678

GET <target_ip>/cgi-bin/simcard.cgi?pin=12345678&puk=87654321

3.14.2 Query Parameters

Name	Description
status	Request status about SIM PIN
pin	Enter SIM PIN
puk	Unblock SIM by PUK, and set new PIN

3.15 Return Values

```
{"success":<value>,"needspin":<pin validation status>,"needspuk":<puk unblock sim status>"message": "<additional info>"}
```

value	Description
true	SIM is ready
false	Some action is needed, insert SIM, enter PIN or PUK, etc

pin validation status	Description
true	SIM PIN is needed
false	SIM PIN not needed

puk unblock sim status	Description
true	Unblock SIM by PUK is needed
false	Unblock SIM by PUK is not needed

3.15.1 Example

Query:

GET <target_ip>/cgi-bin/simcard.cgi?status=

Responses:

```
{"success":true,"needspin":false,"needspuk":false,"message":"SIM PIN code correctly entered, or not required. "}
```



```
{ "success":false,"needspin":false,"needspuk":false,"message":"SIM
card absent. " }
{ "success":false,"needspin":true,"needspuk":false,"message":"SIM
card present and needs PIN, 3 tries left. " }
{ "success":false,"needspin":false,"needspuk":true,"message":"SIM
card blocked, 10 PUK tries left. " }
```

3.15.2 Example

Query:

```
GET <target_ip>/cgi-bin/simcard.cgi?pin=0000
```

Responses:

```
{ "success":true,"needspin":false,"needspuk":false,"message":"SIM
PIN code correctly entered, or not required.
{ "success":true,"needspin":false,"needspuk":false,"message":"SIM
PIN code correctly entered, or not required. SIM PIN validated
successfully, and stored for subsequent use" }
{ "success":false,"needspin":true,"needspuk":false,"message":"SIM
card present and needs PIN, 2 tries left. SIM PIN validation failed" }
{ "success":false,"needspin":false,"needspuk":true,"message":"SIM
card blocked, 10 PUK tries left. SIM PIN validation failed" }
```

3.15.3 Example

Query:

```
GET <target_ip>/cgi-bin/simcard.cgi?pin=12345678&puk=87654321
```

Responses:

```
{ "success":true,"needspin":false,"needspuk":false,"message":"SIM
PIN code correctly entered, or not required.
{ "success":true,"needspin":false,"needspuk":false,"message":"SIM
PIN code correctly entered, or not required. Successfully unblocked the
SIM, and stored PIN for subsequent use" }
{ "success":false,"needspin":true,"needspuk":false,"message":"SIM
card present and needs PIN, 3 tries left. " }
{ "success":false,"needspin":false,"needspuk":true,"message":"SIM
card blocked, 9 PUK tries left. Set PUK failed" }
{ "success":false,"needspin":false,"needspuk":false,"message":"SIM
PIN terminated, no PUK tries left. Set PUK failed" }
```

3.15.4 Example

Queries:

```
GET <target_ip>/cgi-bin/simcard.cgi?something
```

```
GET <target_ip>/cgi-bin/simcard.cgi?puk=12345678
```

Response:

```
{ "success":false,"message":"Missing parameters: no 'status' or
'pin'" }
```

3.16 Positioning

Return the position (GNSS) of the device.

3.16.1 Syntax

GET <target_ip>/cgi-bin/position.cgi

3.16.2 Query Parameters

None

3.17 Return Values

A JSON structure with the following information.

Value	Description
state	State of positioning service
	Disabled: The GNSS device is disabled
	Uninitialized: The GNSS device is not initialized
	Ready: The GNSS device is ready
	Active: The GNSS device is active
	Unknown: State is not known
valid	Valid position (true/false)
latitude	Latitude (decimal value)
longitude	Longitude (decimal value)
accuracy	Accuracy in meters

3.17.1 Examples

Query:

GET http://192.168.0.98/cgi-bin/position.cgi

Response:

```
{"state": "Active", "valid": true, "latitude": 56.665728, "longitude": 12.863710, "accuracy": 26.000000}
```

This page intentionally left blank

