LwM2M advanced firmware update

Approved Version: 1.0 - 2022-06-14

Open Mobile Alliance

OMA-WID-LwM2M_advanced_firmware_update-V1_0-20220614-A

main: 20 Jun 2022 11:14:00 *rev: aaf2e32*

Use of this document is subject to all of the terms and conditions of the Use Agreement located at https://www.omaspecworks.org/about/policies-and-terms-of-use/.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification.
However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the “OMA IPR Declarations” list at https://www.omaspecworks.org/about/intellectual-property-rights/. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR’S REPRESENTED ON THE “OMA IPR DECLARATIONS” LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

THIS DOCUMENT IS PROVIDED ON AN "AS IS" "AS AVAILABLE" AND "WITH ALL FAULTS" BASIS.

Table of Contents

Table of Tables

1. Work Item Details

Concept	Value
Work Item Title:	LwM2M advanced firmware update
Short Name:	LwM2M advanced firmware update
Version:	1.0

2. Impacts, Relationships and Dependencies

Concept	Value
Existing Specifications Affected	None
Other Work Items affected	None
External Organizations Affected	Feedback to IETF
Any Other Impact	None

3. Supporting Companies

Company Name	Membership Type
AVSystem	Associate
Itron	Full
IoTerop	Full

4. Motivation

The LwM2M specifications defines a Firmware Update Object, which allows a LwM2M Server to determine the firmware version running on the device, to update firmware on the device and to monitor the progress of the update. This Firmware Update Object has been widely implemented and is in deployment today in a number of IoT devices.

With advanced microcontrollers appearing on the market the firmware update process has become more complex since firmware became composed of multiple images, which are necessary for applications to work. Examples include

- microcontrollers containing separate images for a bootloader, modem firmware and application firmware,
- microcontrollers with multiple cores, and
- microcontrollers using modern security features, like TrustZone, with firmware partitioned into secure and non-secure processing environments.

While these firmware images can also be updated independently, they cannot be activated or deactivated, installed independently without considering the dependencies and might require restart of the device to apply the upgrade.

The LwM2M Firmware Update Object does not support multiple instances of firmware. It was designed to support basic microcontrollers consisting of a single firmware image only. Modifying the LwM2M Firmware Update Object definition to match multi-image firmware requirements introduces backwards compatibility problems.

To offer solutions for high-end IoT devices, for example those running embedded Linux, a dedicated Software Component Object was standardized. It specifically supports updates where many software packages need to be managed. The Software Component Object is not a good fit for the advanced microcontrollers for the following reasons:

- Create and Delete operations could not be supported, with each predefined instance modelling one of the firmware component.
- Each package would need to be permanently activated, with Activate, Deactivate and Uninstall operations either being no-ops or always returning errors.
- Features like dependency handling, error reporting, firmware encryption, etc. would need to be added to the object definition to fulfill the requirements for advanced microcontrollers.

4.1. Market benefits

This WID aims to start work on an Advanced Firmware Update Object to support these emerging microcontrollers in a standardized way using the LwM2M protocol.

5. Main Use Case(s)

5.1. Single CPU with Partitioning between Secure Mode and Normal Mode

This configuration consists of a single CPU whereby this CPU supports a security partitioning scheme that allows memory and other system components to be divided into secure and normal mode. There will generally be two images: one for secure mode and one for normal mode. In this configuration, firmware upgrades will generally be done by the CPU in secure mode, which is able to write to both areas of the flash device. In addition, there are requirements to be able to update either image independently as well as to update them together atomically, as specified in the associated manifests.

5.2. Dual CPU, Shared Memory

This configuration has two or more heterogeneous CPUs in a single SoC that share memory (flash and RAM). Generally, there will be a mechanism to prevent one CPU from unintentionally accessing memory currently allocated to the other. Upgrades in this case will typically be done by one of the CPUs and is similar to the single CPU with secure mode.

5.3. Dual CPU, Other Bus

This configuration has two or more heterogeneous CPUs, each having their own memory. There will be a communication channel between them, but it will be used as a peripheral, not via shared memory. In this case, each CPU will have to be responsible for its own firmware upgrade. It is likely that one of the CPUs will be considered the primary CPU and will direct the other CPU to do the upgrade. This configuration is commonly used to offload specific work to other CPUs. Firmware dependencies are similar to the other solutions above: sometimes allowing only one image to be upgraded, other times requiring several to be upgraded atomically. Because the updates are happening on multiple CPUs, upgrading the two images atomically is challenging.

6. Requirements and Assumptions

The following requirements need to be fulfilled:

- Support the use cases listed above.
- Allow for updates of individual partitions.
- Support for complete updates (i.e. updates to all partitions).
- The group will develop a new object (with a separate specification). There is no expectation that a constrained client would support both, the LwM2M Firmware Update Object and this new Advanced Firmware Update Object.

6.1. Update Requirements

This enabler MUST provide a mechanism to indicate the dependencies of a firmware package on other firmwares.

This enabler MUST provide a mechanism to indicate where to store a firmware package.

This enabler MUST support conditions to respect before updating a firmware.

This enabler SHOULD provide a mechanism to indicate custom installation steps.

This enabler SHOULD provide a mechanism to indicate custom update process steps.

6.2. LwM2M Server Capabilities Requirements

This enabler MUST give the LwM2M Server the ability to identify the updatable firmwares on the device, eg: main MCU, bootloader, and modem firmwares.

This enabler MUST give the LwM2M Server the ability to gather information on the firmwares active on the device (applicable component, version, etc.).

This enabler MUST give the LwM2M Server the ability to monitor the update process.

This enabler MUST give the LwM2M Server the ability to control the update process.

6.3. Security Requirements

Security of the firmware update process is critical. The LwM2M protocol already secures the exchanges between the LwM2M Server and the device. Furthermore, the LwM2M protocol provides an access control mechanism. However, this is not sufficient and end-to-end security is necessary between the firmware package producer and the target device.

This enabler MUST provide authentication of the received firmware package.

This enabler MUST provide integrity protection of the received firmware package.

This enabler SHOULD provide confidentiality of the received firmware package.

This enabler MUST provide a rollback protection, preventing an attacker from installing a previous version of a firmware (which may contain known vulnerabilities).

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
OMA-WID-LwM2M_advanced_firmware_update-V1_0-20220614-A	14 Jun 2022	Status changed to Approved by DMSE WG

Table: A.1-1 Approved Version History