LoRaWAN® Firmware Management Protocol Specification TS006-1.0.0-rc4

NOTICE OF USE AND DISCLOSURE

Copyright © 2021 LoRa Alliance, Inc. All rights reserved. The information within this document is the property of the LoRa Alliance ("The Alliance") and its use and disclosure are subject to LoRa Alliance Corporate Bylaws, Intellectual Property Rights (IPR) Policy and Membership Agreements.

Elements of LoRa Alliance specifications may be subject to third-party intellectual property rights, including without limitation, patent, copyright or trademark rights (such a third party may or may not be a member of the LoRa Alliance). The Alliance is not responsible and shall not be held responsible in any manner for identifying or failing to identify any or all such third-party intellectual property rights.

This document and the information contained herein are provided on an "AS IS" basis and THE ALLIANCE DISCLAIMS ALL WARRANTIES EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO (A) ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OF THIRD PARTIES (INCLUDING WITHOUT LIMITATION ANY INTELLECTUAL PROPERTY RIGHTS INCLUDING PATENT, COPYRIGHT OR TRADEMARK RIGHTS) OR (B) ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NONINFRINGEMENT. IN NO EVENT WILL THE ALLIANCE BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF USE OF DATA, INTERRUPTION OF BUSINESS, OR FOR ANY OTHER DIRECT, INDIRECT, SPECIAL OR EXEMPLARY, INCIDENTIAL, PUNITIVE OR CONSEQUENTIAL DAMAGES OF ANY KIND, IN CONTRACT OR IN TORT, IN CONNECTION WITH THIS DOCUMENT OR THE INFORMATION CONTAINED HEREIN, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The above notice and this paragraph must be included on all copies of this document.

LoRa Alliance®

5177 Brandin Court

Fremont, CA 94538

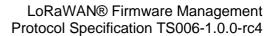
35 United States

Note: LoRa Alliance® and LoRaWAN® are trademarks of the LoRa Alliance, used by permission. All company, brand and product names may be trademarks that are the sole property of their respective owners.





41	
42	
43	LoRaWAN® Firmware Management
44	Protocol Specification
45	TS006-1.0.0-rc4
46 47 48	Authored by the FUOTA Working Group of the LoRa Alliance Technical Committee
4 0 49	Technical Committee Chair and Vice-Chair:
50 51	A. YEGIN (Actility), O. SELLER (Semtech)
52	Working Group Chair:
53 54	J. CATALANO (Kerlink)
55	Editor:
56 57	J. CATALANO (Kerlink)
58	Contributors:
59 60 61	J. CATALANO (Kerlink), N. SORNIN (Semtech), J. SWETINA (NEC)
62	Version : 1.0.0-rc4
63	Date : May 28, 2021
64	Status: RELEASE CANDIDATE

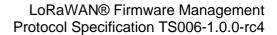




Contents

66

1 Revisions (Remove in final version) Error! Bookmark not defined.4
2 Conventions45
3 Introduction <u>5</u> 6
4 Firmware management package
4.1 Package Version Commands (Package Version Req, Package Version Ans)
4.2 Device Version Commands (DevVersionReq, DevVersionAns)
4.3 Device Reboot Time Commands (DevRebootTimeReq, DevRebootTimeAns)78
4.4 Device Reboot Countdown Commands (DevRebootCountdownReq,
DevRebootCountdownAns)91
4.5 Device Upgrade Image Commands (<i>DevUpgradeImageReq</i> , <i>DevUpgradeImageAns</i>) <u>10</u> 14
DevUpgradeImageAns) <u>10</u> 44
4.6 Device Delete Image Commands (DevDeleteImageReq, DevDeleteImageAns) 1142
5 Glossary
6 Bibliography14
6.1 References14
Tables
Table 1: Firmware Management commands summary67
Table 2: PackageVersionAns
Table 3: DevVersionAns
Table 4: DevRebootTimeReq
Table 5: DevRebootTimeAns89
Table 6: DevRebootCountdownReq910
Table 7: DevRebootTimeAns1044
Table 8: DevUpgradeImageAns1044
Table 9: DevUpgradeImageAns Status fields
Table 10 : UpImageStatus field encoding
Table 11: DevDeleteImageReq
Table 12: DevDeleteImageAns
Table 13: DevDeleteImageAns Status fields





1 Conventions

In this document:

100 101 102

103

104

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in IETF Best Current Practice 14 (BCP14 [RFC2119] [RFC8174]) when, and only when, they appear in all capitals, as shown here.

105 106 107

The tables in this document are normative. The figures and notes in this document are informative.

108 109 110

Document titles are written as *LoRaWAN Link Layer Specification* and sections within a document are written as "Frequency-Hopping Beacon Transmission".

111 112 113

Commands are written **PackageVersionReq**, bits and bit fields are written PackageIdentifier, constants are written RECEIVE_DELAY1, variables are written N.

114 115

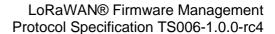
• The octet order for all multi-octet fields SHALL be little endian.

116

• EUI are 8-octet fields and SHALL be transmitted as little endian.

117118

• By default, RFU bits are Reserved for Future Use and SHALL be set to 0 by the transmitter of the packet and SHALL be silently ignored by the receiver.





Introduction

120 121 122

This document proposes a standard application layer firmware management protocol allowing to query / manage the following end-device's properties.

- 124
- What is the version of the firmware running on the end-device?
- 123 125
- What is the end-device's hardware version?
- 126
- Reboot the end-device at a given time
- 128 129 130
- Is an image available on the end-device for firmware upgrade, and what version? 127
 - Delete an image



3 Firmware Management Package

 The PackageIdentifier of the firmware management package is 4. The PackageVersion of this package is version 1.

The following messages are sent to each end-device individually, using unicast downlink on a port specifically used for the firmware management package. It is RECOMMENDED that a default port value of 203 be used to promote interoperability, though implementations MAY use a value from those not assigned by the LoRa Alliance[®].

These messages SHALL NOT be sent using multicast. If these messages are received on a multicast address the end-device SHALL drop them silently.

These commands SHALL be invoked using one of the solutions described in *LoRaWAN Multi Package Access Protocol Specification* [TS007].

The following table summarizes the list of firmware management commands.

CID	Command name	Transmitted by		Short Description
		End- device	App Server	
0x00	PackageVersionReq		Х	Used by the Application Server to request the package version implemented by the end-device
0x00	PackageVersionAns	Х		Conveys the answer to Package VersionReq
0x01	DevVersionReq		Х	Asks an end-device for its hardware version, and its currently running firmware version
0x01	DevVersionAns	X		Conveys the answer to the DevVersionReq request
0x02	DevRebootTimeReq		х	Instructs an end-device to program a reboot at a given time or immediately
0x02	DevRebootTimeAns	Х		Conveys the answer to DevRebootTimeReq
0x03	DevRebootCountdownReq		х	Instructs an end-device to program a reboot after a certain period of time has elapsed
0x03	DevRebootCountdownAns	х		Conveys the answer to DevRebootCountdownReq
0x04	DevUpgradelmageReq		Х	Asks an end-device if a firmware upgrade image is present in memory, its version and status
0x04	DevUpgradelmageAns	Х		Conveys answer to the DevUpgradeReq request
0x05	DevDeletelmageReq		Х	Instructs the end-device to delete a given firmware upgrade image
0x05	DevDeletelmageAns	Х		Conveys answer to the DevDeleteImageReq request

Table 1: Firmware Management commands summary



3.1 Package Version Commands (*PackageVersionReq*, *PackageVersionAns*)

The **PackageVersionReg** command has no payload.

The end-device SHALL answer this command with a *PackageVersionAns* command with the following payload.

Size (octets)	1	1
Field	PackageIdentifier	PackageVersion
•	Table 2. Dealers Varaion Ana	

Table 2: PackageVersionAns

 ${\tt PackageIdentifier} \ uniquely \ identifies \ the \ package.$

 Package Version corresponds to the major version of the package specification implemented by the end-device.

3.2 Device Version Commands (DevVersionReq, DevVersionAns)

The **DevVersionReg** command has no payload.

The end-device SHALL respond to this command with a **DevVersionAns** command with the following payload:

Size (octets)	4	4
Field	FW version	HW version
	Table 3: DevVersionA	ns

The FW version and HW version fields are device manufacturer specific and freely allocated.

Example: the HW version field might be composed of a 24bits hardware identifier followed by an 8 bits revision index of this hardware platform.

Example: the FW version field might be a 32bits CRC of the firmware currently running on the end-device.

3.3 Device Reboot Time Commands (*DevRebootTimeReq*, *DevRebootTimeAns*)

The **DevRebootTimeReq** command is used by the Application Server to instruct an end-device's application to program a hardware reboot either immediately or at a given time. If a valid firmware upgrade image is present at the time of reboot this upgrade SHALL be installed and the end-device SHALL reboot with a new firmware version.

The command's payload is:





190 Reboot Time encodes the time at which the reboot SHALL take place.

A value > 0 but < 0xFFFFFFF of RebootTime encodes the point in time, expressed in seconds since GPS epoch, *i.e.* Sunday January the 6th 1980 at midnight, (refer to the LoRaWAN Link Layer Specification [TS001] **DeviceTimeReq** MAC command) at which reboot takes place.

A value of 0x00000000 instructs the end-device to reboot as soon as possible.

A value of 0xFFFFFFF instructs the end-device to cancel a currently programmed reboot.

196 197 198

195

The end-device SHALL only store a single programmed reboot event, meaning that any received valid **DevRebootTimeReq** command or **DevRebootCountdownReq** command overwrites all previously received reboot commands.

199 200 201

202

203

204

205

206

207208

209210

211

212213

214215

216

217218

219

220 221

222

Behavior of the end-device upon reception of a *DevRebootTimeReq* command:

- If the RebootTime field has a value of 0x00000000 the end-device SHALL reboot as soon as possible. If the RebootTime field has a value > 0 but < 0xFFFFFFF:
 - 1. If the end-device knows the current time and if
 - The value of the RebootTime field indicates a time in the future then the end-device SHALL reboot at that time.
 - The value of the RebootTime field indicates a time in the past then the end-device SHALL report an error with a *DevRebootTimeAns* command with a value of RebootTime of 0x00000000.
 - 2. If the end-device does not know the current time but is capable of measuring duration and is activated on a LoRaWAN 1.0.3 or later compatible Network Server then the end-device MAY request the current time from the Network Server using the **DeviceTimeReq** command (refer to the LoRaWAN Link Layer Specification [TS001] **DeviceTimeReq** MAC command) and proceed as in case (1).
 - 3. In all other cases the end-device SHALL report an error with a **DevRebootTimeAns** command with a value of RebootTime of 0x00000000.
- If the RebootTime field has a value of 0xFFFFFFF the end-device SHALL cancel a currently programmed reboot and SHALL acknowledge with a **DevRebootTimeAns** command with a value of RebootTime of 0xFFFFFFF.
- If a valid firmware upgrade image is present at the end-device, then that image SHALL be installed at reboot.

223224225

226

227

228229

DevRebootTimeAns command:

If the end-device reboots following a 0x00000000 command, it SHALL NOT answer with a **DevRebootTimeAns** command.

In all other cases, the end-device SHALL sends a **DevRebootTimeAns** command back with the following payload:

230231



232

233

234

235

236

The RebootTime field of the answer encodes the number of seconds between sending the **DevRebootTimeAns** message until reboot.

The **DevRebootTimeAns** command can also indicate an error or acknowledge a reboot cancellation:



- A value of 0x00000000 of the RebootTime field in the **DevRebootTimeAns** message indicates the inability of the end-device to reboot at the requested time.
- 239 A value of 0xFFFFFFFF of the RebootTime field in the DevRebootTimeAns message
- 240 acknowledges the cancellation of a currently programmed reboot.

3.4 Device Reboot Countdown Commands (*DevRebootCountdownReq*, *DevRebootCountdownAns*)

The **DevRebootCountdownReq** command is used by the Application Server to instruct an end-device's application to program a hardware reboot after a given time period. If a valid firmware upgrade image is present at the time of reboot this upgrade SHALL be installed and the end-device SHALL reboot with a new firmware version.

The command's payload is:

241

242243244

245

246

247

248249

250

251252

253254

255

256257

258 259 260

261

262

263264

265266

267

268

269 270

271

272273

274275

276277

278

279280

281

282

Size (octets)	3	
Field	Countdown	

Table 6: DevRebootCountdownReq

A value > 0 but < 0xFFFFFF of the 24-bit integer Countdown field encodes the number of seconds after which, after reception of the *DevRebootCountdownReq* command, the end-device SHALL reboot.

Note: the maximum allowed value 0xFFFFE corresponds to 16777214 seconds or 194 days, 4 hours, 20 minutes and 14 seconds.

A value of 0x000000 instructs the end-device to reboot as soon as possible. A value of 0xFFFFFF instructs the end-device to cancel a currently programmed reboot.

The end-device SHALL only store a single programmed reboot event, meaning that any received valid **DevRebootTimeReq** or **DevRebootCountdownReq** command overwrites all previously received reboot commands.

Behavior of the end-device upon reception of a **DevRebootCountdownReq** command:

- If the Countdown field has a value of 0x000000 the end-device SHALL reboot as soon as possible.
- If the Countdown field has a value > 0 but < 0xFFFFFF:
 - If the end-device is capable of measuring time duration in seconds the enddevice SHALL reboot when the duration indicated by the Countdown field has elapsed.
 - o If the end-device is not capable of measuring time duration in seconds the end-device SHALL report an error with a **DevRebootCountdownAns** command with a value of Countdown of 0x000000.
- If the Countdown field has a value of 0xFFFFFF the end-device SHALL cancel a currently programmed reboot and SHALL acknowledge with a **DevRebootCountdownAns** command with a value of Countdown of 0xFFFFFF.
- If a valid firmware upgrade image is present at the end-device, then that image SHALL be installed at reboot.

DevRebootCountdownAns command:

If the end-device reboots following a 0x000000 command, it SHALL NOT answer with a **DevRebootCountdownAns** command.



In all other cases the end-device SHALL send a **DevRebootCountdownAns** command back with the following payload:

Size (octets)	3	
Field	Countdown	

 Table 7: DevRebootTimeAns

The Countdown field of the answer encodes the number of seconds between sending the DevRebootCountdownAns message until reboot.

The **DevRebootCountdownAns** command can also indicate an error or acknowledge a reboot cancellation:

A value of 0x000000 of the Countdown field in the **DevRebootCountdownAns** message indicates the inability of the end-device to reboot at the requested time.

A value of 0xFFFFFF of the Countdown field in the **DevRebootCountdownAns** message acknowledges the cancellation of a currently programmed reboot.

3.5 Device Upgrade Image Commands (*DevUpgradeImageReq*, *DevUpgradeImageAns*)

The **DevUpgradeImageReq** command is used to ask an end-device if a firmware upgrade image is present in its memory, the validity of that firmware upgrade image and its version. The command has no payload.

The end-device SHALL answer with a **DevUpgradeImageAns** message with the following payload.

Size (octets)	1	4
Field	Status	nextFirmwareVersion
		(conditional)
Table 8: DevUpgradeImageAns		

Where

Bits	7:2	1:0
Status Fields	RFU	UpImageStatus

Table 9: DevUpgradeImageAns Status fields

The UpImageStatus field encodes the status of the firmware upgrade image according to the following table.

value	UpImageStatus
0	No firmware upgrade image currently present
1	A firmware upgrade image is present but is either
	corrupted or its cryptographic signature is wrong
2	An authenticated firmware upgrade image is present but does not corresponds to the end-device's hardware platform



3

	3	installe	illillware upgrade illiage is pr	esent and can be	
315	Table 10 : UplmageStatus field encoding				
316 317 318 319 320 321	The nextFirmwareVersion field SHALL be present only when UpImageStatus = 3, in that case it encodes the firmware version that will be running once this firmware upgrade image is installed. Installation of the firmware upgrade image SHALL be performed when the end-device reboots. After successful reboot with a new firmware its firmware upgrade image SHALL be deleted from the end-device and SHALL no longer be reported.				
322 323 324	`	Status	alid or invalid) firmware 1, 2 or 3) is expected to be ove age download takes place.	10	
325 326			nd storage of the firmware up scope of the current specification		
327 328 329	3.6 Device Delete Image Commands (<i>DevDeleteImageReq</i> , <i>DevDeleteImageAns</i>)				
330 331 332	The DevDeleteImageReq command is used to delete a firmware upgrade image currently stored in an end-device. The command payload is:				
	;	Size (octe			
333		FI	FirmwareToDeleteVe Table 11: DevDeleteImageReq	ersion	
334 335 336	The end-device SHAL payload.	L answe	r with a DevDeleteImageAn : Size (octets) 1	s message with the following	
			Field Status		
337			Table 12: DevDeletelmageAns		
338 339	Where				
	Bit: Status Field:		1	0	
340	Status Field	RFU	ErrorInvalidVersion 13: DevDeleteImageAns Status fie	ErrorNoValidImage	
	TI		_		
341	ine ErrorNoValid	ımage b ı	t signals that the end-device h	nas currently no valid firmware	

A valid firmware upgrade image is present and can be

The ErrorNoValidImage bit signals that the end-device has currently no valid firmware upgrade image stored.

The ErrorInvalidVersion bit signals that the version specified in the delete request FirmwareToDeleteVersion field does not match the version of the valid firmware upgrade image currently stored.

If any of those two bits is 1, the delete command did not succeed, the state of the end-device SHALL stay unmodified.

Upon reception of a **DevDeleteImageReq** command, a valid, uncorrupted, and authenticated firmware upgrade image that corresponds to the end-device's hardware platform and that has

342

343 344

345

346

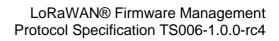
349



351

352

a firmware version equal to Firmware ToDelete Version SHALL be deleted or invalidated by the end-device.





353	4 Gloss	sary
354		-
355	CRC	Cyclic Redundancy Check
356	EUI	Extended Unique Identifier
357	FM	Firmware Management
358	FUOTA	Firmware Update Over-The-Air
359	FW	Firmware
360	GPS	Global Positioning System
361	HW	Hardware
362	RFU	Reserved for Future Use
363		



5 Bibliography

364

365	5.1 References
366 367	[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCF 14, RFC 2119, March 1997
368 369	[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCF 14, RFC8174, May 1997
370	[TS001] LoRaWAN® 1.0.4 Link Layer Specification, LoRa Alliance, October 2020
371 372	[TS003] LoRaWAN® Application Layer Clock Synchronization Specification TS003-2.0.0 LoRa Alliance, <u>TBD</u>
373	[TS004] LoRaWAN® Fragmented Data Block Transport Specification TS004-2.0.0, TBD
374	[TS005] LoRaWAN® Remote Multicast Setup Specification TS005-2.0.0, LoRa Alliance, TBD
375 376	[TS007] LoRaWAN [®] Multi Package Access Protocol Specification TS007-1.0.0, LoRa Alliance, to be published