International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.1368
(01/2021)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Secure applications and services (2) – Internet of things (IoT) security

Secure firmware or software update for Internet of things devices

Recommendation ITU-T X.1368

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	X.1–X.199
OPEN SYSTEMS INTERCONNECTION	X.200–X.299
INTERWORKING BETWEEN NETWORKS	X.300–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600–X.699
OSI MANAGEMENT	X.700–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	X.850–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	
General security aspects	X.1000–X.1029
Network security	X.1030–X.1049
Security management	X.1050–X.1069
Telebiometrics	X.1080–X.1099
SECURE APPLICATIONS AND SERVICES (1)	
Multicast security	X.1100–X.1109
Home network security	X.1110–X.1119
Mobile security	X.1120–X.1139
Web security	X.1140–X.1149
Security protocols (1)	X.1150–X.1159
Peer-to-peer security	X.1160–X.1169
Networked ID security	X.1170–X.1179
IPTV security	X.1180–X.1199
CYBERSPACE SECURITY	
Cybersecurity	X.1200–X.1229
Countering spam	X.1230–X.1249
Identity management	X.1250–X.1279
SECURE APPLICATIONS AND SERVICES (2)	
Emergency communications	X.1300–X.1309
Ubiquitous sensor network security	X.1310–X.1319
Smart grid security	X.1330–X.1339
Certified mail	X.1340–X.1349
Internet of things (IoT) security	**X.1360–X.1369**
Intelligent transportation system (ITS) security	X.1370–X.1389
Distributed ledger technology security	X.1400–X.1429
Distributed ledger technology security	X.1430–X.1449
Security protocols (2)	X.1450–X.1459
CYBERSECURITY INFORMATION EXCHANGE	
Overview of cybersecurity	X.1500–X.1519
Vulnerability/state exchange	X.1520–X.1539
Event/incident/heuristics exchange	X.1540–X.1549
Exchange of policies	X.1550–X.1559
Heuristics and information request	X.1560–X.1569
Identification and discovery	X.1570–X.1579
Assured exchange	X.1580–X.1589
CLOUD COMPUTING SECURITY	
Overview of cloud computing security	X.1600–X.1601
Cloud computing security design	X.1602–X.1639
Cloud computing security best practices and guidelines	X.1640–X.1659
Cloud computing security implementation	X.1660–X.1679
Other cloud computing security	X.1680–X.1699
QUANTUM COMMUNICATION	
Terminologies	X.1700–X.1701
Quantum random number generator	X.1702–X.1709
Framework of QKDN security	X.1710–X.1711
Security design for QKDN	X.1712–X.1719
Security techniques for QKDN	X.1720–X.1729
DATA SECURITY	
Big Data Security	X.1750–X.1759
5G SECURITY	X.1800–X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1368

Secure firmware or software update for Internet of things devices

Summary

Recommendation ITU-T X.1368 specifies: 1) basic models and procedures for securely updating firmware or software (FW/SW) of Internet of things (IoT) devices; and 2) requirements and capabilities for updating IoT FW.

A common secure update procedure is specified with general requirements. This procedure allows common IoT SW/FW updates to be securely implemented among stakeholders in the IoT environment, such as IoT device developers and IoT system/service providers.

This Recommendation focuses on updating FW, but it is applicable to updating any other SW of IoT devices.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.1368	2021-01-07	17	11.1002/1000/14445

Keywords

IoT, security, software update.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

© ITU 2021

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

Introduction

Cyberattacks against Internet of things (IoT) devices or systems are becoming increasingly sophisticated, intelligent and varied. Previously, functions of most IoT devices were fixed by IoT device vendors in their initial release phase. However, recently, devices are connected to the Internet to provide an enhanced set of IoT services. Therefore, IoT devices in use are facing cyber threats or attacks. It needs to be recognized that firmware or software (FW/SW) implemented in IoT devices need to be securely updated to fix their vulnerabilities and weaknesses. Some device vendors have already started providing an FW update service by means of their own schemes.

This Recommendation provides basic models and procedures for secure updating of IoT SW/FW, as well as associated requirements and capabilities. With the basic models and common update procedure, IoT SW/FW can be securely exchanged among stakeholders in the IoT environment with encouragement for stakeholders to update outdated IoT SW/FW.

Recommendation ITU-T X.1368

Secure firmware or software update for Internet of things devices

1 Scope

This Recommendation specifies basic models and procedures for securely updating the firmware or software (FW/SW) of IoT devices. It also describes requirements and capabilities for IoT FW/SW updates. This Recommendation focuses on updating FW, but it is applicable to updating any other SW of IoT devices.

2 References

None.

3 Definitions

3.1 Terms defined elsewhere

None.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 author: An entity that produces firmware (FW) images and SW for Internet of things devices. Examples include an individual or a group, such as a company or any other type of organization. It may upload the image to one or more FW servers that are not necessarily trusted.

3.2.2 firmware consumer: An entity that stores, verifies, and runs firmware (FW) images on an Internet of things (IoT) device. It should decide whether to run the current FW images. An IoT device has one or more FW consumers.

3.2.3 firmware server: An entity that distributes firmware (FW) images. It could accept FW images from multiple author; it could be a repository for a particular vendor or a repository that accepts various vendors. Ideally, an FW server is trustful, but it could be untrusted; it could try to view or modify the FW packages received from authors.

3.2.4 manifest: A record that contains metadata of a firmware image.

3.2.5 status tracker: An entity that checks and keeps tabs on the status of the firmware (FW) images inside one or more FW consumers and initiates the FW updates as needed. This includes fine-grained monitoring of changes at the device, e.g., the version of the running FW images and the state of the FW update cycle the device is currently in. A status tracker may reside inside an Internet of things device, on the Intranet, or on the Internet.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

DLT	Distributed Ledger Technology
FW	Firmware
IoT	Internet of Things
SW	Software
URL	Uniform Resource Locator

5 Conventions

This Recommendation follows the following conventions, which are aligned with [b-IETF RFC 2119].

Shall This word means that the definition is an absolute requirement of the specification.

Shall not This phrase means that the definition is an absolute prohibition of the specification.

Should This word means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

Should not This phrase means that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label.

May This word means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item.

6 Basic model

The networking architecture of IoT devices can differ, but four functional entities should be used in all the cases, i.e., FW consumer (see clause 3.2.2), status tracker (see clause 3.2.5), author (see clause 3.2.1) and FW server (see clause 3.2.3). Note that multiple functional entities can reside inside one node. For instance, a webcam device contains multiple FW consumers and a status tracker, while a web server contains a status tracker and an FW server. Multiple FW consumers can reside inside one network and are monitored by a status tracker implemented inside the gateway. Depending on the degree of constraints on the IoT devices, such designs may differ. Typical deployment models are described in clause 8.

In the basic model, these entities play indispensable roles in updating IoT SW/FW. The basic scenario in this model is simple: *a status tracker that recognizes the need for an IoT SW/FW update initiates the SW/FW update procedure that allows FW consumers to receive an SW/FW image from an author through an FW server*.

7 Update procedures

Figure 1 depicts the basic procedure for updating the FW. Prior to initiating the FW update procedure, an author needs to upload a new FW image to an FW server. It is desirable that the image be accompanied with a digital signature and is encrypted by the author.

When a status tracker receives the request to update an FW image with its location [e.g., its uniform resource locator (URL)], it verifies the request and then, if the request is valid, checks the status of the FW by communicating with the FW consumer in order to confirm the need for an FW update. Some typical ways of sending such requests are listed in clause 9.

If the need to update the FW is verified, the FW consumer initiates an FW update by communicating the location of the available FW. The FW consumer then requests the updated FW image from the FW server. The FW server provides the FW image to the FW consumer provided that the FW consumer has the legitimate right to receive the update. Otherwise, the server sends an update message with an error code.

On receiving the update message, the FW consumer verifies the image. If no error is found, the FW consumer installs the FW and sends status information to the status tracker. Note that the cardinalities of the above four functional entities are many to many, i.e., multiple status trackers may communicate with multiple FW consumers, which may communicate with multiple FW servers, which may communicate with multiple authors.

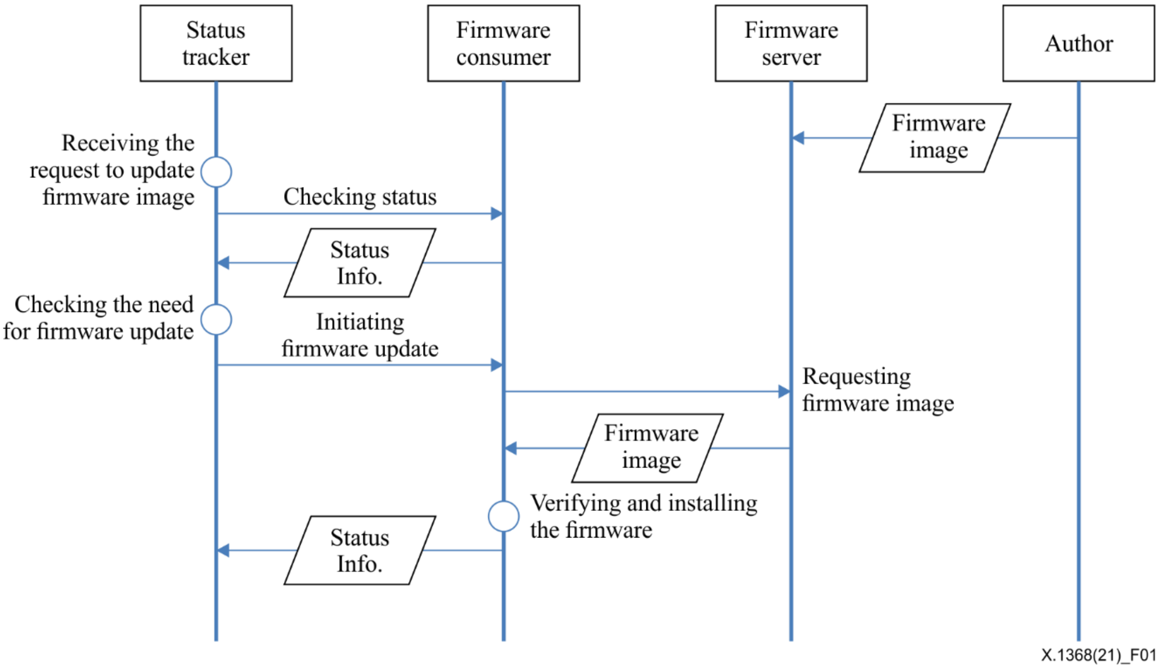
Status tracker
Firmware consumer
Firmware server
Author
Firmware image
Receiving the request to update firmware image
Checking status
Status Info.
Checking the need for firmware update
Initiating firmware update
Requesting firmware image
Firmware image
Verifying and installing the firmware
Status Info.
X.1368(21)_F01

Figure 1 – Protocol procedure

8 Deployment scenarios

As mentioned in clause 6, multiple functional entities can reside inside one node, and multiple entities can serve as a functional entity; the deployment scenarios may differ, depending on cases. In this clause, several deployment scenarios are illustrated.

8.1 Functional entities inside IoT devices

Figure 2 shows four types of IoT devices. An IoT device shall contain at least one FW consumer because it is natural that an IoT device contains multiple FW images.

An IoT device shall contain at least one status tracker. It could contain multiple status trackers to handle multiple FW consumers, but having a single status tracker that handles all of the FW consumers also works fine.

A resource-constrained IoT device may wish to minimize the functionality of the status tracker. In this case, the functionality of status tracker is divided into client side module and server side module deployed outside the IoT device. Minimal functionality, e.g., interactions with an FW consumer, are left on the client side, while the other functionalities are exported to the server side. A server side module may take care of multiple client side modules. This is often the preferred type of deployment.

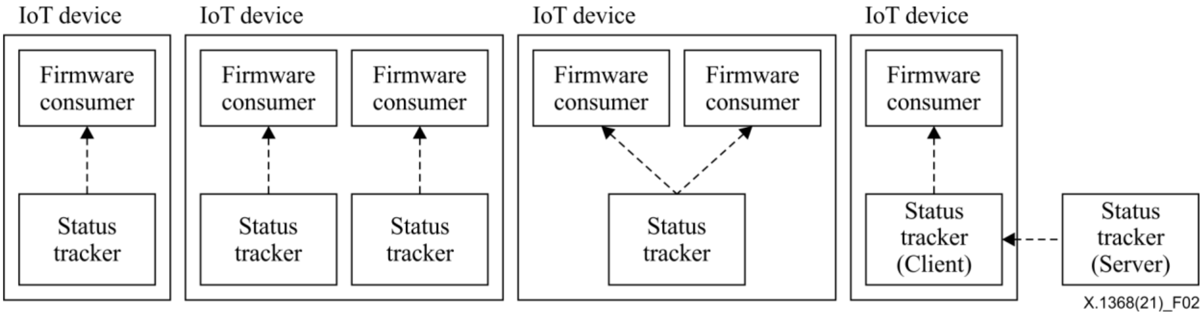

Figure 2 – Different types of IoT device

8.2 Status tracker deployment types

The resources of IoT devices differ a lot; a status tracker may reside inside an IoT device, but resource-constrained IoT devices may wish to keep a tracker separate and minimize its resource consumption. Moreover, there could be cases where it is preferable for IoT devices to be managed by a centralized entity for convenience.

To cope with these situations, a status tracker may be divided into several modules and implemented in a hierarchical manner. In this case, multiple modules can be cascaded so that an upstream module may judge the need for an FW update and initiate the procedure for it through the downstream modules.

Cases where: (1) a status tracker inside an IoT device directly communicates with an FW server; (2) a status tracker inside an IoT device communicates with an FW server via another status tracker residing inside the Intranet; and (3) a status tracker inside an IoT device communicates with an FW server via multiple status trackers are discussed in clauses 8.2.1 to 8.2.3.

8.2.1 In-device status tracker model

Figure 3 shows the in-device status tracker model. In this model, an FW consumer and a status tracker reside inside an IoT device. When the status tracker realizes the need for the FW update (see clause 9), it asks the FW consumer to receive FW images from the FW server. Note that the communication channels are abstracted in this section, but they could be connected via the Internet or other protocols, or a combination of them that requires bridges for communication among the entities. Figure 3 shows the Internet, but the techniques mentioned in this Recommendation can also cope with all sorts of other networks.

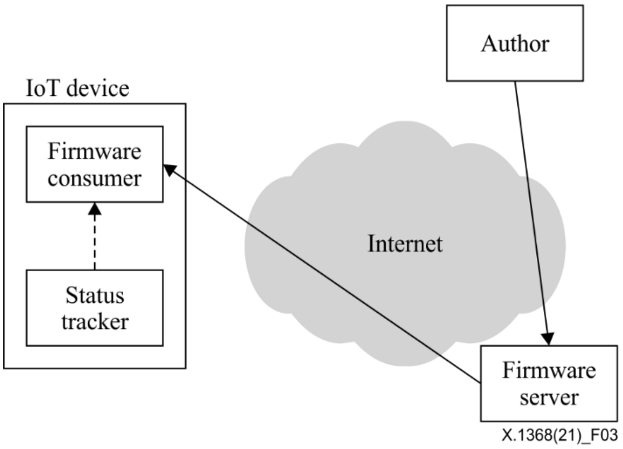

Figure 3 – Illustrative deployment of in-device status tracker model

8.2.2 Client-server status tracker model

Figure 4 shows the client-server status tracker model. In this model, a status tracker is divided into client module and server module. The client modules reside within IoT devices, while a server module resides inside a network. The server module monitors several IoT devices by communicating with the client modules. The client modules simply verify the message from the server module and act accordingly. The server module initiates the FW update procedure.

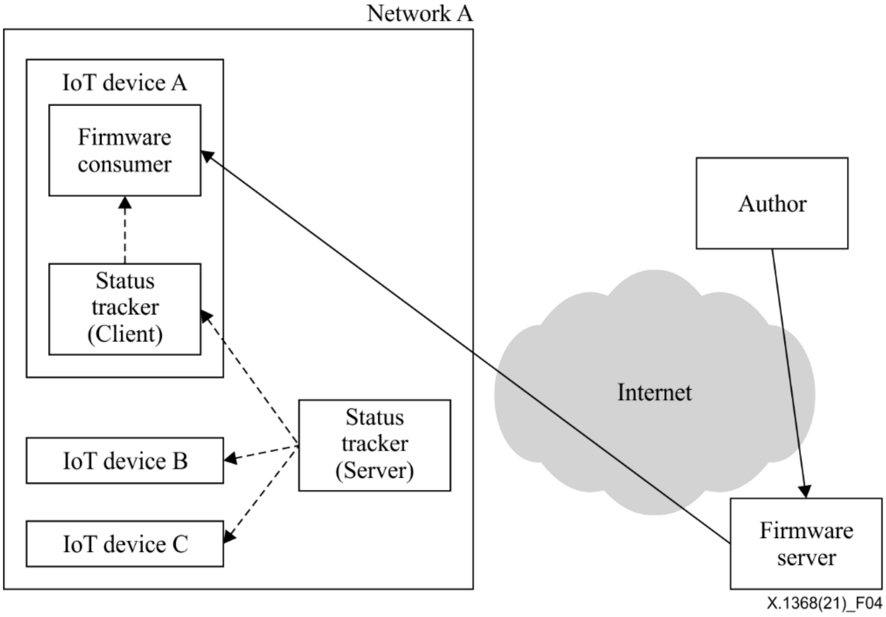

Figure 4 – Illustrative deployment of client-server status tracker model

8.2.3 Hierarchical status tracker model

Figure 5 shows the hierarchical status tracker model. In this model, a status tracker is divided into several modules: client module; intermediate modules; and server module. The client modules reside within IoT devices, the server module and intermediate modules reside inside networks. The intermediate modules monitor several IoT devices and by communicating with the client modules, the server module monitors all the client modules by communicating with intermediate modules. Note that intermediate modules could be cascaded to generate further hierarchy. The client modules simply verify the message from the server module and act accordingly. The server module initiates the FW update procedure. The server module initiates the FW update procedure.

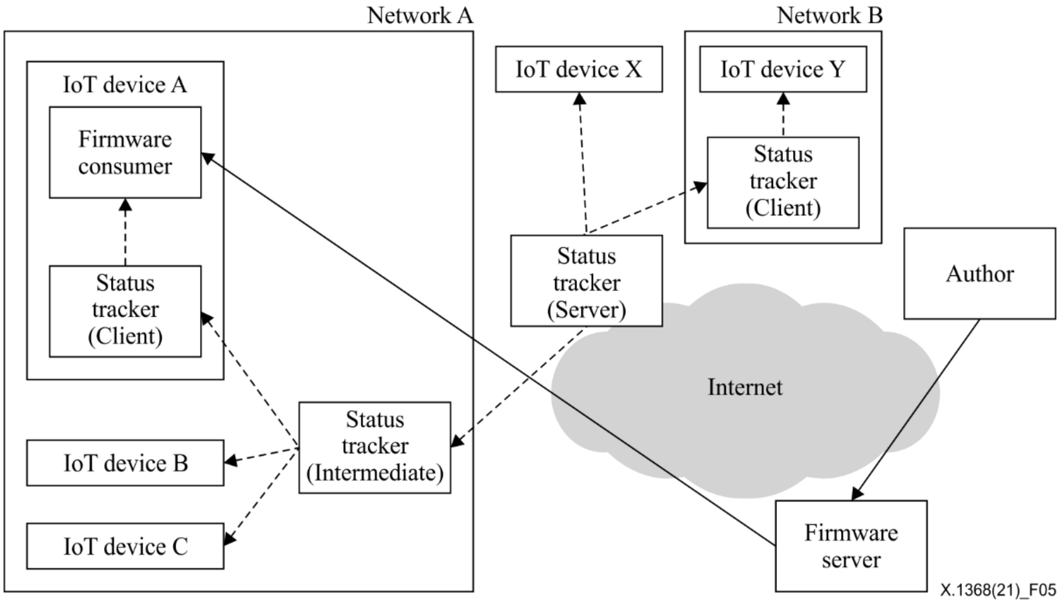

Figure 5 – Illustrative deployment of hierarchical status tracker model

9 Discovery of available new firmware images and initiation of the procedure

The whole process is initiated by a status tracker when it receives a request to update the FW image. This type of request can take various forms:

a) an author who publishes new versions of FW images may send the request;

b) an FW server who receives a new version of an FW image may send the request;

c) an IoT device administrator recognizes the release of a new version of the FW images and may send the request;

d) a status tracker or one of its upstream status trackers discovers the new version of the FW images by periodically polling the FW server;

e) a status tracker or one of its upstream status trackers recognizes the existence of a new version of the FW image by observing an FW update procedure of another IoT device that it takes care of.

Various other events may issue the requests, but those requests need to deliver information on the URL of the FW images and their versions to the status tracker. If the status tracker judges that the information is reliable and trustful, it can initiate the procedure mentioned in clause 7.

10 Requirements

This clause lists the functional requirements for IoT SW updates. Due to resource constraints, not all SW update procedures available in a non-constrained environment are applicable. It is often the case that no human user or operator is near the IoT devices. Therefore, when designing the concrete security update procedure, these differences need to be taken into account. Note that confidentiality, integrity and availability of the four functional entities shall be preserved, and these are prerequisites for the SW update; thus, they are omitted from the requirements in the following list.

a) Malicious SW/FW shall not be distributed:

   i) malicious images should be identified before being uploaded or exchanged;

   ii) integrity of FW images shall be verifiable;

   iii) the provider of FW images shall be verifiable.

b) Vulnerable SW/FW shall not be left without any proper measures:
   - i) obsolete versions of SW/FW should be detected;
   - ii) vulnerable SW/FW should be detected.

c) Failure caused during the update procedure shall be recoverable:
   - i) if an SW/FW update fails, there should be a means of notification of that situation;
   - ii) there should be a rollback means or protection means in the case of update process failure.

d) Only intended and necessary updating should be conducted:
   - i) only the newer versions of IoT SW/FW shall be installable;
   - ii) only trusted IoT SW/FW images shall be installable.

e) Resource constraints shall be considered:
   - i) an update procedure should not occur if there is no need for it to minimize network resources;
   - ii) status tracker functions may be cascaded to minimize the burden of resource-constrained IoT devices.

f) The intellectual property rights of authors shall be preserved:
   - i) the SW/FW images shall be encrypted by authors;
   - ii) the confidentiality, integrity and availability of IoT SW/FW images shall be preserved;
   - iii) the SW/FW images shall be securely transferred from the author to the final destination.

11 Capabilities

Based on the requirements mentioned in clause 10, this clause lists the capabilities of functional entities.

11.1 Capabilities of a firmware consumer

a) An FW consumer should be capable of:
   - i) verifying whether the previous run of an FW update was successful;
   - ii) sharing the information on its current SW/FW images (e.g., version number) with the parties who request this information with legitimate rights;
   - iii) implementing a rollback means or protection means in the case of update process failure;
   - iv) notifying the status tracker of the need for an FW update;
   - v) confirming the authenticity and integrity of FW images by verifying their certificates itself or by other means (e.g., by delegating the verification process to other entities);
   - vi) choosing not to install the new version of SW/FW images.

b) An FW consumer is recommended to have a "safe mode" that runs the IoT device with minimal functionality and that at least provides a means to manually install, restore or update FW.

11.2 Capabilities of a status tracker

A status tracker should be capable of:

a) maintaining lists of FW consumers, those with updated FW images and those with obsolete ones:
   - i) those lists should minimally contain their unique identifiers,
   - ii) a status tracker should be able to identify FW consumers with obsolete FW;

b) knowing the status of the FW consumers under its administration, there should be means:

   i) to confirm whether the FW consumer (and the IoT device that contains the FW consumer) is up by communicating with the FW consumer and by checking the communication logs with the FW consumer,

   ii) to confirm whether a previous run of an FW update at an FW consumer was successful,

   iii) to know the versions of FW that FW consumers run;

c) judging whether an SW/FW update is necessary and initiating an FW update procedure when needed;

d) verifying the authenticity of an upstream status tracker when implemented in a hierarchical manner.

When the functionality of a status tracker is divided into more than one module, the modules shall be capable of:

a) maintaining confidentiality and integrity of communication among them;

b) verifying the authenticity of the signals sent from one another;

c) maintaining their reachability information.

11.3 Capabilities of a firmware server

An FW server should be capable of:

a) accepting submissions of IoT SW/FW images from authors;

b) providing SW/FW images it contains to FW consumers;

c) identifying malicious SW/FW images and taking appropriate action, such as removing them from its internal storage and banning submission from authors who submit such images;

d) managing the list of authors and FW consumers that use it;

e) managing versions of IoT SW/FW images;

f) maintaining the list of FW consumers and the SW/FW images they downloaded in the past;

g) notifying IoT devices that have downloaded obsolete IoT SW/FW images in the past of the availability of new versions;

h) checking the geographical or logical location of IoT devices and to allow or deny downloading SW/FW images in order to avoid distributing them in forbidden locations identified by policies or other means.

11.4 Capabilities of an author

An author should be capable of maintaining the authenticity, confidentiality and integrity of the FW images it produced.

a) FW should not be replaced or compromised by third parties (authenticity and integrity).

b) The intellectual property of vendors within the FW should be preserved (confidentiality).

c) An author should implement measures to secure the FW image it uploads to FW servers because an FW server is not necessarily trusted.

Appendix I

Related activities outside ITU-T

(This appendix does not form an integral part of this Recommendation.)

Activities related to IoT SW updates that have been studied outside ITU-T include:

1) IOTSU workshop [b-ISOC IoTSU];

2) IETF SUIT working group [b-IETF suit]:
   - on manifest file [b-IETF manifest],
   - on FW update architecture [b-IETF architecture];

3) oneM2M: Standards for M2M and the IoT [b-oneM2M] etc.

Appendix II

An example scenario of IoT software update using distributed ledger technology

(This appendix does not form an integral part of this Recommendation.)

II.1 Overview

The IoT infrastructure contains numerous devices that an administrator needs to manage. An IoT device undergoes various revisions according to hardware features, and different FW may be applied according to the additional sensor board. In addition, depending on the hardware version, the supported SW version differs. Furthermore, differences in installed SW packages can cause dependency problems. This problem can be solved by using distributed ledger technology (DLT).

DLT has a smart contract that allows hardware FW or SW updates based on contracts pre-written by the administrator. Also, solutions to security vulnerabilities that can occur during the update process can be based on consensus algorithms and the cryptography layer.

This example describes how to provide secure FW/SW updates based on DLT in environments with different hardware revisions and SW versions, such as IoT infrastructure.

II.2 Software update procedure

See Table II.1 and Figures II.1 and II.2.

Table II.1 – Block structure for SW update

Block header	– Block size, version – Previous block header hash
Block data	– Merkle root – Name of provider, publication time, version number, hash code of the file, file link, filename, file size, support hardware, SW dependency

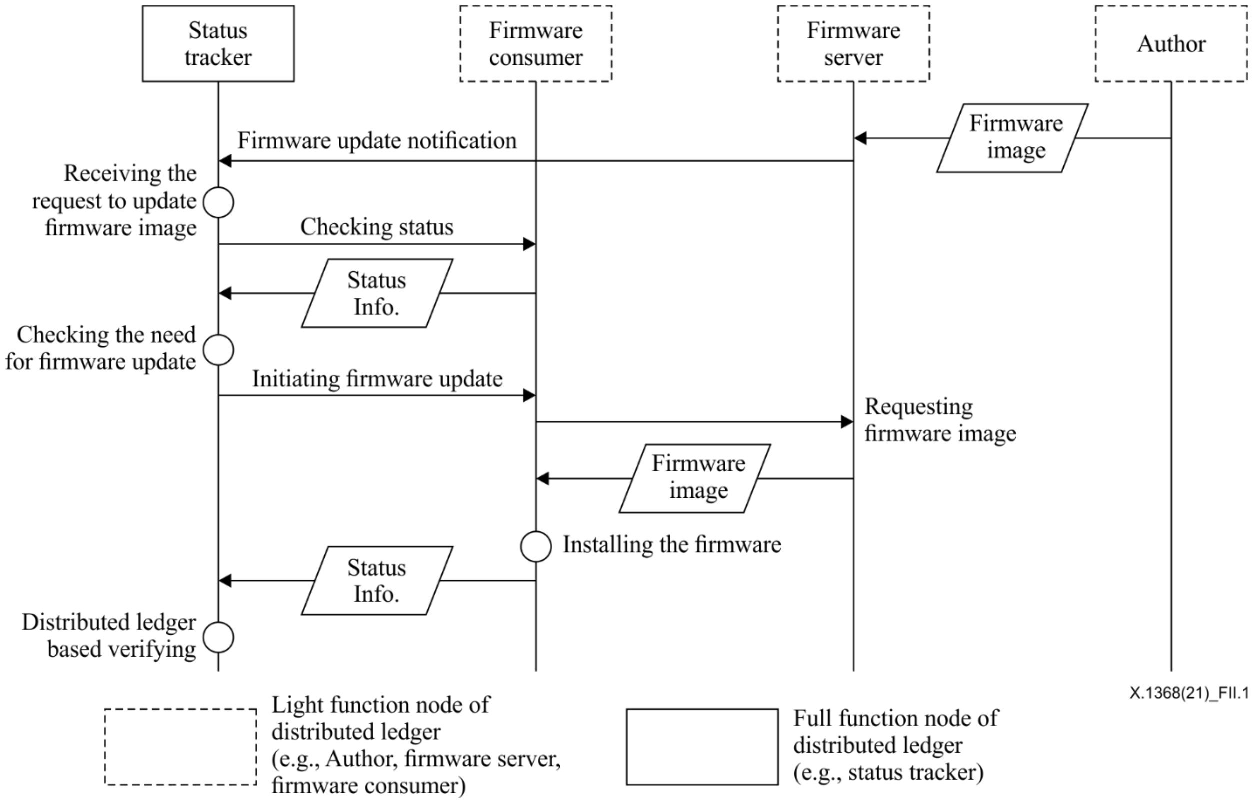
Status tracker
Firmware consumer
Firmware server
Author
Firmware image
Firmware update notification
Receiving the request to update firmware image
Checking status
Status Info.
Checking the need for firmware update
Initiating firmware update
Requesting firmware image
Firmware image
Installing the firmware
Status Info.
Distributed ledger based verifying
X.1368(21)_FII.1
Light function node of distributed ledger (e.g., Author, firmware server, firmware consumer)
Full function node of distributed ledger (e.g., status tracker)

Figure II.1 – Distributed ledger-based software update procedure

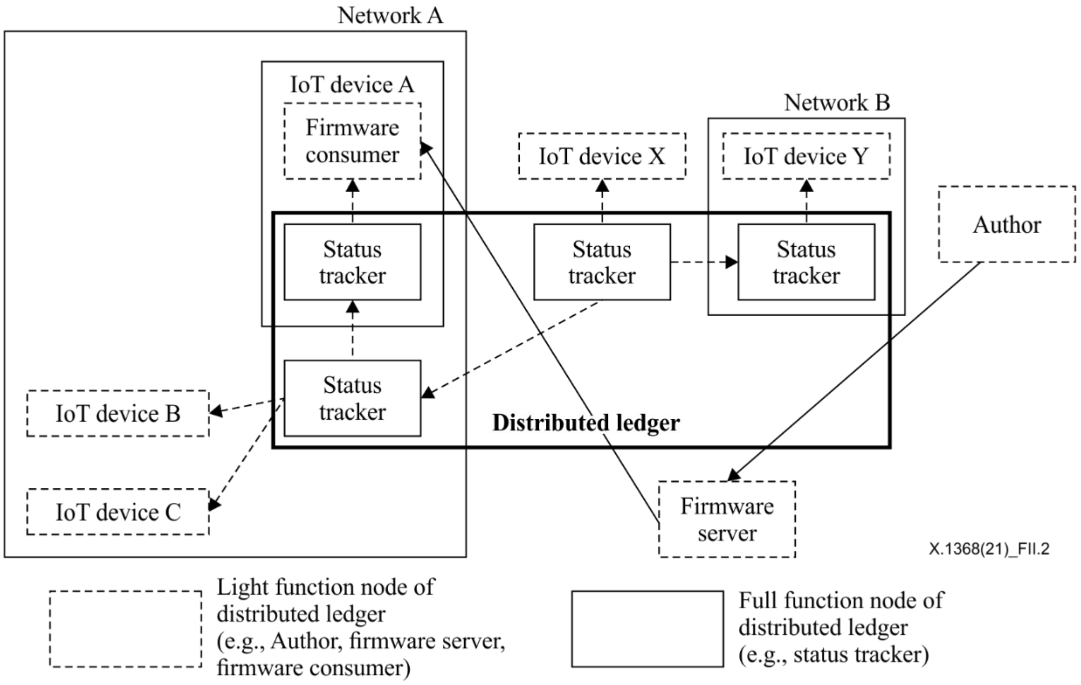

Figure II.2 – Distributed ledger-based software update for multiple networks

Bibliography

[b-IETF RFC 2119] IETF RFC 2119 (1997), *Key words for use in RFCs to indicate requirement levels.*

[b-IETF architecture] IETF SUIT (2019). *A firmware update architecture for Internet of things*., Wilmington, DE: Internet Engineering Task Force. Available [viewed 2021-02-19] at: https://tools.ietf.org/html/draft-ietf-suit-architecture-08

[b-IETF manifest] Moran, B., Tschofenig, H., Birkholz, H. (2019). *Firmware updates for Internet of things devices – An information model for manifests.* Wilmington, DE: Internet Engineering Task Force. Available [viewed 2021-02-19] at: https://tools.ietf.org/id/draft-ietf-suit-information-model-02.html

[b-IETF suit] IETF (2021). *Software updates for the internet of things (suit)*, version 7.26.0. Wilmington, DE: Internet Engineering Task Force. Available [viewed 2021-02-19] at: https://datatracker.ietf.org/wg/suit/about/

[b-ISOC iotsu] Internet Architecture Board (Internet), *Internet of things software update workshop (IoTSU) 2016*. Reston, VA: Internet Society. Available [viewed 2021-02-20] at: https://www.iab.org/activities/workshops/iotsu/

[b-oneM2M] oneM2M (2017), *Standards for M2M and the Internet of things*. oneM2M. Available [viewed 2021-02-20] at: http://www.onem2m.org/technical/published-drafts

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2021