iSCA

SMART COMMUNICATIONS ALLIANCE

IoT Secure Communications Module Protection Profile (IoT-SCM-PP)

Version 1.0.0, 2019-12-19

developed by
Secure Communications Alliance (SCA),
IoT PP working group:
Shanghai AOH Smart Technology Co., Ltd.
ChengDu JAVEE Microelectronics Co., Ltd.
ESIM Technology Co., Ltd.
FEITIAN Technologies Co., Ltd.
Haier Uplus Intelligent Technology (Beijing) Co., Ltd.
Infineon Technologies AG Co., Ltd.
NXP Semiconductors B.V.
STMicroelectronics
TechKnowledge Services Group Inc.
Wise Security Technology Co., Ltd.
WuHan TianYu Information Industry Co., Ltd.

Contents

1 PP introduction

The purpose of this Common Criteria (CC) Protection Profile (PP) is to standardize the security requirements of an IoT Secure Communications Module (IoT SCM) to be used in an IoT device. This PP targets IoT devices, which are home appliances like washing machines, refrigerators, air conditioners, etc.

The dedicated IoT SCM shall provide secure authentication of itself and of the user of the IoT host device the IoT SCM is integrated in, and confidentiality and data authentication of data exchanged between the IoT SCM, respective the IoT host device, and external entities. The main goal for the IoT SCM is to protect itself and its IoT host device against unauthorized use, and to prevent disclosure or undetectable modification of data exchanged with external entities. This document is intended to provide a detailed description of the requirements for the IoT SCM, the implementation of a concrete solution still be reserved for the IoT SCM developer. This PP also does not contain concepts how to use the IoT SCM in certain applications, i.e. the functional interface is not (yet) specified (this may to be done as a separate standardization step elsewhere).

Besides from the required functionality and some assumptions about its integration into the IoT host device this PP makes no restrictions about the form factor or internal architecture of the IoT SCM. Due to the fact that the IoT SCM shall implement communication down to the physical layer, the IoT SCM for sure has to consist of some dedicated hardware, likely containing additional software/firmware parts. As the name Secure Communications Module implies, a modular approach would be preferable, i.e. that a TOE compliant to this PP can be evaluated and certified once, and then integrated and used in different IoT host devices according to its certification without any modification.[1]

A TOE evaluated and certified according to this PP shall make use of an evaluated and certified IoT Secure Element (IoT SE) as specified by the separate Protection Profile IoT-SE-PP. One of the main goals of the IoT-SE-PP is to define requirements how an IoT SE protects all data, which are stored and processed internally, from unauthorized disclosure or modification. To do so, the IoT-SE-PP also defines requirements concerning physical protection and side-channel resistance of the IoT SE. The hardware of the IoT SE may be shared by the IoT SCM (and even the IoT application) for a higher level of integration, e.g. in terms of a system on chip (SoC).

[1] Less practical, but also a whole IoT host device with interwoven IoT SCM functionality could be TOE for this PP.

1.1 PP reference

Title:	IoT Secure Communications Module Protection Profile (IoT-SCM-PP)
Version:	1.0.0
Date:	2019-12-19
CC version used:	3.1 Revision 5
Assurance:	EAL2 augmented with ALC_FLR.1
PP registration:	BSI-CC-PP-0110, registered by the German Bundesamt für Sicherheit in der Informationstechnik (BSI)
PP authors:	Secure Communications Alliance (SCA), IoT PP working group: Shanghai AOH Smart Technology Co., Ltd. ChengDu JAVEE Microelectronics Co., Ltd. ESIM Technology Co., Ltd. FEITIAN Technologies Co., Ltd. Haier Uplus Intelligent Technology (Beijing) Co., Ltd. Infineon Technologies AG Co., Ltd. NXP Semiconductors B.V. STMicroelectronics TechKnowledge Services Group Inc. Wise Security Technology Co., Ltd. WuHan TianYu Information Industry Co., Ltd.

1.2 TOE overview

The TOE type addressed in this PP is a network device, which is intended to be integrated into an IoT host device. This network device is called IoT Secure Communications Module (shortly IoT SCM or just SCM) and is basically providing services – mainly secure channel functionality and information flow control – for the IoT application of its IoT host device. The IoT SCM relies on an IoT Secure Element (short IoT SE or just SE) certified based on [PP-SE], which is integrated in or connected to the IoT SCM. The IoT SCM makes sure that the IoT application cannot use inappropriate or disallowed connections, thus ensuring a minimum level of network security regardless what the IoT application is trying to do communication-wise.

Furthermore, as implementation of all layers of the OSI model is covered by the evaluation of the TOE, the confidence that an evaluated and certified IoT SCM will be resistant against network-based penetration attacks will be much higher than for a network device product, whose communication protocol implementations were not third-party reviewed and tested. Not being hacked is of course of high interest for the home user of an IoT device, but also important to prevent easy creation of bot nets or attacks on critical infrastructures like the electricity grid.

The following figure shows the context of the IoT SCM TOE. The IoT SCM is integrated in the IoT host device together with the IoT SE and the IoT application. The IoT SE (to be evaluated and certified according to IoT-SE-PP) is providing services to the IoT SCM, which is mediating, controlling and protecting any communication of the IoT device with network devices in a WAN (typically the internet), which provide services to the IoT device in the “IoT cloud”. The connection may be direct or mediated by an IoT gateway (which by the way could be an IoT device utilizing an IoT SE and an IoT SCM on its own). The IoT device user may interact with the IoT device indirectly by services provided in the IoT cloud (to control or monitor IoT SE and IoT SCM as far as the cloud-based functionality allows), but they also have a LAN-accessible interface to the IoT SCM, enabling them at least to read the SE ID, the firmware versions of IoT SE, IoT SCM and IoT application, and the network connection control rules currently stored in the IoT SCM. The IoT device user also may be a role known by the IoT application and therefore connect directly to the IoT device to control or monitor the IoT application, mediated by the IoT SCM within the connection limits it enforces (as configured by the IoT device admin).

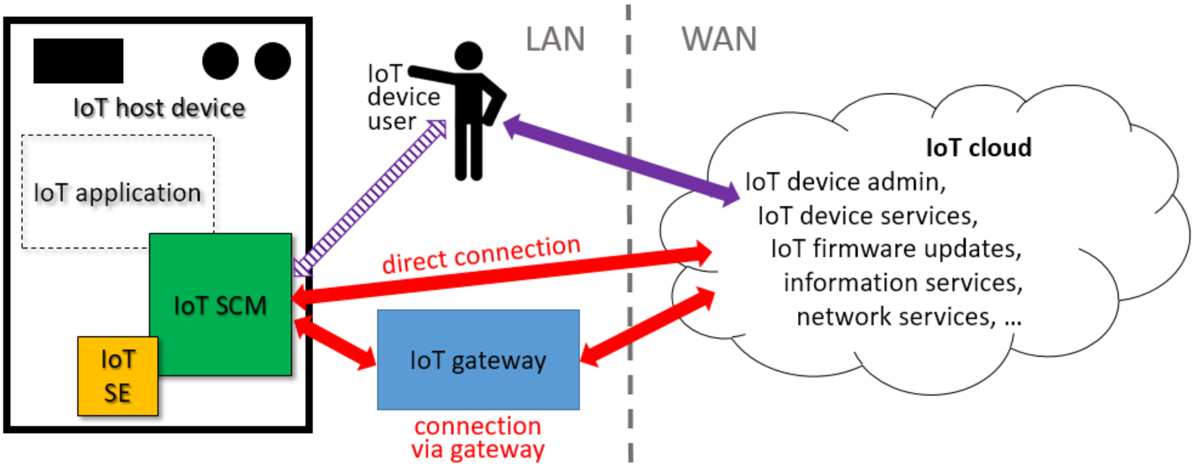


Figure 1: IoT SCM TOE in the greater IoT device context

Separation between LAN and WAN typically will be realized by a home router of the IoT device user. Though usually such a router will integrate a consumer-grade firewall to protect the LAN devices from unauthorized access from the WAN, the IoT SCM shall be able to defend itself against network attacks even if all of its network ports and services were exposed to the attacker (like there would be no firewall between LAN and WAN). This shall prevent compromise of the IoT SCM to overcome its security features or to use it in a botnet (for further attacks, e.g. on critical infrastructure network components). This prevention shall even hold if an attacker already was able to penetrate the LAN, e.g. by compromising another LAN device and using that one as a relay to attack the IoT SCM. To gain confidence about such attack resistance of the IoT SCM, AVA_VAN.2 component has been refined accordingly.

Physical scope of the TOE

The IoT SCM TOE shall consist of dedicated hardware containing software/firmware (hardware is needed as the hardware layer of the OSI model shall be covered by the TOE). The form factor of the TOE may be a single integrated circuit, a dedicated secure microcontroller in a system on chip (SoC), or any other multiple-chip solution that covers all communication layers from hardware layer to application (support) layer as indicated in Figure 3 and fulfils – among others – the requirements concerning physical protection, information leakage protection and fault-injection resistance.

The TOE, i.e. the IoT SCM, is intended to be integrated into an IoT host device including its IoT application, the latter being the IT hardware and firmware of the IoT host device finally making use of the IoT SCM for secure communication. The IoT SCM relies on an integrated or connected IoT Secure Element (IoT SE; to be evaluated and certified according to [IoT-SE-PP]) and therefore represents the operational environment of the IoT SE. Hence, the assumptions and objectives for the operational environment for the IoT SE need to be addressed in this PP and partly turn out to be specific requirements for the IoT SCM. Neither IoT SE nor IoT application belong to the IoT SCM TOE by definition, though it might be possible that the physical scopes of IoT SCM and IoT SE or even of IoT SCM and IoT SE and IoT application match or overlap (then both, this PP and the IoT-SE-PP would have to be applied to the product integrating IoT SCM and IoT SE, but likely in separated evaluations and certifications due to different assurance requirements in the two PPs).

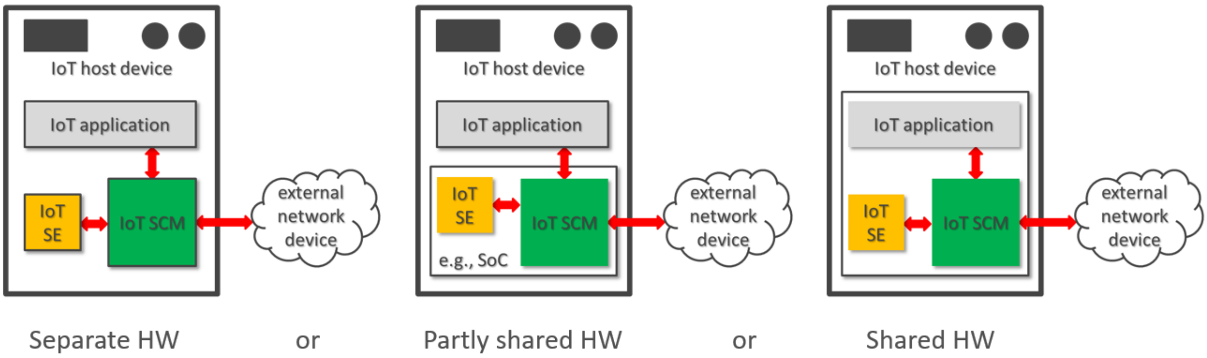


Figure 2: Examples for physical scope of the IoT SCM TOE inside the IoT host device

Depending on the concrete form factor of the IoT SCM TOE, dedicated evaluation and certification procedures may apply, as defined or adopted by the corresponding certification scheme. For instance, in SOG-IS evaluation and certification schemes, special evaluation requirements may apply according to supporting documents of the Joint Interpretation Library (JIL), e.g. if the TOE would be considered a single security IC or a hardware security box.

Logical scope of the TOE

To effectively secure communication with any network devices external of the IoT host device the TOE is built in, all layers of the OSI model shall be covered by the TOE. The application of the IoT host device can only communicate to external network devices via the IoT SCM TOE, whose logical scope is indicated by green colour in the following figure. Depending on the communication protocols used by the IoT SCM, security in terms of cryptography may be realised in various layers, all of which shall be covered by the evaluation, and all of which may receive cryptographic support and high-quality random numbers from the IoT SE (indicated in yellow in the following figure).

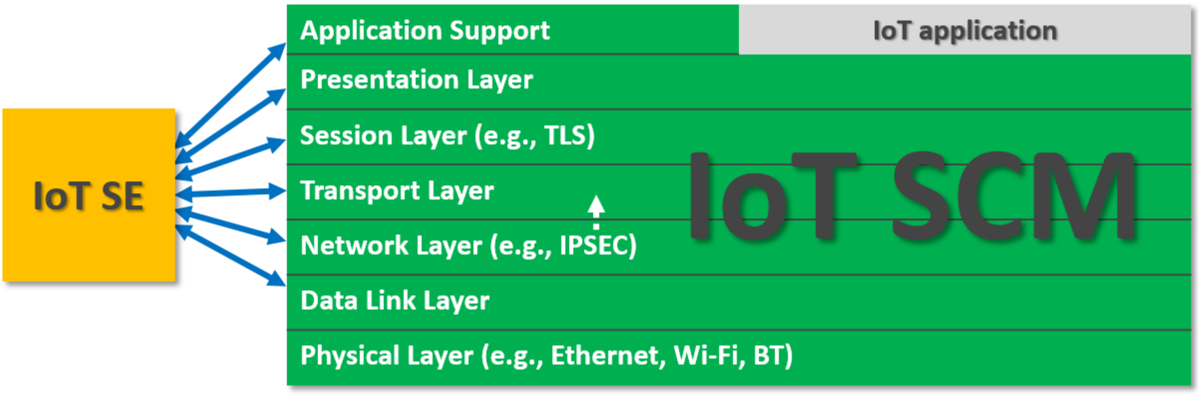


Figure 3: Coverage of all layers of the OSI model by the IoT SCM TOE

Functionality-wise, the logical scope of the IoT SCM TOE shall therefore cover

- secure channel functionality for communication to external network devices, regardless whether those are located in the same LAN as the IoT device or in some WAN (typically the internet);
- enforcement of strong cryptographic protection concerning authentication of channel endpoints, and confidentiality-protection and authenticity-protection of data transferred;
- allowing the IoT application to connect/communicate only to/with those network addresses, which have been configured by the remote IoT device admin and securely transferred into the TOE;
- blocking the IoT application from any non-allowed incoming connections, from non-authentic communication data and from communication data, which were not meeting the enforced minimum cryptographic protection level;
- providing cryptographic services and random numbers to the IoT application[2];
- secure update of the IoT SCM firmware; and furthermore
- enabling communication of the IoT device admin with the IoT SE and enabling firmware updates of the IoT SE (the latter only if firmware updates are supported by the IoT SE).

The main task of the IoT SCM TOE is to make sure that, once being configured properly by the IoT device admin, the IoT application cannot misconfigure communication in terms of choosing an insecure cipher suite, insufficient key size, wrong destination address, etc. Together with the fact that the implementation of all communication layers will be subject to testing and vulnerability analysis during the evaluation, the IoT SCM TOE shall be a secure and reliable communication platform for the IoT application of the IoT host device. This is of particular importance as the IoT application is not intended to be CC-evaluated in addition, because of the fact that there will be a very broad spectrum of different, device-specific IoT applications for each IoT SCM TOE).

Optional functionality

The IoT SCM TOE may also include other security functionality as a service for the IoT application. The ST author may add requirements for other security functionality of the TOE, as long as those are not in conflict with or can be used to deactivate or bypass the security

[2] At least a part, if not all of those cryptographic services are provided by the IoT SE and only mediated by the IoT SCM. In particular random number generation may solely be performed by the IoT SE. It still shall be possible for the ST author to add requirements for cryptographic operation (e.g., encryption/decryption using session keys) and random number generation (e.g., by a fast deterministic random number generator) to the ST.

functionality as required by this PP. E.g., though not required by this PP, the ST author may add requirements for functionality enabling the IoT device user to directly configure communication settings and/or rules, using an SCM interface accessible from the LAN without involvement of the IoT device admin. If such capabilities for the IoT device user would be added by the ST author, these shall not contradict or violate any security objectives of this PP. Furthermore, some sensitive settings shall not be possible for the IoT device user, as they could cut off communication with the IoT cloud unintendedly otherwise, e.g. by changing the necessary network connection rule in the TOE.

TOE life-cycle

The life-cycle of the IoT SCM TOE can be separated into the following phases:

1. Development of hardware and firmware of IoT SCM
2. Production of hardware and firmware IoT SCM
   (with optional integration of IoT SE into IoT SCM)
3. Delivery of completed IoT SCM to IoT device manufacturer.
4. Integration of IoT SCM (and IoT SE) into IoT host device
5. Delivery of IoT device to IoT device user
6. Normal operation by IoT device user and IoT admin

Phases 1 to 3 are within responsibility of the IoT SCM developer. It shall be ensured that these phases are performed by trusted personnel in secure environments. Since the realization of the phases depend on the concrete SCM, it is important that the TOE developer considers and enforces appropriate security measures during phases 1 to 3.
All relevant development, production and delivery sites used in phases 1 to 3 shall be subject to evaluation of assurance aspect ALC.

Phases 4 and 5 are already considered usage phases, which are within responsibility of the IoT device manufacturer. The IoT device manufacturer shall regard the assumptions as stated in section 3.2 hereinafter (as far as these assumptions are applicable, according to the concrete form factor of the IoT SCM and the way of integration into the IoT host device).

In phase 3, the certified IoT SCM TOE has to be complete and no more modification of the TOE configuration is allowed after that (other than updating its firmware with a newer version, which is also certified according to this PP on the same IoT SCM hardware).

2 Conformance claims

2.1 CC conformance claim

This PP uses the Common Criteria version 3.1 Revision 5.

This PP is conforming to Common Criteria Part 2 extended.

This PP is conforming to Common Criteria Part 3.

2.2 PP claim and package claim

This PP does not claim conformance to any other PP.[3]

This PP is conforming to assurance package EAL2 augmented by ALC_FLR.1 as defined in Common Criteria Part 3.

2.3 Conformance claim rationale

This PP does not claim conformance to any other PP.

Nevertheless, the PP is based on [IoT-SE-PP] as a certified IoT-SE has to be used for the IoT-SCM.

2.4 Conformance statement

This PP requires strict conformance of the ST or PP claiming conformance to this PP.

[3] IoT SCM relies on a certified IoT SE and certain requirements about the operational environment of the IoT SE need to be addressed in this PP, but this does not correspond to a PP conformance claim acc. to ASE_CCL.

3 Security problem definition

3.1 Terms and assets

Term	Description
IoT SE	IoT Secure Element, the component in the IoT host device that securely stores and processes persistent cryptographic keys.
IoT SCM (TOE of this PP)	IoT Secure Communication Module, the component in the IoT device that can actually connect to external network devices. Provides services of network connection control and secure channel functionality. The SCM uses the IoT SE to securely store and process persistent cryptographic keys.
IoT host device	A device like e.g. a home appliance that uses the functionality of IoT SE and IoT SCM integrated into that IoT host device.
IoT application	IT part of the IoT host device, which is using services of IoT SE and IoT SCM.
IoT device	Combination of IoT host device, IoT application, IoT SE and IoT SCM. An IoT device may be any kind of device that connects to a network (presumably a LAN connected to the internet) and that is able to send or receive information to or from the network or via the network to the internet. IoT devices may communicate with various entities like other IoT devices, IoT gateways and the IoT device admin.
External network device	Any network device external to the IoT device, which the IoT device establishes a network connection to, via its IoT SCM. May be in the same LAN as the IoT device (e.g., an IoT gateway) or in the WAN (e.g., a server in the IoT cloud).
IoT gateway	A device placed in the same LAN as the IoT device, mediating the connection of the IoT device (and supposedly of other IoT devices in the same LAN) to the IoT device admin or to external network devices in the IoT cloud.
IoT cloud	Sum of all external network devices (clients, servers, etc.) in the WAN, which the IoT device is connecting to, either directly or indirectly, to send data to or receive data from. The IoT device admin is administering the IoT device from the IoT cloud.
IoT device admin	The IoT device admin (administrator) is responsible for the management of the security services of the TOE and the corresponding key management.
IoT device user	The individual who is the actual user of an IoT device, typically its owner or leaseholder. Most of the interaction with the IoT device the IoT user is doing via the IoT cloud, in those cases the IoT device is not aware of the IoT user, but receiving corresponding requests from the IoT admin (on behalf of the IoT device user instead). Still, the IoT device user is a role the IoT SCM is aware of, as the IoT device user can read the version information of IoT SE and IoT SCM and configuration settings of the IoT SCM via a direct connection to the IoT device (i.e. not mediated by the IoT cloud). The ST writer may decide to allow the IoT device user to set the configuration, perform a firmware update etc.of the IoT application, IoT SCM and/or IoT SE via a direct connection, but only to the extent not violating the security objectives of this PP).
SE developer	Developer of the IoT SE. Can generate firmware update images for the IoT SE and is the only entity that has got the keys to encrypt and sign or MAC-protect those firmware update images, if any.
SCM developer	Developer of the IoT SCM. Can generate firmware update images for the IoT SCM and is the only entity that has got the keys to encrypt and sign or MAC-protect those firmware update images.

Table 1: Terms

Asset	Description	Protection needs
IoT device data	Any data sent from the IoT device to the IoT cloud / IoT device admin. IoT device data may originate from the IoT application, IoT SE or IoT SCM itself. Examples of IoT device data are general status data, current configuration data, consumption/billing information, etc. (the exact specification of those data cannot be given here since it depends on the concrete use case of the IoT device that utilizes the TOE). The TOE, with the help of the IoT SE, cryptographically protects authenticity and confidentiality of IoT device data before these are transmitted from the IoT device by the TOE.	Integrity/ authenticity, confidentiality
External device data	Any data received by the IoT device, originating from an external network device the IoT device has established a network connection to. External device data may directly originate from the external network device (e.g., a server in the IoT cloud), or they may originate from somewhere else and are just forwarded by the external network device (e.g., IoT admin data, which are received by the IoT device through an IoT gateway). The term external device data does not refer to specific kind of data, but shall simply express all data, which are received by the IoT device via an established network connection. The TOE, with the help of the IoT SE, cryptographically verifies authenticity of external device dat and decrypts external device data when these are received from the external network device.	Integrity/ authenticity, confidentiality
IoT admin data	Any data originating from the IoT admin, which are sent to the IoT device. Examples of IoT admin data are any kind of control data and new/updated configuration data for all parts of the IoT device, i.e. IoT application, IoT SCM or IoT SE (the exact specification of those data cannot be given here since it depends on the concrete use case of the IoT device that utilizes the TOE). The TOE, with the help of the IoT SE, cryptographically verifies authenticity and decrypts IoT admin data when these are received by the TOE.	Integrity/ authenticity, confidentiality
IoT session key (ISK)	Cryptographic session key, established in the IoT SE, using one or more IDKs belonging to the IoT SCM, during establishment of a trusted channel or trusted path. There may be multiple ISKs established by the IoT SE. ISKs can be output from the IoT SE, to be used in the IoT SCM. As disclosure of an ISK means only a relatively low risk for the IoT device use case, protection of the ISKs against local attacks shall be no concern of the IoT SCM (which may receive ISKs and store those in memory as plain text). Nevertheless ISKs shall be resistant against timing analysis, which also could be performed via a network commection to the IoT device.	Integrity/ authenticity, resistance against timing analysis
SCM FW	All firmware parts as stored in the TOE (making up the main part of the TSF).	Integrity, confidentiality

Asset	Description	Protection needs
SCM FW update image	An authenticity-protected and confidentiality-protected firmware update image that is imported into the TOE to update/replace in whole or part the current TOE firmware. Presented to the TOE during the firmware update process, and stored/activated inside the TOE if authenticity verification and decryption is successful.	Integrity/ authenticity, confidentiality
SCM FW update version	Attribute of the SCM FW update image specifying its version. Presented to the TOE during the firmware update process, and stored as latest SCM FW version in the TOE if the update is successful.	Integrity/ authenticity
Latest SCM FW version	Attribute of the last successfully installed firmware update, specifiying its version. TSF data, which is stored persistently in the TOE.	Integrity
SCM FW authentication key (SCM-FAK)	Public key or secret key used to verify the authenticity of a presented SCM FW update image, generated by the IoT SCM developer. SCM-FAK can be updated in the TOE (using the same authenticity-protection and confidentiality-protection mechanisms used for the SCM FW update image). If SCM-FAK is a secret key, it shall be device-individual for each copy of the TOE. SCM-FAK is stored as an IoT device key (IDK) in the IoT SE, compare IoT-SE-PP.	Integrity/ authenticity, if secret key also confidentiality
SCM FW confidentiality key (SCM-FCK)	Private key or secret key used to decrypt a presented SCM FW update image, generated by the IoT SCM developer. SCM-FCK has to be updateable in the TOE (using the same authenticity-protection and confidentiality-protection mechanisms used for the SCM FW update image). SCM-FCK is stored as an IoT device key (IDK) in the IoT SE, compare IoT-SE-PP.	Integrity/ authenticity, confidentiality
SCM-FAK signature/MAC	During firmware update: attribute of the SCM FW update image and its verson, in terms of a signature or MAC over both. Presented to the TOE during the firmware update process. Can only be generated by the IoT SCM developer, as only them shall know the necessary private key or as only them shall have the MAC key as stored in the TOE, respectively. During update of SCM-FAK and/or SCM-FCK: attribute of the value of the SCM-FAK and/or SCM-FCK to be updated, in terms of a signature or MAC over the value(s), which is verified by the TOE (using the currently stored SCM-FAK). SCM-FAK signature/MAC is presented to the TOE during the firmware key update process. SCM-FAK signature/MAC can only be generated by the IoT SCM developer, as only them shall know the necessary private key or as only them shall have the MAC key as stored in the TOE, respectively.	None (provides integrity/ authenticity protection itself)

Table 2: Assets

Though formally not being assets of the TOE, the reader may consult the list of assets of the IoT SE as stated in [IoT-SE-PP] in addition, for better understanding how the TOE uses the IoT SE and its services. For instance, as the IoT SCM is physically bound to an IoT SE, the identity of the IoT SCM TOE is realized by the asset “SE ID” as defined in [IoT-SE-PP].

3.2 Assumptions

A.SCM.Admin

It is assumed that the IoT device admin is trustworthy and well-trained to perform their duties. It is also assumed that the IoT device admin will configure the network connection control rules in the TOE in a way that only connections necessary for the operation of the TOE, the IoT SE and the IoT application can be established to external network devices.

A.SCM.Application

As the IoT SCM only provides a generic framework to perform secure communication, it is assumed that the IoT application makes sure that it uses the functionality of the IoT SCM consistently with its own security needs. This includes that the IoT application enables/uses cryptographic protection provided by the IoT SCM (and by the IoT SE via the IoT SCM) whenever sending or receiving confidential data and/or data to be authenticity-protected, and that the IoT application makes sure that data are sent to or accepted from the intended network entities/addresses only.

A.SCM.Integration

It is assumed that the IoT device manufacturer integrates the IoT SCM TOE into the IoT host device in a way that without significant physical modifications the IoT SCM TOE can only be used in connection with its intended IoT host device and IoT SE. Therefore, the TOE is physically bound to the IoT host device and IoT SE in a way that it is not easily possible to break that binding or physically inject data or commands between those parts of the IoT device. It is further assumed that the binding measure allows the IoT device manufacturer to detect if the binding has been physically tampered with (which could lead to loss of warranty or could be used as evidence in case of fraud).[4]

A.SCM.NoBypass

It is assumed that the IoT device manufacturer implements the IoT host device in a way that all communication with external network devices will be mediated via the IoT SCM TOE only, i.e. by construction, communication of the IoT application with external network devices is only possible if mediated by the TOE. This does not only mean that the IoT application does not make use of other ways to communicate to external network devices, the IoT application is unable to do so because it cannot access any hardware usable for network access (with the only exception of using the IoT SCM TOE for that purpose).[5]

A.SCM.FirmwareKeys

It is assumed that SCM-FAK and SCM-FCK are stored and used inside the IoT SE (as IoT device keys (IDKs), compare IoT-SE-PP).

If SCM-FAK is a public key (for verification of a signature), it is assumed that the IoT SCM developer generates a corresponding key pair randomly and keeps the corresponding private key confidentiality-protected in their development environment. It is further assumed that a public SCM-FAK is only shared for firmware updates for those IoT SCM products, which can

[4] Strengths of binding and tamper evidence have to be decided by the IoT device manufacturer, as they typically would be interested in that the binding between IoT SCM, its IoT host device and IoT SE cannot be easily broken.

[5] If SCM and IoT application use a shared hardware platform, effectiveness of this non-bypassability has to be examined during evaluation of the SCM TOE, otherwise A.SCM.NoBypass has to be restated as is in the SCM TOE's ST and has to be regarded when integrating IoT SCM and IoT application into the IoT device.

install/execute identical SCM FW Update Images; whereas for IoT SCM products which cannot, different product-specific public SCM-FAKs are used by the IoT SCM developer.

If SCM-FAK is a secret key (for verification of a MAC), it is assumed that the IoT SCM developer chooses it device-individual, either by random generation or by key derivation, and that the IoT SCM developer keeps SCM-FAK and its related key derivation key (if any) confidentiality-protected in their development environment. A key derivation key is only shared for deriving SCM-FAK for firmware updates for those IoT SCM products, which can install/execute identical SCM FW Update Images; for IoT SCM products which cannot, different product-specific key derivation keys for derivation of SCM-FAKs are used by the IoT SCM developer.

3.3 Threats

T.SCM.Impersonation

An attacker may try to send data to the IoT device the IoT SCM TOE is integrated in, impersonating a particular external network device, or to send data to an external network device, impersonating the TOE, without the respective receiving party being able to detect that.

The core of the attack is to trick the external network device into believing that data are sent from the TOE, or to trick the TOE into believing that data are sent from the IoT device admin. Thereby, the attack may require faking the TOE’s identity and/or keys stored in the TOE. Another aspect would be a man-in-the middle attack, in which an attacker could try to act between the TOE and external network device, presenting themselves as being the respective other party to TOE and the external network device.

The attacker does not necessarily need access to the TOE to perform the attack, but may find other ways. They may even be an IoT device user of the IoT device the TOE is integrated in, or IoT device user of another IoT device.

T.SCM.Modification

An attacker may try to intercept communication between the IoT device the IoT SCM TOE is integrated in and the external network device to modify or replay transmitted IoT device data or external device data, without the respective receiving party being able to detect that.

The attacker has access to data sent or received by the IoT device the TOE is integrated in by eavesdropping from a network and may modify, combine or replay those data in any way (maybe also using recorded communication data from a different IoT device).

The attacker may even be a rightful IoT device user of the IoT device the TOE is integrated in, or IoT device user of a different IoT device.

T.SCM.Disclosure

An attacker may try to intercept communication between the IoT device the IoT SCM TOE is integrated in and the external network device to gain knowledge about transmitted IoT device data or external device data.

The attacker has access to data sent or received by the IoT device the TOE is integrated in and retrieves confidential assets from that data.

T.SCM.IllegalConnection

A faulty or maliciously modified IoT application may try to establish a network connection to external network devices/addresses, which are not related to the operation of the IoT device, possibly ending up in confidential data being sent to the wrong entity in the network. Furthermore, a faulty or maliciously modified IoT application may try to establish a network connection to external network devices/addresses without establishing a secure communication channel, possibly ending up in confidential data being disclosed during transit or data being modified, substituted or replayed without the receiving party being able to detect that.

3.4 Organizational security policies

OSP.SCM.Auditability

The TOE shall provide functionality to output its SCM firmware version, the SE firmware version (as provided by the SE), and the network connection control rules currently configured in the SCM on request from external (this request may be non-authenticated if coming from inside of the LAN the SCM resides in).

OSP.SCM.SecureUpdate

The TOE shall provide functionality to securely update its firmware or parts thereof, protected concerning authenticity and confidentiality. Only authentic SCM firmware update images as provided by the developer of the TOE shall be accepted by the TOE. Non-authentic SCM firmware update images or those being issued by the TOE developer, but modified thereafter shall be rejected by the TOE. The TOE shall not accept a SCM firmware update image, if its firmware version is older than the version of the latest successfully installed firmware. The keys to protect the authenticity and confidentiality of the SCM firmware update image, i.e. SCM-FAK and SCM-FCK, respectively, shall be updateable, this update protected concerning authenticity and confidentiality the same way as the SCM firmware update image itself. The authenticity-protection mechanism and the confidentiality-protection mechanism used shall provide a cryptographic security level of at least 100 bit.

OSP.SCM.UtilizeSE

The TOE shall use and rely on an IoT SE, which is evaluated and certified according to Protection Profile IoT-SE-PP, for the following cryptographic operations:

- Generation of a signature or MAC over data to be sent to the IoT device admin as a proof of authenticity, using SE authentication key (SAK) stored in the IoT SE (see IoT-SE-PP);
- Verification of authenticity of data sent by the IoT device admin by verification of the corresponding signature or MAC coming with those data, using the admin authentication key (AAK) stored in the IoT SE (compare IoT-SE-PP);
- All cryptographic operations using static or ephemeral keys, which are used for establishing a trusted channel, from authentication of the end points to establishment of session keys, which are then used to counter T.SCM.Impersonation, T.SCM.Modification and T.SCM.Disclosure and to enforce OSP.SCM.SecureUpdate; the corresponding keys (e.g., SCM-FAK and SCM-FCK) shall be stored exclusively in the IoT SE;
- Random number generation.

The TOE may rely on the outputs of these operations of the IoT SE as it would have performed these itself, i.e. authenticity protection is not required between the IoT SE and the TOE.

OSP.SCM.LeakageProt

Countermeasures against disclosing cryptographic keys stored in the TOE by performing timing analysis on operations with those keys shall be employed by the TOE. The countermeasures shall be suitable to protect the cryptographic keys in the TOE also against the legitimate IoT device user of the IoT device the TOE is integrated in.

OSP.SCM.StrongCrypto

All cryptographic functions used by the security functionality of the TOE shall provide a cryptographic strength of at least 100 bit.[6]

[6] During certification of a specific SCM TOE, the certification body in charge may impose additional requirements concerning the choice and minimum strength of cryptographic functions.

4 Security objectives

4.1 Security objectives for the TOE

O.SCM.AuthProt

The TOE shall provide functionality of data authenticity protection by adding electronic signatures or message authentication codes (MACs) to data to be sent to an external network device, and by verification of electronic signatures or message authentication codes (MACs) of data received from an external network device. In case such verification fails, the corresponding potentially non-authentic or corrupted data shall not be output or used TOE-internally. The authenticity-protection mechanism(s) used shall also counter undetected modification, substitution, insertion and replay of data and provide a security level of at least 100 bit. The keys used for authenticity protection shall be session keys, which were established using functionality and keys stored in the IoT SE.

O.SCM.ConfProt

The TOE shall provide functionality of data confidentiality protection by encryption of data sent to an external network device, and by decryption of ciphertext data received from an external network device. The encryption mechanism(s) used shall provide security level of at least 100 bit. The keys used for confidentiality protection shall be session keys, which were established using functionality and keys stored in the IoT SE.

O.SCM.ConnectControl

The TOE shall provide functionality to allow the IoT device admin to specify/limit the network addresses the TOE is allowed to establish network connections to. The TOE shall provide functionality to allow the IoT device admin to specify per network address or network address range whether a secure channel (providing authenticity and confidentiality protection) shall be established when connecting to that address. If requested by IoT application, the TOE shall establish only those network connections, which are allowed according to destination address and secure channel requirement setting and which furthermore provide a cryptographic strength of at least 100 bit for both, data authentication as well as confidentiality protection mechanisms.

O.SCM.LeakageProt

Countermeasures against disclosing cryptographic keys stored in the TOE by performing timing analysis on operations with those keys shall be employed by the TOE. The countermeasures shall be suitable to protect the cryptographic keys in the TOE also against the legitimate IoT device user of the IoT device the TOE is integrated in.

O.SCM.Auditability

The TOE shall provide functionality to output its SCM firmware version, the SE firmware version (as provided by the SE), and the network connection control rules currently configured in the SCM on request from external (this request may be non-authenticated).

O.SCM.SecureUpdate

The TOE shall provide functionality to securely update its firmware or parts thereof, protected concerning authenticity and confidentiality. Only authentic SCM firmware update images as provided by the developer of the TOE shall be accepted by the TOE. Non-authentic SCM firmware update images or those being issued by the TOE developer, but modified thereafter shall be rejected by the TOE. The TOE shall not accept a SCM firmware update image, if its firmware version is older than the version of the latest successfully installed firmware. The keys to protect the authenticity and confidentiality of the SCM firmware update image, i.e. SCM-FAK and SCM-FCK, respectively, shall be updateable, this update protected concerning authenticity and confidentiality the same way as the SCM firmware update image itself. The authenticity-protection mechanism and the confidentiality-protection mechanism used shall provide a cryptographic security level of at least 100 bit.

O.SCM.UtilizeSE

The TOE shall use and rely on an IoT SE, which is evaluated and certified according to Protection Profile IoT-SE-PP, for the following cryptographic operations:

- Generation of a signature or MAC over data to be sent to the IoT device admin as a proof of authenticity, using the SE authentication key (SAK) stored in the IoT SE (compare IoT-SE-PP);
- Verification of authenticity of data sent by the IoT device admin by verification of the corresponding signature or MAC coming with those data, using the admin authentication key (AAK) stored in the IoT SE (compare IoT-SE-PP);
- All cryptographic operations using static or ephemeral keys, which are used for establishing a trusted channel, from authentication of the end points to establishment of session keys, which are then used in context of O.SCM.AuthProt and O.SCM.ConfProt;
- Random number generation.

The TOE shall rely on the outputs of these operations as it would have performed these itself, without further proof given by the IoT SE.

4.2 Security objectives for the operational environment

OE.SCM.Admin

The IoT device admin shall be trustworthy and well-trained to perform their duties. The IoT device admin shall configure the network connection control rules in the TOE in a way that only connections necessary for the operation of the TOE, the IoT SE and the IoT application can be established to external network devices.

OE.SCM.Application

As the IoT SCM only provides a generic framework to perform secure communication, the IoT application has to make sure that it uses the functionality of the IoT SCM consistently to its own security needs. This includes that the IoT application sends all data to the corresponding intended network entities/addresses only (that the right communication protection is applied is then enforced by the IoT SCM).

OE.SCM.Integration

The IoT device manufacturer shall integrate the IoT SCM TOE into the IoT host device in a way that without significant physical modifications the IoT SCM TOE can only be used in connection with its intended IoT host device and IoT SE. Therefore, the TOE shall be physically bound to the IoT host device and IoT SE in a way that it is not easily possible to break that binding or physically inject data or commands between those parts of the IoT device.

Furthermore, the binding measure shall allow the IoT device manufacturer to detect if the binding has been physically tampered with.[7]

OE.SCM.NoBypass

The IoT device manufacturer shall implement the IoT host device in a way that all communication with external network devices will be mediated via the IoT SCM TOE only, i.e. by construction, communication of the IoT application with external network devices shall only be possible if mediated by the TOE. This does not only mean that the IoT application shall not make use of other ways to communicate to external network devices, the IoT application shall be unable to do so because it shall not be able to access any hardware usable for network access (with the only exception of using the IoT SCM TOE for that purpose).[8]

OE.SCM.FirmwareKeys

If SCM-FAK is a public key (for verification of a signature), the IoT SCM developer shall generate a corresponding key pair randomly and keep the corresponding private key confidentiality-protected in their development environment. A public SCM-FAK may only be shared for firmware updates for those IoT SCM products, which can install/execute identical SCM FW Update Images; for IoT SCM products which cannot, different product-specific public SCM-FAKs shall be used by the IoT SCM developer.

If SCM-FAK is a secret key (for verification of a MAC), the IoT SCM developer shall choose it device-individually, either by randomly generating SCM-FAK per device or by deriving SCM-FAK per device, and keep SCM-FAK and its related key derivation key (if any) confidentiality-protected in their development environment. A key derivation key may only be shared for deriving SCM-FAK for firmware updates for those IoT SCM products, which can install/execute identical SCM FW Update Images; for IoT SCM products which cannot, different product-specific key derivation keys for derivation of SCM-FAKs shall be used by the IoT SCM developer.

[7] Strengths of binding and tamper evidence have to be decided by the IoT device manufacturer, as they typically would be interested in that the binding between IoT SCM, its IoT host device and IoT SE cannot be easily broken.

[8] If SCM and IoT application use a shared hardware platform, effectiveness of this non-bypassability has to be shown during evaluation of the SCM TOE, otherwise OE.SCM.NoBypass has to be restated in the TOE's ST.

4.3 Security objectives rationale

Security objectives Threats, OSPs and Assumptions from SPD	O.SCM.AuthProt	O.SCM.ConfProt	O.SCM.ConnectControl	O.SCM.Auditability	O.SCM.SecureUpdate	O.SCM.LeakageProt	O.SCM.UtilizeSE	OE.SCM.Admin	OE.SCM.Application	OE.SCM.Integration	OE.SCM.NoBypass	OE.SCM.FirmwareKeys
T.SCM.Impersonation	X											
T.SCM.Modification	X											
T.SCM.Disclosure		X										
T.SCM.IllegalConnection			X									
OSP.SCM.Auditability				X								
OSP.SCM.SecureUpdate					X							
OSP.SCM.LeakageProt						X						
OSP.SCM.UtilizeSE							X					
OSP.SCM.StrongCrypto	X	X	X		X							
A.SCM.Admin								X				
A.SCM.Application									X			
A.SCM.Integration										X		
A.SCM.NoBypass											X	
A.SCM.FirmwareKeys												X

Table 3: Coverage of SPD items by the security objectives

T.SCM.Impersonation is directly countered by **O.SCM.AuthProt**, which states that the TOE shall provide authenticity protection of data exchanged with the IoT device admin using an authenticity-protection mechanism.

T.SCM.Modification is directly countered by **O.SCM.AuthProt**, which states that the TOE shall provide authenticity protection of data exchanged with the IoT device admin using an authenticity-protection mechanism.

T.SCM.Disclosure is directly countered by **O.SCM.ConfProt**, which states that the TOE shall provide confidentiality protection of data exchanged with the IoT device admin by encryption.

T.SCM.IllegalConnection is directly countered by **O.SCM.ConnectControl**, which states that the TOE shall limit network connections to those allowed by the IoT device admin, in terms of network address and secure channel requirement.

OSP.SCM.Auditability is directly enforced by **O.SCM.Auditability** (objective re-states OSP).

OSP.SCM.SecureUpdate is directly enforced by **O.SCM.SecureUpdate** (objective re-states OSP)**.**

OSP.SCM.LeakageProt is directly enforced by **O.SCM.LeakageProt** (objective re-states OSP).

OSP.SCM.UtilizeSE is directly enforced by **O.SCM.UtilizeSE** (objective re-states OSP).

OSP.SCM.StrongCrypto is enforced by the combination of **O.SCM.AuthProt**, **O.SCM.ConfProt**, **O.SCM.ConnectControl** and **O.SCM.SecureUpdate**, which state that the corresponding cryptographic functions used shall have a security level of at least 100 bit.

A.SCM.Admin is directly upheld by **OE.SCM.Admin** (objective re-states assumption).

A.SCM.Application is directly upheld by **OE.SCM.Application** (objective re-states assumption).

A.SCM.Integration is directly upheld by **OE.SCM.Integration** (objective re-states assumption).

A.SCM.NoBypass is directly upheld by **OE.SCM.NoBypass** (objective re-states assumption).

A.SCM.FirmwareKeys is directly upheld by **OE.SCM.FirmwareKeys** (objective re-states assumption).

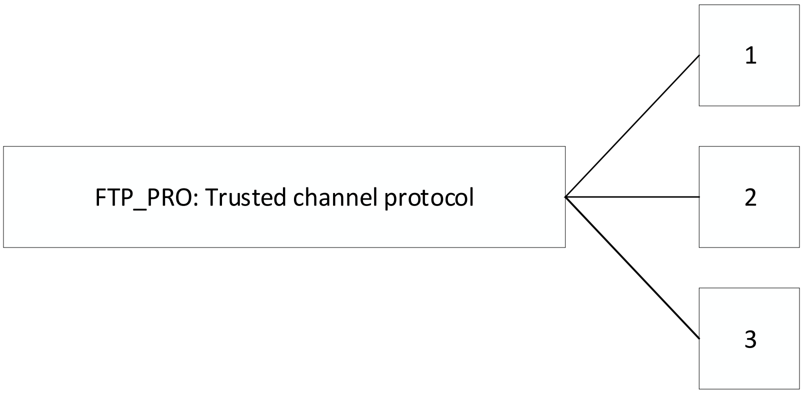
5 Extended component definition

5.1 Definition of the family trusted channel protocol (FTP_PRO)

Family behaviour

This family defines requirements for establishing a trusted channel and using the trusted channel to transfer the TSF data or user data securely.

Component levelling



FTP_PRO.1 Trusted channel protocol requires that communication be established in accordance with a defined protocol.

FTP_PRO.2 Trusted channel establishment requires that keys be securely established between the peers.

FTP_PRO.3 Trusted channel data protection requires that data in transit be protected.

Management of FTP_PRO.1

The following actions could be considered for the management functions in FMT:

a) Configuring the protocols needed for the trusted channel

b) Configuring the credentials for using the trusted channel

c) Configuring the conditions for initializing and terminating the trusted channel.

Management of FTP_PRO.2

The following actions could be considered for the management functions in FMT:

a) Configuring the parameters for shared secrets

b) Configuring the parameters for cryptographic key derivation.

Management of FTP_PRO.3

The following actions could be considered for the management functions in FMT:

a) Configuring the encryption and integrity mechanisms used by the trusted channel.

Audit of FTP_PRO.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: Failure of the trusted channel establishment

b) Minimal: Identification of the initiator and target of failed trusted channel establishment

c) Basic: All attempted uses of the trusted channel

d) Basic: Identification of the initiator and target of all trusted channel attempts.

Other events should be considered according to the specific protocols used.

Audit of FTP_PRO.2

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: Authentication failures during channel establishment

b) Basic: All authentication attempts.

Audit of FTP_PRO.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Minimal: Failures when attempting to verify channel properties in FTP_PRO.3.2.

FTP_PRO.1 Trusted channel protocol

Hierarchical to: No other components.

Dependencies: FTP_PRO.2 Trusted channel establishment

FTP_PRO.3 Trusted channel data protection.

FTP_PRO.1.1 **The TSF shall implement [assignment: *trusted channel protocol*] acting as [assignment: *defined protocol role(s)*] in accordance with: [assignment: *list of standards*].**

FTP_PRO.1.2 **The TSF shall enforce usage of the trusted channel for [assignment: *purpose(s) of the trusted channel*] in accordance with: [assignment: *list of standards*].**

FTP_PRO.1.3 **The TSF shall permit [selection: *itself, its peer*] to initiate communication via the trusted channel.**

FTP_PRO.1.4 **The TSF shall enforce the following rules for the trusted channel: [assignment: *rules governing operation and use of the trusted channel and/or its protocol*].**

FTP_PRO.1.5 **The TSF shall enforce the following static protocol options: [assignment: *list of options and references to standards in which each is defined*].**

FTP_PRO.1.6 **The TSF shall negotiate one of the following protocol configurations with its peer: [assignment: *list of configurations and reference to standards in which each is defined*].**

User application notes

FTP_PRO.1 may be iterated by the PP/ST author for different protocols, but also for different protocol roles of the same protocol, if completion of FTP_PRO.1 operations needs to be different for each protocol role.

Where values used in the completion of FTP_PRO.1 operations have dependencies between different FTP_PRO.1 elements, these need to be made clear in the instantiation of FTP_PRO.1. For example, a table could be given in which the columns represent the relevant selections and assignments, and the rows define the valid combination of completion values.

Operations

Assignment:

In FTP_PRO.1, examples of “defined protocol roles” would be ‘client’ or ‘server’ (e.g. in case of TLS protocol), ‘initiator’ or ‘responder’ (e.g., in case of IKEv2/IPsec protocol), ‘Trust Center’ (e.g., in case of ZigBee protocol) or ‘Key Distribution Centre’ (e.g., in case of Kerberos protocol).

In the first assignment in FTP_PRO.1.5, the PP/ST author should state rules for when the secure channel is required to be used by the TOE, such as mandating its use for communications with an audit server. If no specific uses of the channel are mandated for the TOE, this assignment can be completed with “none specified” (in this case, also the second assignment shall be completed with “none specified”).

In FTP_PRO.1.5, the PP/ST author should state rules related to implementation of the protocol (e.g., rules on maximum packet sizes or rekeying intervals). If there are no rules required, or if the standards referenced in other elements of FTP_PRO.1 include the relevant rules and no specific evaluator check is required for the context in which FTP_PRO.1 is being used, this assignment can be completed with “none specified”.

In FTP_PRO.1.6, the PP/ST author should state rules related to negotiable aspects of the protocol, when intending to narrow the options provided by the TOE compared to the standard that defines the protocol (e.g., selection of cipher suites or acceptance of older protocol versions). If no rules are required, this assignment can be completed with “none specified”. Where the assignment is completed with a list then that list specifies the only configurations permitted – any other configuration would be a violation of the SFR. FTP_PRO.1.6 may be used to specify mandatory supported configurations without limiting the TOE to using these configurations by, for example, listing the required configurations with “(support required)” after each entry in the list and then including a final element which states that any other configuration permitted by the standard is allowed.

FTP_PRO.2 Trusted channel establishment

Hierarchical to: No other components.

Dependencies: FTP_PRO.1 Trusted channel protocol

[FCS_CKM.1 Cryptographic key generation, or
FCS_CKM.2 Cryptographic key distribution]

FCS_CKM.5 Cryptographic key derivation

FCS_COP.1 Cryptographic operation.

FTP_PRO.2.1 **The TSF shall establish a shared secret with its peer using one of the following mechanisms: [assignment: *list of key establishment mechanisms*].**

FTP_PRO.2.2 **The TSF shall authenticate [selection*: its peer, itself to its peer*] using one of the following mechanisms: [assignment: *list of authentication mechanisms*] and according to the following rules: [assignment: *list of rules for carrying out the authentication*].**

FTP_PRO.2.3 **The TSF shall use [assignment: *key derivation function*] to derive the following cryptographic keys from a shared secret: [assignment: *list of cryptographic keys*].**

User application notes

For each iteration of FTP_PRO.1 by the PP/ST author, which represents a different protocol, a corresponding iteration of FTP_PRO.2 is needed in the PP/ST. For iterations of FTP_PRO.1 by the PP/ST author, which only express the behaviour of the TSF for different protocol roles of the same protocol, the same instantiation of FTP_PRO.2 may be suitable to fulfil the dependency of such FTP_PRO.1 iterations.

Operations

Assignment:

In FTP_PRO.2.2, the PP/ST author may use the ‘list of rules for carrying out the authentication’ to limit available parameters for the authentication mechanisms. For example, rules might be stated for the format (e.g. FQDN or IP address, use of wildcards) or prioritisation of identifiers when alternative sources of an identifier are available in the authentication data exchanged.

FTP_PRO.3 Trusted channel data protection

Hierarchical to: No other components.

Dependencies: FTP_PRO.1 Trusted channel protocol

FTP_PRO.2 Trusted channel establishment

FCS_COP.1 Cryptographic operation.

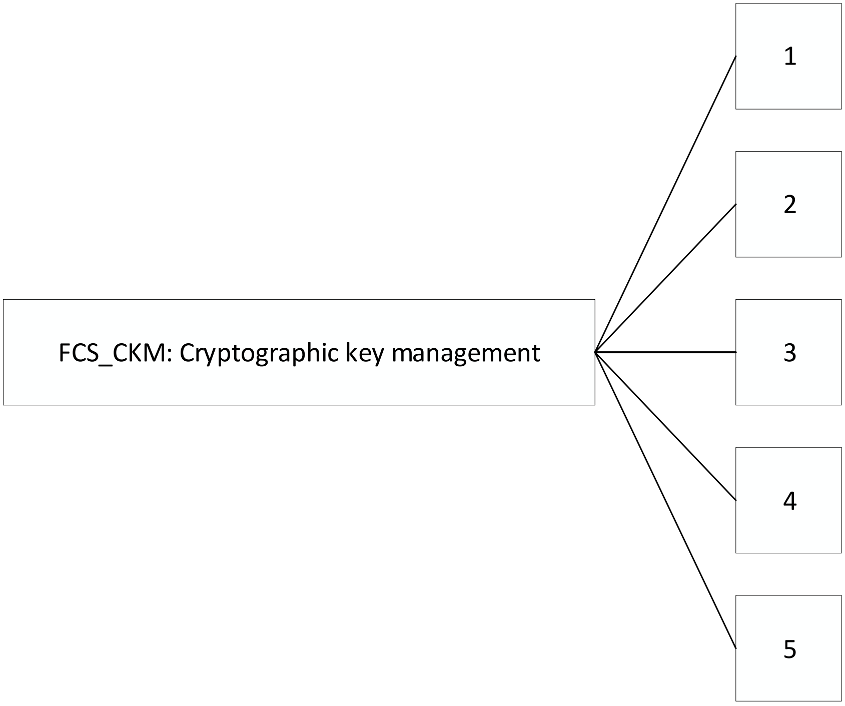
FTP_PRO.3.1 **The TSF shall protect data in transit from unauthorised disclosure using one of the following mechanisms: [assignment: *list of encryption mechanisms*].**

FTP_PRO.3.2 **The TSF shall protect data in transit from [selection: *modification, deletion, insertion, replay,* [assignment: *other*]] using one of the following mechanisms: [assignment: *list of integrity protection mechanisms*].**

5.2 Definition of the component cryptographic key management (FCS_CKM.5)

This chapter describes functional requirements for key derivation as process by which one or more keys are calculated from either a pre-shared key or a shared secret and other information. The component is part of the family FCS_CKM of the class FCS. The component FCS_CKM.5 has been specified as follows:

Component levelling



Management: FCS_CKM.5

There are no management activities foreseen.

Audit: FCS_CKM.5

There are no actions defined to be auditable.

FCS_CKM.5 Cryptographic key derivation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1 **The TSF shall derive cryptographic keys [assignment: *key type*] from [assignment: *input parameters*] in accordance with a specified cryptographic key derivation algorithm [assignment: *cryptographic key derivation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].**

5.3 Definition of the family TOE emanation (FPT_EMS)

Family behaviour

This family defines requirements to mitigate intelligible emanations.

Component levelling

Management: FPT_EMS.1

There are no management activities foreseen.

Audit: FPT_EMS.1

There are no actions defined to be auditable.

FPT_EMS.1 TOE emanation

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_EMS.1.1 **The TOE shall not emit [assignment: *types of emissions*] in excess of [assignment: *specified limits*] enabling access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].**

FPT_EMS.1.2 **The TSF shall ensure [assignment: *type of users*] are unable to use [assignment: *types of interfaces/ports*] to gain access to [assignment: *list of types of TSF data*] and [assignment: *list of types of user data*].**

6 Security requirements

6.1 Security functional requirements

6.1.1 Trusted channel and trusted path

FTP_PRO.1 Trusted channel protocol

Hierarchical to: No other components.

Dependencies: FTP_PRO.2 Trusted channel key establishment

FTP_PRO.3 Trusted channel data protection

FTP_PRO.1.1 The TSF shall implement [assignment: *trusted channel protocol*] acting as [assignment: *defined protocol role(s)*] in accordance with: [assignment: *list of standards*].

FTP_PRO.1.2 The TSF shall enforce usage of the trusted channel for [assignment: *purpose(s) of the trusted channel*] in accordance with: [assignment: *list of standards*].

FTP_PRO.1.3 The TSF shall permit [selection: *itself, its peer*] to initiate communication via the trusted channel.

FTP_PRO.1.4 The TSF shall enforce the following rules for the trusted channel: [assignment: *rules governing operation and use of the trusted channel and/or its protocol*].

FTP_PRO.1.5 The TSF shall enforce the following static protocol options: [assignment: *list of options and references to standards in which each is defined*].

FTP_PRO.1.6 The TSF shall negotiate one of the following protocol configurations with its peer: [assignment: *list of configurations and reference to standards in which each is defined*].

AN(FTP_PRO.1): The ST/PP author shall model both, trusted channel between the TSF and a network device and trusted path (i.e. end-to-end secured connection) between the TSF and the IoT device admin, by FTP_PRO.1. If different protocols are used to realize trusted channel and trusted path, the ST/PP author shall iterate FTP_PRO.1 as FTP_PRO.1/TC and FTP_PRO.1/TP. Furthermore, according to the user application notes for FTP_PRO.1, the ST/PP author may have to further iterate FTP_PRO.1 (or FTP_PRO.1/TC and/or FTP_PRO.1/TP, if applicable) for different protocol roles.

For specific functions listed in O.SCM.UtilizeSE that are assigned to the generation of a trusted channel, the IoT SE has to be used.

FTP_PRO.2 Trusted channel establishment

Hierarchical to: No other components.

Dependencies: FTP_PRO.1 Trusted channel protocol

[FCS_CKM.1 Cryptographic key generation, or
FCS_CKM.2 Cryptographic key distribution]

FCS_CKM.5 Cryptographic key derivation

FCS_COP.1 Cryptographic operation

FTP_PRO.2.1 The TSF shall establish a shared secret with its peer using one of the following mechanisms: [assignment: *list of key establishment mechanisms*].

FTP_PRO.2.2 The TSF shall authenticate [selection: *its peer, itself to its peer*] using one of the following mechanisms: [assignment: *list of authentication mechanisms*] and

according to the following rules: [assignment: *list of rules for carrying out the authentication*].

FTP_PRO.2.3 **The TSF shall use [assignment: *key derivation function*] to derive the following cryptographic keys from a shared secret: [assignment: *list of cryptographic keys*].**

AN(FTP_PRO.2): The ST/PP author shall model both, trusted channel establishment between the TSF and a network device and trusted path (i.e. end-to-end secured connection) establishment between the TSF and the IoT device admin, by FTP_PRO.2. If different protocols are used to realize trusted channel and trusted path, the ST/PP author shall iterate FTP_PRO.2 as FTP_PRO.2/TC and FTP_PRO.2/TP.
To satisfy remaining open dependencies of FTP_PRO.2, the ST/PP author has to include FCS_CKM.1 or FCS_CKM.2 in the ST/PP according to the actual key management related to the chosen trusted channel protocols.

For specific functions listed in O.SCM.UtilizeSE that are assigned to the generation of a trusted channel, the IoT SE has to be used.

FTP_PRO.3 Trusted channel data protection

Hierarchical to: No other components.

Dependencies: FTP_PRO.1 Trusted channel protocol

FTP_PRO.2 Trusted channel key establishment

FCS_COP.1 Cryptographic operation

FTP_PRO.3.1 **The TSF shall protect data in transit from unauthorised disclosure using one of the following mechanisms: [assignment: *list of encryption mechanisms*].**

FTP_PRO.3.2 **The TSF shall protect data in transit from [selection: *modification, deletion, insertion, replay, [assignment: other]*] using one of the following mechanisms: [assignment: *list of integrity protection mechanisms*].**

AN(FTP_PRO.2): The ST/PP author shall model both, trusted channel data protection between the TSF and a network device and trusted path (i.e. end-to-end) data protection between the TSF and the IoT device admin, by FTP_PRO.3. If different protocols are used to realize trusted channel and trusted path, the ST/PP author shall iterate FTP_PRO.3 as FTP_PRO.3/TC and FTP_PRO.3/TP.

For specific functions listed in O.SCM.UtilizeSE that are assigned to the generation of a trusted channel, the IoT SE has to be used.

FCS_CKM.5 Cryptographic key derivation

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution, or
FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.5.1 **The TSF shall derive cryptographic keys [assignment: *key type*] from [assignment: *input parameters*] in accordance with a specified cryptographic key derivation algorithm [assignment: *cryptographic key derivation algorithm*] and specified cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].**

AN(FCS_CKM.5): The ST/PP author shall iterate FCS_CKM.5 if necessary to cover all corresponding dependencies concerning cryptographic key derivation arising from FTP_PRO.2 or iterations thereof.

For specific functions listed in O.SCM.UtilizeSE that are assigned to cryptographic key generation, the IoT SE has to be used.

6.1.2 Network connection control

FDP_ACC.1/NCC Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/NCC **The TSF shall enforce the *network connection control* policy[i] on**
(1) objects: external network devices;
***(2) operations: establishing network connection*[ii].**

FDP_ACF.1/NCC Security attribute based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/NCC **The TSF shall enforce the *network connection control policy*[iii] to objects based on the following:**
(1) objects: external network devices;
(2) attributes: requested network address of external device,
requested connection protection level, i.e. 'not protected' or 'protected by a trusted channel',
***connection control rule (tuple of allowed network address and required minimum connection protection level, i.e. 'no protection necessary' or 'to be protected by a trusted channel')*[iv].**

FDP_ACF.1.2/NCC **The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:**
Establishing network connection to an external network device is allowed, if there is a connection control rule configured in the TOE, whose allowed network address matches the requested network address and whose required minimum connection protection level is matched or exceeded by the requested connection protection level.

FDP_ACF.1.3/NCC **The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*[v].**

FDP_ACF.1.4/NCC **The TSF shall explicitly deny access of subjects to objects based on the following additional rules:**
***Establishing connection to an external network device is denied, if the corresponding connection rule contains the attribute 'to be protected by a trusted channel', but the TSF fails to establish the corresponding trusted channel to the external network device*[vi].**

AN(FDP_ACF.1/NCC): In FDP_ACF.1/NCC, the definition of security attributes and their possible values shall be just seen as means to express the access control rules. The TOE developer shall be free to implement the access control rules based on the definition of security attributes as stated above, or by different means, e.g. by taking a different number of attributes, or different values for the security attributes, as long as the access control rules above are still enforced as intended.

In FDP_ACF.1.4/NCC, failure to establish the trusted channel means not (completely) fulfilling FTP_PRO.1, FTP_PRO.2 and/or FTP_PRO.3 and iterations and dependencies thereof, which are related to the trusted channel protocol used for the trusted channel to be established.

The dependency to FMT_MSA.3 is not applicable. There are no default values for the attributes of this access control policy as the controlled objects, i.e. the external network devices, are not created under this access control policy

6.1.3 TOE management

FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1 **The TSF shall maintain the roles *IoT device admin and IoT device user*[vii].**

FMT_SMR.1.2 **The TSF shall be able to associate users with roles.**

FIA_UID.1 Timing of identification

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

FIA_UID.1.1 **The TSF shall allow *querying version information of the TOE and version information of the IoT SE*[viii] on behalf of the user to be performed before the user is identified.**

FIA_UID.1.2 **The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.**

AN(FIA_UID.1): The IoT device admin is identified and authenticated during establishment of a trusted path between the TSF and the IoT device admin, therefore there is no need for the TOE developer to come up with an additional identification and authentication mechanism for the IoT device admin. Still, for the IoT device user a dedicated identification and authentication mechanism is needed.

FIA_UAU.1 Timing of authentication

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.1.1 **The TSF shall allow *querying version information of the TOE and version information of the IoT SE*[ix] on behalf of the user to be performed before the user is authenticated.**

FIA_UAU.1.2 **The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.**

AN(FIA_UAU.1): The IoT device admin is identified and authenticated during establishment of a trusted path between the TSF and the IoT device admin, therefore there is no need for the TOE developer to come up with an additional identification and authentication mechanism for the IoT device admin. Still, for the IoT device user a dedicated identification and authentication mechanism is needed.

FMT_SMF.1 Specification of management functions

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 **The TSF shall be capable of performing the following management functions:**

1) ***Query version information of the TOE.***
2) ***Query version information of the IoT SE.***
3) ***Create, query, modify, delete network connection control rules (tuples of allowed network address and required minimum connection protection level)***[x]***.***

FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles

FMT_SMF.1 Specification of management functions

FMT_MTD.1.1 **The TSF shall restrict the ability to *modify, delete, create*[xi] the network *connection control rules (tuples of allowed network address and required minimum connection protection level)*[xii] to [assignment: the authorised identified roles][xiii].**

AN(FMT_MTD.1): In the uncompleted assignment operation in FMT_MDT.1.1, the ST/PP author shall enter either 'IoT device admin' or 'IoT device admin and IoT device user'.

6.1.4 SCM firmware update control

FDP_ACC.1/SCMFW Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/SCMFW **The TSF shall enforce the *IoT SCM firmware update policy*[xiv] on**

(1) objects: SCM FW update image, SCM-FAK, SCM-FCK;
(2) operations: SCM FW update, SCM FW key update[xv]***.***

FDP_ACF.1/SCMFW Security attribute based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/SCMFW **The TSF shall enforce the *IoT SCM firmware update policy*[xvi] to objects based on the following:**

(1) objects: SCM FW update image, SCM-FAK, SCM-FCK;
(2) attributes: SCM-FAK signature/MAC, SCM FW update version, Latest SCM FW version[xvii]***.***

FDP_ACF.1.2/SCMFW **The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:**

(1) SCM FW update is allowed, if the SCM-FAK signature/MAC is successfully verified against the corresponding SCM FW update image and SCM FW update version presented in the SCM FW firmware update request;
(2) SCM FW key update is allowed, if the SCM-FAK signature/MAC is successfully verified against the corresponding new SCM-FAK and/or the new SCM-FCK presented in the SCM FW key update request[xviii]***.***

FDP_ACF.1.3/SCMFW **The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*[xix].**

FDP_ACF.1.4/SCMFW **The TSF shall explicitly deny access of subjects to objects based on the following additional rules: SCM FW update is denied, if the SCM FW version presented in the SCM FW update request is older than the Latest SCM FW version[xx].**

AN(FDP_ACF.1/SCMFW): The dependency to FMT_MSA.3 is not applicable. There are no default values for the attributes of this access control policy as the controlled objects, i.e. the SCM FW update image, SCM-FAK and SCM-FCK, are not created under this access control policy.

Remark: By enforcement of the explicit deny rule it shall be prevented that an attacker, by just applying a signed or MAC-protected SCM firmware update image as officially released by the SCM developer, can downgrade the SCM firmware to an older version (e.g., to undo security fixes that were introduced in a newer SCM firmware version). Still, the SCM developer would have the ability to revert the SCM firmware back to an older release (e.g., in case a newly issued firmware release shows problems or errors), by creating a new signature or MAC over the SCM firmware update image of the older release together with some newer version number (which would be just introduced to enable this intentional firmware downgrading).

Downgrading protection concerning SCM FW authentication key and SCM FW confidentiality key is not necessary, as an old key update request cannot be replayed successfully once the SCM FW authentication key has been updated in the TOE.

6.1.5 Cryptographic operation

FCS_COP.1 Cryptographic operation

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 **The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].**

AN(FCS_COP.1): There are several SFRs in this PP, which model functionality making use of cryptographic operations. The author of this PP cannot decide, how many different cryptographic operations (also in terms of cryptographic algorithm, key size and applicable standard) would be necessary for a concrete TOE conformant to this PP. To avoid that this PP is bloated up with a lot of iterations of FCS_COP.1, which in the end could lead to a highly redundant set of SFRs in the ST/PP based on this PP, it is left open to the ST/PP author to iterate FCS_COP.1 in a way that all SFR dependencies requiring FCS_COP.1 are satisfied, and that also all cryptographic operations, which are needed to cover the security objectives of the TOE, are included in the final set of SFRs of the ST/PP. (Completeness of the FCS_COP.1 iterations will have to be shown in the ST/PP in terms of the SFR dependency rationale and the security objectives rationale anyway.)

Furthermore, as the dependencies concerning the key management related to the cryptographic operation modelled by FCS_COP.1, i.e. FDP_ITC.1, FDP_ITC.2, FCS_CKM.1 and FCS_CKM.4,

- may be satisfied very differently for different concrete TOEs,
- may be satisfied very differently even for different keys of the same TOE,

- may be rightfully left unsatisfied with a corresponding rationale given, or
- may be satisfied by the very same iteration of FDP_ITC.1, FDP_ITC.2, FCS_CKM.1 and/or FCS_CKM.4 even for several iterations of FCS_COP.1,

none of these dependencies SFRs have been included in this PP already. It is up to the ST/PP author to make sure that all those dependencies will be satisfied for all iterations of FCS_COP.1 as finally stated in the ST/PP. Satisfaction of dependencies has to be shown in the SFR dependency rationale in the ST/PP for all iterations of all SFRs independently anyway.

To still allow a somehow meaningful dependency rationale and security objectives rationale in this PP, in the following the dependencies and security functional requirements needing instances/iterations of FSC_COP.1 in the ST/PP are listed:

- cryptographic operation needed for FTP_PRO.2 shared secret establishment,
- cryptographic operation needed for FTP_PRO.2 key derivation,
- cryptographic operations ‘encryption and decryption’ according to FTP_PRO.3,
- cryptographic operation ‘integrity protection’ according to FTP_PRO.3,
- cryptographic operation needed for signature/MAC verification of SCM FW image and SCM FW keys according to FDP_ACF.1/SCMFW,
- cryptographic operation needed for decryption of SCM FW image and SCM FW keys according to FDP_ACF.1/SCMFW.

In each iteration of FCS_COP.1 in the ST/PP, in the assignment about the ‘list of cryptographic operations’ the ST/PP author shall also enter the corresponding keys being used, e.g., ‘signature/MAC verification using SCM-FAK’ or ‘decryption using SCM-FCK’. This will allow to easier map the FCS_COP.1 iterations to the related dependencies and security objectives, respectively.

Finally, for all iterations of FCS_COP.1 the choice of cryptographic algorithms and cryptographic key sizes has to ensure the required minimal security level of 100 bit for all cryptographic operations in their corresponding use case/protocol.

6.1.6 Logical protection

FPT_EMS.1 TOE emanation

Hierarchical to: No other components.

Dependencies: None.

FPT_EMS.1.1 **The TOE shall not emit *information in terms of timing*[xxi] in excess of [assignment: *specified limits*] enabling access to *cryptographic keys*[xxii] and [assignment: *list of types of user data*].**

FPT_EMS.1.2 **The TSF shall ensure *all users*[xxiii] are unable to use *any kind of external network interface/port of the TOE*[xxiv] to gain access to *cryptographic keys*[xxv] and [assignment: *list of types of user data*].**

AN(FPT_EMS.1): If there should be no user data to be protected from timing attacks, for better readability the ST/PP author may refine FPT_EMS.1.1 and FTP_EMS.1.2 by removing the text “and [assignment: *list of types of user data*]”.

6.2 Security assurance requirements

The security assurance requirements for this TOE shall be EAL2 augmented by ALC_FLR.1 as defined in CC Part 3:

Assurance class	Assurance components	
ADV: Development	ADV_ARC.1 ADV_FSP.2 ADV_TDS.1	Security architecture description Security-enforcing functional specification Basic design
AGD: Guidance documents	AGD_OPE.1 AGD_PRE.1	Operational user guidance Preparative procedures
ALC: Life-cycle support	ALC_CMC.2 ALC_CMS.2 ALC_DEL.1 ALC_FLR.1	Use of a CM system Parts of the TOE CM coverage Delivery procedures Basic flaw remediation (augmented)
ASE: Security Target evaluation	ASE_CCL.1 ASE_ECD.1 ASE_INT.1 ASE_OBJ.2 ASE_REQ.2 ASE_SPD.1 ASE_TSS.1	Conformance claims Extended components definition ST introduction Security objectives Derived security requirements Security problem definition TOE summary specification
ATE: Tests	ATE_COV.1 ATE_FUN.1 ATE_IND.2	Evidence of coverage Functional testing Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2	Vulnerability analysis (refined)

Table 4: Security assurance requirements (EAL2 augmented by ALC_FLR.1)

AVA_VAN.2 shall be refined in three of its elements as follows:

AVA_VAN.2.2E **The evaluator *shall perform* a search of public domain sources to identify potential vulnerabilities in the TOE.**
Refinement: This search for potential vulnerabilities shall cover all network services provided by the TOE, regardless whether these are part of the TSF or not.

AVA_VAN.2.3E **The evaluator *shall perform* an independent vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design and security architecture description to identify potential vulnerabilities in the TOE.**
Refinement: This independent vulnerability analysis shall cover all network services provided by the TOE, regardless whether these are part of the TSF or not.

AVA_VAN.2.4E **The evaluator *shall conduct* penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.**
Refinement: This penetration testing shall cover all network services provided by the TOE (regardless whether these are part of the TSF or not) as far as non-exploitability of the corresponding identified potential vulnerabilities cannot be determined by other means.

6.3 Security requirements rationale

6.3.1 Security functional requirement (SFR) rationale

6.3.1.1 Fulfilment of the security objectives of the TOE

The following table shows that all SFRs chosen trace back to TOE security objectives. As TOE security objectives O.SCM.LeakageProt and O.SCM.UtilizeSE cannot be covered by TOE functionality, but by aspects of the TOE's architecture and security architecture, there are also some SARs tracing back to those security objectives.

TOE security objectives SFRs and SARs	O.SCM.AuthProt	O.SCM.ConfProt	O.SCM.ConnectControl	O.SCM.Auditability	O.SCM.SecureUpdate	O.SCM.LeakageProt	O.SCM.UtilizeSE
FTP_PRO.1	X	X					X
FTP_PRO.2	X	X					X
FTP_PRO.3	X	X					X
FCS_CKM.5	X	X					X
FDP_ACC.1/NCC			X				
FDP_ACF.1/NCC			X				
FMT_SMR.1				X			
FMT_UID.1				X			
FMT_UAU.1				X			
FMT_SMF.1				X			
FMT_MTD.1				X			
FDP_ACC.1/SCMFW					X		
FDP_ACF.1/SCMFW					X		
FPT_EMS.1						X	
FCS_COP.1	X	X			X		

Table 5: Tracing back security requirements to TOE security objectives

The following table shows that the SFRs meet the TOE security objectives:

TOE security objective	SFR	Rationale
O.SCM.AuthProt	FTP_PRO.1	Defines the requirement to use a well-defined trusted channel protocol including protocol options, operational rules, allowed configurations, etc. and is therefore the base for the authenticity protection from the objective
	FTP_PRO.2	Defines the requirement for well-defined authentication and key establishment mechanisms in the trusted channel protocol. Authentication directly contributes to meeting the objective, the key establishment may be used as a base to derive further data authentication keys (e.g., session keys)
	FTP_PRO.3	Defines the requirement for well-defined key derivation mechanisms in the trusted channel protocol, which may be used to derive further data authentication keys (e.g., session keys)
	FCS_CKM.5	Defines the requirement to use a specific standardized key derivation algorithm with specified key size
	FCS_COP.1	Defines the requirement to use a specific standardized cryptographic operation (primitive) as part of the key derivation algorithm
O.SCM.ConfProt	FTP_PRO.1	Defines the requirement to use a well-defined trusted channel protocol including protocol options, operational rules, allowed configurations, etc. and is therefore the base for the confidentiality protection from the objective
	FTP_PRO.2	Defines the requirement for well-defined authentication and key establishment mechanisms in the trusted channel protocol. Authentication directly contributes to meeting the objective, the key establishment may be used as a base to derive further data authentication keys (e.g., session keys)
	FTP_PRO.3	Defines the requirement for well-defined key derivation mechanisms in the trusted channel protocol, which may be used to derive further data authentication keys (e.g., session keys)
	FCS_CKM.5	Defines the requirement to use a specific standardized key derivation algorithm
	FCS_COP.1	Defines the requirement to use a specific standardized cryptographic operation (primitive) as part of the key derivation algorithm
O.SCM.ConnectControl	FDP_ACC.1/NCC	Defines the requirement for a connection control policy and defines the corresponding objects (external network devices) and operations (connection establishment)
	FDP_ACF.1/NCC	Defines the requirement for security attribute based access control for the connection establishment, the corresponding security attributes and the rules allowing only those connections, which have been configure in terms

TOE security objective	SFR	Rationale
		of connection control rules (security attribute). Requested connections, which are not configured at all or whose connection rules do not match the request, are denied.
O.SCM.Auditability	FMT_SMR.1	Defines the requirement that the TOE is aware of the necessary roles
	FMT_UID.1	Defines the requirement that querying version of the TOE and of the SE is possible prior to user identification
	FMT_UAU.1	Defines the requirement that querying version of the TOE and of the SE and the network control rules is possible prior to user authentication
	FMT_SMF.1	Defines the requirement that – among others – functionality for querying TOE and SE version and network connection control rules is provided by the TOE
	FMT_MTD.1	Defines the user roles and the accessible settings by the defined roles
O.SCM.SecureUpdate	FDP_ACC.1/NCC	Defines the requirement for a firmware update policy and defines the corresponding objects, which can be updated, and the update operations
	FDP_ACF.1/NCC	Defines the requirement for security attribute based access control for the update operations, the corresponding security attributes and the rules allowing only authentic images and keys to be updated, and preventing downgrading
	FCS_COP.1	Shall define at the latest in the ST the cryptographic operations signature/MAC verification and decryption, which are needed for securing the update process. SFR is included in this PP, but not iterated/customized to the necessary cryptographic operations yet. Nevertheless, sufficient guidance is given to the ST author in AN(FCS_COP.1) about the cryptographic operations to be modelled and how these shall be modelled, that coverage of the cryptographic aspect of the objective is deemed covered
O.SCM.LeakageProt	FPT_EMS.1	Defines the requirement that cryptographic keys, especially the IoT session key received from the IoT SE, shall be protected against disclosure by timing information by all users via all interfaces/ports of the TOE.
O.SCM.UtilizeSE	FTP_PRO.1	Defines the requirement to use a well-defined trusted channel protocol including protocol options, operational rules and allowed configurations. For specific functions assigned to the setup of a trusted channel, functions provided by the IoT SE have to be used, see also AN(FTP_PRO.1).
	FTP_PRO.2	Defines the requirement for well-defined authentication and key establishment mechanisms in the trusted channel protocol. As described in the objective, the IoT SE has to be

TOE security objective	SFR	Rationale
		used for specific functions assigned to the setup of a trusted channel, see also AN(FTP_PRO.2).
	FTP_PRO.3	Defines the requirement for well-defined key derivation mechanisms in the trusted channel protocol, which may be used to derive further data authentication keys (e.g., session keys). As described in the objective, the IoT SE has to be used for specific functions assigned to the setup of a trusted channel, see also AN(FTP_PRO.3).
	FCS_CKM.5	Defines the requirement to use a specific standardized key derivation algorithm. As described in the objective the IoT SE has to be used for specific key derivation scenarios, see also AN(FCS_CKM.5).

Table 6: Mapping of security requirements to TOE security objectives

6.3.1.2 Fulfilment security functional requirement (SFR) dependencies

The following table shows that the dependencies between the SFRs are either satisfied within this PP or corresponding rationale is referenced (typically provided in SFR application notes AN), why a dependency is either not applicable at all or why it has been left open to be satisfied by the ST/PP developer:

SFR	Dependency	Satisfied in this PP?
FTP_PRO.1	FTP_PRO.2 FTP_PRO.3	Yes Yes
FTP_PRO.2	FTP_PRO.1 FCS_CKM.1 or FCS_CKM.2 FCS_CKM.5 FCS_COP.1	Yes No, qualified by AN(FTP_PRO.2) Yes, qualified by AN(FCS_CKM.5) Yes, qualified by AN(FCS_COP.1)
FTP_PRO.3	FTP_PRO.1 FTP_PRO.2 FCS_COP.1	Yes Yes Yes, qualified by AN(FCS_COP.1)
FCS_CKM.5	FCS_CKM.2 or FCS_COP.1 FCS_CKM.4	Yes No, qualified by AN(FCS_COP.1)
FDP_ACC.1/NCC	FDP_ACF.1	Yes (by FDP_ACF.1/NCC)
FDP_ACF.1/NCC	FDP_ACC.1 FMT_MSA.3	Yes (by FDP_ACC.1/NCC) No, not applicable as qualified by AN(FDP_ACF.1/NCC)
FMT_SMR.1	FIA_UID.1	Yes
FMT_UID.1	none	
FMT_UAU.1	FIA_UID.1	Yes
FMT_SMF.1	none	
FMT_MTD.1	FMT_SMR.1 FMT_SMF.1	Yes Yes
FDP_ACC.1/SCMFW	FDP_ACF.1	Yes (by FDP_ACF.1/SCMFW)

SFR	Dependency	Satisfied in this PP?
FDP.ACF.1/SCMFW	FDP_ACC.1 FMT_MSA.3	Yes (by FDP_ACC.1/SCMFW) No, not applicable as qualified by AN(FDP_ACF.1/SCMFW)
FCS_COP.1	FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1 FCS_CKM.4	No, qualified by AN(FCS_COP.1) No, qualified by AN(FCS_COP.1)

Table 7: Satisfaction of SFR dependencies

6.3.2 Security assurance requirement (SAR) rationale

The primary use case for the IoT SCM is to be integrated in IoT host devices like home appliances running in a household with no physical access by potential attackers[9]. Furthermore, there is an SFR that requires countermeasures against timing analyses[10]. As finally the TOE has to be resistant against network-based penetration attacks (which are already made more difficult by the requirement of a security level of at least 100 bit for all cryptographic security functions, see OSP.SCM.StrongCrypto and corresponding footnote hereinafter), the evaluation assurance level **EAL2** including vulnerability assessment component **AVA_VAN.2 (refined)** was chosen (providing assurance concerning **resistance of the TOE against attackers possessing *basic* attack potential** and requiring **vulnerability analysis concerning all network services provided**). The AVA_VAN.2 vulnerability analysis has to regard all applicable publicly known vulnerabilities. For the TOE's main security needs, i.e. securely implemented network protocols and – if used – a securely implemented underlying general-purpose operating system, those are readily available in form of comprehensive public databases containing commonly known vulnerabilities. By the refinement of AVA_VAN.2, the vulnerability analysis explicitly has to cover known vulnerabilities for all network services provided by the TOE (and not only those related to its evaluated security functions), to make sure that the IoT SCM cannot be compromised by any kind of known network attack.

For security flaws detected in the TOE once evaluated and certified, the TOE developer is expected to have basic flaw remediation procedures in place, therefore **ALC_FLR.1** is augmented. (As long as a flaw could be remediated in firmware and no changes to the hardware of the TOE would be necessary, the TOE developer would simply issue a corresponding firmware update for the TOE as part of their flaw remediation procedure.)

[9] IoT device users are not deemed attacking the IoT SCM, as they would have no motivation to do so (IoT SCM is protecting data and LAN of IoT device users).

[10] Timing analyses could be mounted remotely over network connections, whereas analyses of power consumption or electromagnetic emanation is deemed unrealistic, because of the lack of physical access.

7 Annex

7.1 References

[IoT-SE-PP] *IoT Secure Element Module Protection Profile (IoT-SE-PP)*, version 1.0.0, 2019-12-19, by Secure Communications Alliance (SCA).

7.2 Glossary

AIS	Applications and Interpretations of the Scheme (by German BSI)
AN	Application Note
Authenticity	Provable property of data that data have been created by a specific originator and that the data have not been corrupted after its creation (the latter meaning that authenticity also covers integrity of the data)
CC	Common Criteria
EAL	Evaluation Assurance Level
IDK	IoT Device Key
IoT	Internet of Things
LAN	Local Area Network
PP	Protection Profile
SAR	Security Assurance Requirement
SCA	Secure Communications Alliance
SCM	Secure Communications Module
SCM-FAK	SCM Firmware Authenticity Key
SCM-FCK	SCM Firmware Confidentiality Key
SE	Secure Element
security level	The security level of a cryptographic mechanism is usually given as the number of operations necessary for an adversary to successfully break the security provided by the mechanism. It is expressed as a base 2 logarithm, e.g., 100 bits of security means that 2^{100} operations are necessary.[11]
SFR	Security Functional Requirement

[11] The reader may consult NIST SP 800-57 part 1, Tables 2 and 3, for a first orientation on the security level of some well-known cryptographic algorithms. The final rating is up to the TOE's CC certification scheme, though.

7.3 Original SFR Operations as Defined in CC Part 2

End notes (indicated by Roman numerals) on assignment and selection operations in SFRs in section 6.1, which have partially or completely been executed in this PP, will lead to the following original assignment or selection operation statements from CC part 2:

[i] **[assignment:** ***access control SFP*****]**

[ii] **[assignment:** ***list of subjects, objects, and operations among subjects and objects covered by the SFP*****]**

[iii] **[assignment:** ***access control SFP*****]**

[iv] **[assignment:** ***list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*****]**

[v] **[assignment:** ***rules, based on security attributes, that explicitly authorise access of subjects to objects*****]**

[vi] **[assignment:** ***rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*****]**

[vii] **[assignment:** ***the authorised identified roles*****]**

[viii] **[assignment*****: list of TSF-mediated actions*****]**

[ix] **[assignment:** ***list of TSF mediated actions*****]**

[x] **[assignment:** ***list of management functions to be provided by the TSF*****]**

[xi] **[selection:** ***change_default, query, modify, delete, clear, [assignment: other operations]*****]**

[xii] **[assignment:** ***list of TSF data*****]**

[xiii] **[assignment:** ***the authorised identified roles*****]**

[xiv] **[assignment:** ***access control SFP*****]**

[xv] **[assignment:** ***list of subjects, objects, and operations among subjects and objects covered by the SFP*****]**

[xvi] **[assignment:** ***access control SFP*****]**

[xvii] **[assignment:** ***list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*****]**

[xviii] **[assignment:** ***rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*****]**

[xix] **[assignment:** ***rules, based on security attributes, that explicitly authorise access of subjects to objects*****]**

[xx] **[assignment:** ***rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*****]**

[xxi] **[assignment:** ***types of emissions*****]**

[xxii] **[assignment:** ***list of types of TSF data*****]**

[xxiii] **[assignment:** ***type of users*****]**

[xxiv] **[assignment:** ***types of interfaces/ports*****]**

[xxv] **[assignment:** ***list of types of TSF data*****]**