INTERNATIONAL STANDARD

ISO/IEC 27071

First edition
2023-07

Cybersecurity — Security recommendations for establishing trusted connections between devices and services

Cybersécurité — Recommandations de sécurité pour l'établissement de connexions de confiance entre dispositifs et services

Reference number
ISO/IEC 27071:2023(E)

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *Information Security, cybersecurity and privacy protection*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

With the development of the internet of things (IoT), mobile services, cloud computing, big data and artificial intelligence (AI), it is essential to establish trusted connections between devices and services in a growing number of scenarios.

Security channels [e.g. secure sockets layer (SSL) or transport layer security (TLS) protocols] are used between devices and services to protect confidentiality and integrity of data, but it is not enough. It is essential for the service to distinguish data collected by sensors of the authorized device from those of other devices or data forged by adversaries. Thus, the service should be able to ensure that the data comes from the authorized device.

In addition, it is crucial for the device to distinguish the genuine service from unintended services or malicious services. In this way, it should be able to reliably identify the genuine and intended service, in particular for cloud services, which may have thousands of such services running.

Identity without a reliable root of trust can be forged, so controls are critical to ensure the utilization of reliable roots of trust. The requirements for establishing reliable virtualized roots of trust are described in ISO/IEC 27070.

Mutual authentication between a device and a service is essential for preventing impersonation attacks. While insufficient in itself, remote attestation between a device and a service is also critical for protecting the data handling processes and establishing a security channel to prevent interception by an adversary on the communication network.

Data captured from sensors integrated in the device, input by users, or generated (or processed) by algorithms in the device should have a label and be digitally signed (or by other crypto mechanisms) using the device's particular key designed for this purpose, to protect the integrity and authenticity of the data. It is possible that services know the parameters of the sensor device which can help it to process the data. Trusted connections have a strong relationship with hardware security modules (HSM), trusted computing (TC), public key infrastructure (PKI) and certification authority (CA) technology. Trusted connection issues can be broken down into several sub-categories such as:

— hardware security modules to establish the reliable root of trust;

— identity of devices and services issued by trusted parties;

— mutual authentication and key establishment between devices and services to establish a security channel;

— mutual remote attestation (or environment assurance) between devices and services;

— data identity to keep the data integrity and authenticity long term.

This document proposes security recommendations for establishing trusted connections between devices and services, which would help the related organisations to set up HSM in devices (including mobile devices, PCs, or IoT devices) and in the infrastructure of cloud services. This document can help to build a trusted environment. This document can also help trusted third parties (i.e. CA) to issue certificates to devices and services, and help applications to mitigate against attacks and identify forged data from the sensors.

Cybersecurity — Security recommendations for establishing trusted connections between devices and services

1 Scope

This document provides a framework and recommendations for establishing trusted connections between devices and services based on hardware security modules. It includes recommendations for components such as: hardware security module, roots of trust, identity, authentication and key establishment, remote attestation, data integrity and authenticity.

This document is applicable to scenarios that establish trusted connections between devices and services based on hardware security modules.

This document does not address privacy concerns.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 27070, *Information technology — Security techniques — Requirements for establishing virtualized roots of trust*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 27070 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1 Terms relating to cloud computing

3.1.1
cloud computing
paradigm for enabling network access to a scalable and elastic pool of shareable physical or virtual resources with self-service provisioning and administration on-demand

Note 1 to entry: Examples of resources include servers, operating systems, networks, software, applications, and storage equipment.

[SOURCE: ISO/IEC 22123-1:2023, 3.1.1, modified — note 2 to entry has been deleted.]

3.1.2
cloud service
capabilities offered via *cloud computing* (3.1.1) invoked using a defined interface

[SOURCE: ISO/IEC 22123-1:2023, 3.1.2]

3.2 Terms relating to cloud computing roles and activities

3.2.1
party
natural person or legal person or a group of either, whether or not incorporated, that can assume one or more roles

[SOURCE: ISO/IEC 22123-1:2023, 3.3.1]

3.2.2
cloud service provider
party (3.2.1) that is acting in a *cloud service* (3.1.2) provider role

[SOURCE: ISO/IEC 22123-1:2023, 3.3.3]

3.2.3
cloud service user
natural person, or entity acting on their behalf, associated with a *cloud service customer* (3.2.2) that uses *cloud services* (3.1.2)

Note 1 to entry: Examples of such entities include devices and applications.

[SOURCE: ISO/IEC 22123-1:2023, 3.3.4]

3.2.4
tenant
cloud service user (3.2.4) sharing access to a set of physical and virtual resources

[SOURCE: ISO/IEC 22123-1:2023, 3.4.2, modified — "one or more" has been deleted from original definition.]

3.3 Terms relating to security and privacy

3.3.1
availability
property of being accessible and usable on demand by an authorized entity

[SOURCE: ISO/IEC 27000:2018, 3.7]

3.3.2
confidentiality
property that information is not made available or disclosed to unauthorized individuals, entities, or processes

[SOURCE: ISO/IEC 27000:2018, 3.10]

3.3.3
integrity
property of accuracy and completeness

[SOURCE: ISO/IEC 27000:2018, 3.36]

3.3.4
hardware security module
HSM
tamper-resistant hardware module which safeguards and manages keys and provides cryptographic functions

Note 1 to entry: Trusted module is a specific kind of HSM.

3.3.5
trust anchor module
TAM
hardware security module (3.3.4) that acts as the *roots of trust* (3.3.8)

Note 1 to entry: Trust anchor module is an abstract module that contains one or more hardware security modules.

3.3.6
trusted user interface
TUI
device component with a user interface whose *integrity* (3.3.3) and authenticity is managed by the *trust anchor module* (3.3.5)

3.3.7
identity key
IK
signing key used for authentication and to sign characteristics of the device (or service) environment (e.g. a digest) in order to prevent forgery and protect the *integrity* (3.3.3) of the device (or service) environment characteristics

3.3.8
root of trust
RoT
physical root of trust
component that needs to always behave in the expected manner because its misbehaviour cannot be detected

Note 1 to entry: The complete set of roots of trust has at least the minimum set of functions to enable a description of the platform characteristics that affect the trust of the platform.

[SOURCE: ISO/IEC 27070:2021, 3.4, modified — "physical root of trust" has been added as an admitted term.]

3.3.9
virtualized root of trust
vRoT
security function component established based on the *root of trust* (3.3.8), which provides similar function as the root of trust

Note 1 to entry: In practical environments, there can be multiple virtualized roots of trust based on the single root of trust simultaneously.

3.3.10
root of trust for measurement
computation engine that resets one or more platform configuration registers, makes the initial *integrity* (3.3.3) measurement, and extends it into a platform configuration register

Note 1 to entry: A *root of trust* (3.3.8) that collects device environment characteristics (e.g. firmware integrity measurements) and puts them in a format suitable for attestation (e.g. trusted platform module platform configuration registers).

3.3.11
root of trust for storage
component of the *root of trust* (3.3.8) that provides storing confidential information and measured values in shielded locations accessed using protected capabilities

3.3.12
root of trust for reporting
component of the *root of trust* (3.3.8) that reliably provides authenticity and nonrepudiation services for the purposes of attesting to the origin and *integrity* (3.3.3) of platform characteristics

Note 1 to entry: A root of trust that uses the device's (or service's) *identity key* (3.3.7) to reliably provide authenticity and nonrepudiation services for the purposes of attesting to the origin and integrity of device (or service) environment characteristics.

3.3.13
secure element
SE
tamper-resistant platform capable of securely hosting applications and their confidential and cryptographic data (for example cryptographic keys) in accordance with the rules and security requirements set by well-identified trusted authorities

3.3.14
trusted computing
TC
technology protected computer which consistently behaves in expected ways

3.3.15
trusted execution environment
TEE
execution environment that runs alongside but is isolated from the device main operating system

3.3.16
chain of trust
extension of trust from a component [e.g. a *root of trust* (3.3.8)] to another component accomplished through the act of measurement and verification of the *integrity* (3.3.3) and authenticity of the new component before the system begins execution of the new component

Note 1 to entry: Such an act builds a chain of trust from the old component to the new component, which is now a trusted component. The old component can be either a root of trust or a trusted component.

3.3.17
trusted environment
TE
execution mode where the functionality is protected by a *root of trust* (3.3.8) service

Note 1 to entry: A *trusted execution environment* (3.3.15) is a specific TE.

3.4 Miscellaneous terms

3.4.1
device
physical entity that communicates directly or indirectly with one or more *cloud services* (3.1.2)

[SOURCE: ISO/IEC 22123-1:2023, 3.13.4, modified — note 1 to entry has been deleted.]

3.4.2
device holder
person possessing and using the device

Note 1 to entry: In some cases, the person who possesses and uses the mobile device is the device holder. But in cases of Internet of Things, it is probably that sensors (devices) do not have a corresponding device holder.

4 Abbreviated terms

API	application programming interface
CA	certification authority (in a PKI)
CPU	central processing unit
HSM	hardware security module
IK	identity key
IMC	integrity measurement collectors
IMV	integrity measurement verifiers
PCR	platform configuration register
PKI	public key infrastructure
RoT	root of trust
REE	rich execution environment
RTM	root of trust for measurement
RTR	root of trust for reporting
RTS	root of trust for storage
SE	secure element
TAM	trust anchor module
TC	trusted computing
TCG	trusted computing group
TCM	trusted cryptography module
TE	trusted environment
TEE	trusted execution environment
TPM	trusted platform module
vRoT	virtualized root of trust

5 Framework and components for establishing a trusted connection

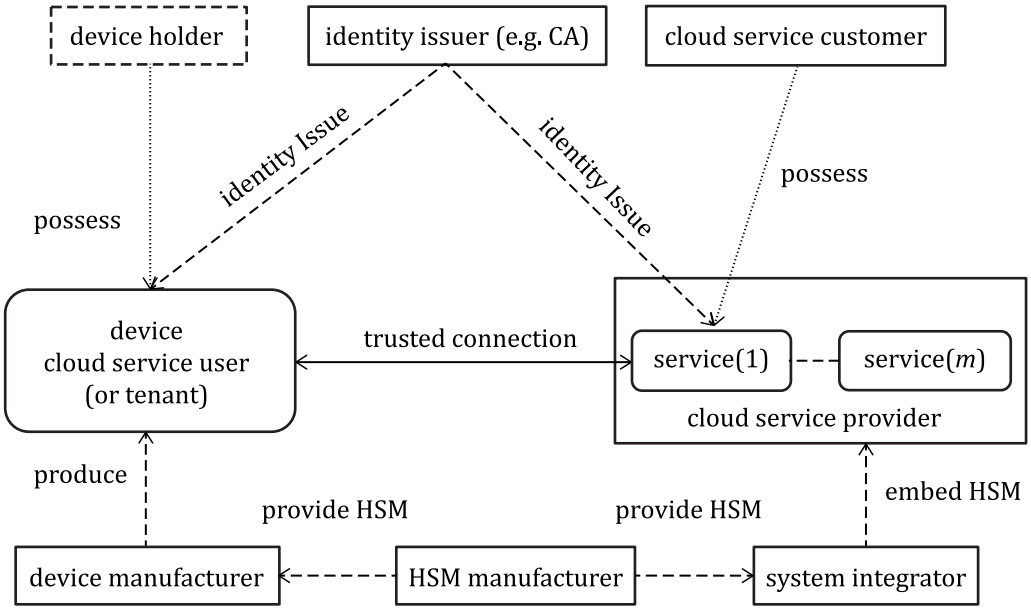
5.1 Overview

This clause provides an overview of the framework and components of a trusted connection between a device and a service based on hardware security modules.

A trusted connection between a device and a service provides the ability to protect confidentiality, integrity and authenticity of data; prevent identity spoofing by binding the identity of the device (or service) to a root of trust; and ensure trusted processing of data by remote attestation or environment assurance. For information on threats on a trusted connection, see Annex A.

Figure 1 describes the parties involved in establishing a trusted connection, including the identity issuer (e.g. CA), HSM manufacturer, device manufacturer, system integrator, cloud service provider, tenant, and device holder (it is possible that the party does in some scenarios such as IoT).

The HSM manufacturer produces HSMs. Device manufacturers produce a device. The cloud service provider runs the cloud service. The cloud service customer possesses the service which has a trusted connection with the device. In some scenarios, the cloud service customer and cloud service provider may be the same party. Devices act as the cloud service users (or tenants). Device holder (e.g. the holder of mobile phone) possesses and uses the device to establish trusted connection with a cloud service.



Key

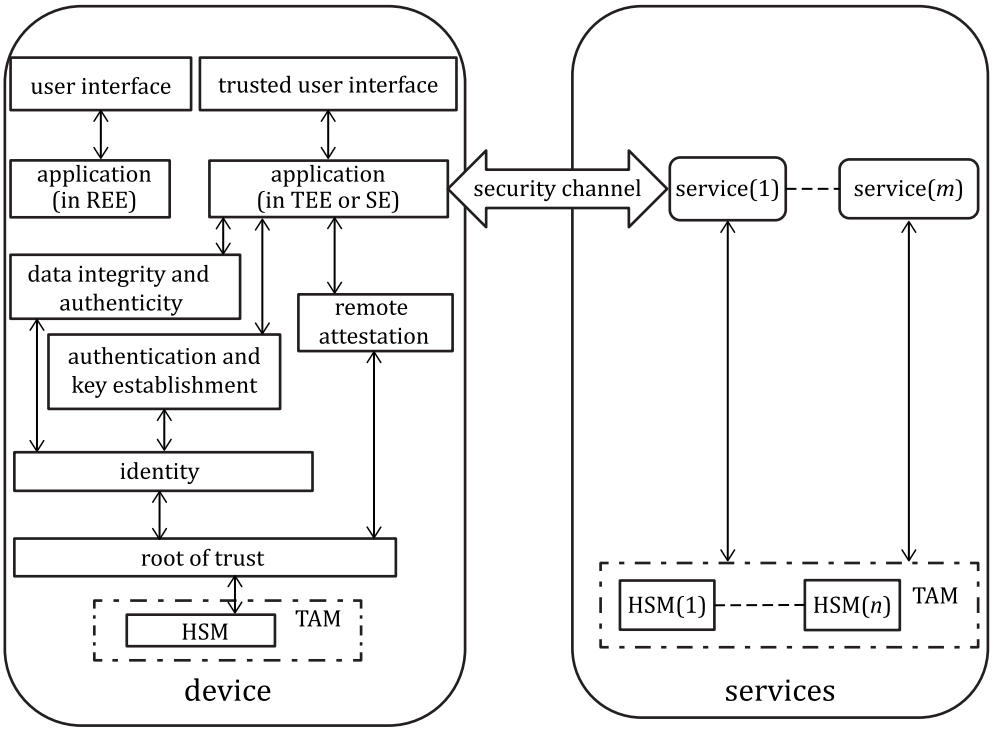
----->	offline or pre-set procedure	party (the party does not exist in some scenarios)	
·····>	ownership	party	
<---->	trusted connection	*m*	number of services

Figure 1 — Parties related in trusted connection

There are several scenarios to establish a trusted connection between a device and a service.

Figure 2 shows the framework of a trusted connection for device with both TEE/SE and REE (such as a mobile device). Applications which are run in a TEE/SE environment and have a root of trust based on the TAM, can build a trusted connection to service. A trusted user interface (TUI) component is provided for interaction between the user and the device.

Figure 3 illustrates the framework of a trusted connection for a device with the TE only (such as an IoT device). To establish a trusted connection between a device (with TE only) and a service, a remote attestation component may not be required, and the user interface (or trusted user interface, TUI) may not exist.

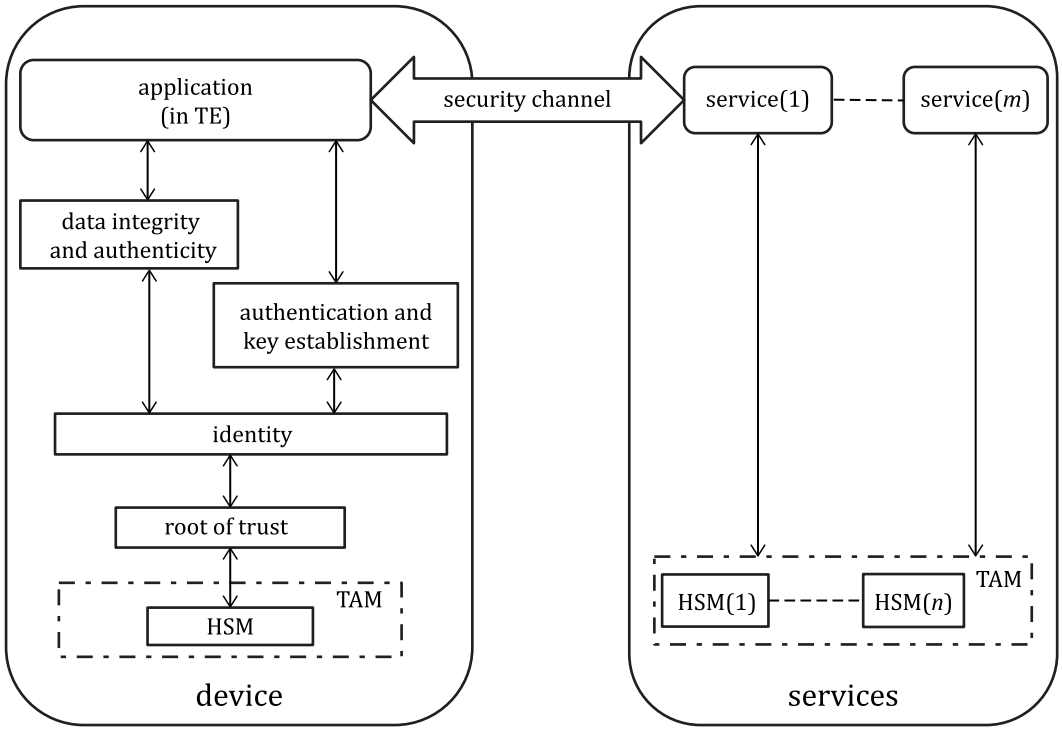


Key

⟷	data and control flow	▭	mandatory component
⬚	abstract component		
m	number of services	*n*	number of HSM

NOTE Trusted connection components in the service side are omitted.

Figure 2 — Framework of a trusted connection for a device with TEE/SE and REE



Key

⟷	data and control flow	▭	mandatory component
⬚	abstract component		
m	number of services	*n*	number of HSM

NOTE Trusted connection components in the service side are omitted.

Figure 3 — Framework of a trusted connection for a device with TE only

Figure 4 gives an overview of the components for a trusted connection.

Both the device and the service consist of multiple components. Each of these components performs a specific task within the trusted connection framework. The components to build a trusted connection are as follows:

— The HSM component safeguards and manages digital keys and provides cryptographic processing. A trust anchor module (TAM) is an abstract component that contains one or more HSMs.

— The root of trust component managing RoTs that are anchored in a specific HSM (e.g. TPM/TCM, TEE/SE) of the TAM.

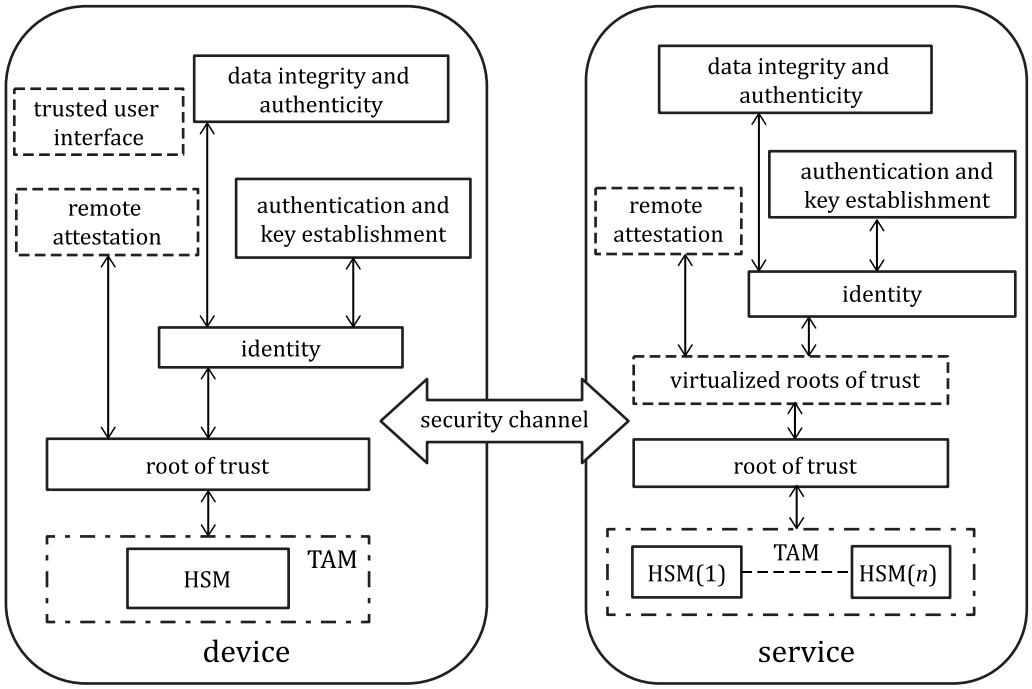
— The identity component manages identity bound to RoT. Trusted parties (including trusted third parties) issue identities to RoTs bound to the device (or service).

— The remote attestation component is responsible for remote attestation between the device and the service in a trusted connection. In some cases, if the device (or service) meets the corresponding security requirements (e.g. level 3 or greater as specified in ISO/IEC 19790), the remote attestation component in the device (or service) side is optional.

— The authentication and key establishment component is responsible for building a security channel between the device and the service based on RoT and IK.

— The data integrity and authenticity component is responsible for protecting the data integrity and authenticity by using cryptographic mechanisms with an IK. This component can also provide the non-repudiation property.

— The TUI component is responsible for the trusted interaction between the user and the device. In scenarios that do not require trusted interaction, there is no TUI component.



Key

⟷	data and control flow	▭	mandatory component
▭ (dashed)	optional component	▭ (dashed)	abstract component
n	number of HSM		

Figure 4 — Components of a trusted connection

The solutions for each component in the framework are contained in Annex B. An example of establishing a trusted connection between a device and a service is provided in Annex C.

5.2 Hardware security module

The hardware security module (HSM) is used for securing cryptographic keys and also provides cryptographic operations using these keys. The trust anchor module (TAM) is an abstract component that contains one or several HSM. The HSM provides protection for identities, applications and transactions by ensuring appropriate security of keys and encryption, decryption and authentication operations. The HSM has protection features such as physical tamper resistance.

5.3 Root of trust

The root of trust (RoT) is an anchor of trust which is based on HSM components that are inherently trusted. In a TPM case, there are three types of RoT, as described in ISO/IEC 11889-1:

— root of trust for reporting;

— root of trust for measurement;

— root of trust for storage.

In implementation, the different types of RoT can be executed by one or several HSM.

5.4 Identity

The identity component manages identities. Identity can be a certificate containing an identifier issued by a trusted party (including a trusted third party, such as PKI/CA). The identity is bound to the RoT on a device (or a service). Each RoT (or vRoT) can have an identity issued by a trusted party.

5.5 Authentication and key establishment

The authentication and key establishment component manage security channels between the device and the service. The device and the service execute a mutual authentication and key establishment protocol (such as TLS1.3[6]) using identity related keys to establish the security channels.

5.6 Remote attestation

The remote attestation is a method by which a device (or service) authenticates its hardware and software configuration to a service (or device).

The remote attestation component allows the device (or service) to convince the service (or device) that the platform has an embedded trustable HSM and the software configuration complies to the requirement of the service (or device). Anonymous remote attestation can be used in some scenarios to protect privacy. Usually, in a TPM case, remote attestation uses integrity measurement collectors (IMCs) to collect and sign device (or service) environment characteristics [e.g. integrity measurements in platform configuration registers (PCRs)], and integrity measurement verifiers (IMVs) to verify the integrity and authenticity of the device (or service) environment characteristics (e.g. integrity measurements from PCRs).

Remote attestation is executed between a device and a service to ensure the hardware and software set up meet the requirements of both sides. But in some scenarios, it is not critical for a service to execute remote attestation if the service meets the security assurance level.

5.7 Data integrity and authenticity

In order for a service (or a device) to verify the integrity and authenticity of data from a device (or a service), the device (or service) can protect the integrity and authenticity of the data, including parameters that describe the environment (such as algorithms used for data processing), by using cryptographic mechanisms to protect the data with an IK.

5.8 Trusted user interface

If the device requires interaction with the users (e.g. mobile phone), and there is a rich execution environment in the device, a trusted user interface (TUI) is used to help the user identify that the application on the device runs in a TE or in a REE.

6 Security recommendations for establishing a trusted connection

6.1 Hardware security module

The security recommendations for HSM are as follows.

— The HSM should be secure by design, and have certain physical security mechanisms to prevent unauthorized access to sensitive security parameters within the cryptographic module.

— The HSM should meet the corresponding security requirements for a cryptographic module (e.g. level 3 or greater as specified in ISO/IEC 19790).

— The HSM should be tamper-resistant and tightly integrated with the CPU and related hardware. This can be physical integration (typically in the case of small devices).

— The HSM should possess a unique identity key for each identity to be asserted.

— A HSM should be integrated in the device and with the service to establish a trusted connection.

6.2 Root of trust

The security recommendations for the RoT are as follows.

— The RoT should be derived from a specific HSM of TAM using cryptographic mechanisms.

— The RoT may be virtualized. If RoT is virtualized, ISO/IEC 27070 should be satisfied.

— The service side (especially cloud service) may have multiple physical roots of trust, and may have many virtualized roots of trust corresponding to single physical root of trust. If a virtualized RoT is used to establish a trusted connection, the virtualized RoT should be bound to a physical root of trust.

6.3 Identity

The security recommendations for the identity component are as follows.

— The identity of the device (service) should be bound to the root of trust.

— The device should contain one (or more) identity bound to a RoT (or virtualized RoT) to establish a trusted connection.

— The service should contain one (or more) identity bound to a RoT (or virtualized RoT) to establish a trusted connection.

— A trusted party (including a trusted third party) may issue certificates to each physical root of trust and each virtualized root of trust. Each RoT (including virtualized RoT) should have an identity (e.g. an X.509 certificate contains the identifier[5]) if the RoT is used to establish a trusted connection.

6.4 Authentication and key establishment

The security recommendations for the authentication and key establishment component are as follows.

— The device and the service should execute mutual authentication based on both IK.

— The device and the service should establish a session key to protect the security channel.

— The session key should be protected by the TE and should not be transferred to other device (or service).

6.5 Remote attestation

The security recommendations for the remote attestation component are as follows.

— If the device meets appropriate security requirements (e.g. ISO/IEC 19790 level 3 or greater), the remote attestation component in the device side is optional. Or else, the device should execute remote attestation to prove the environment of the device satisfy the requirement of the service.

— If the device side requires the environment proof of the service side to establish a trusted connection, the service should execute remote attestation to the device.

6.6 Data integrity and authenticity

The security recommendations for the data integrity and authenticity component are as follows.

— Data collected from device (e.g. sensors) should be protected using corresponding cryptographic mechanisms through an IK bound to the device. Thus, the verifier can verify the integrity and authenticity of the data.

— Data sent by the service to the device should be protected using corresponding cryptographic mechanisms through an IK bound to the service.

— In an application case where traceability and non-repudiation property are required, the data should contain a label to identify who (by identity bound to the RoT) created it, and the environment related parameters of the device (or the service) should be stored in an identity related data object. The data and the data object should be protected using corresponding cryptographic mechanisms through an IK.

6.7 Trusted user interface

The security recommendations for the TUI component are as follows.

— The device should have the TUI component if the device is equipped with user interface and the user interface is shared by REE and TEE/SE.

— TUI component should have physical switches or physical indicators to allow the user to distinguish TE from REE.

Annex A
(informative)

Threats

A.1 Threat examples from a user's perspective

A.1.1 Scenario 1: Traveller in an isolated place

A traveller in a remote village is defrauded by the adversary, where all the network and services are controlled by the adversary. The traveller has experienced a phishing attack.

A.1.2 Scenario 2: Fake data from IoT sensors

In an IoT system, some sensors can be controlled by the malicious code from the adversary. The service cannot distinguish the faked data generated by the malicious code from the data collected by the sensor.

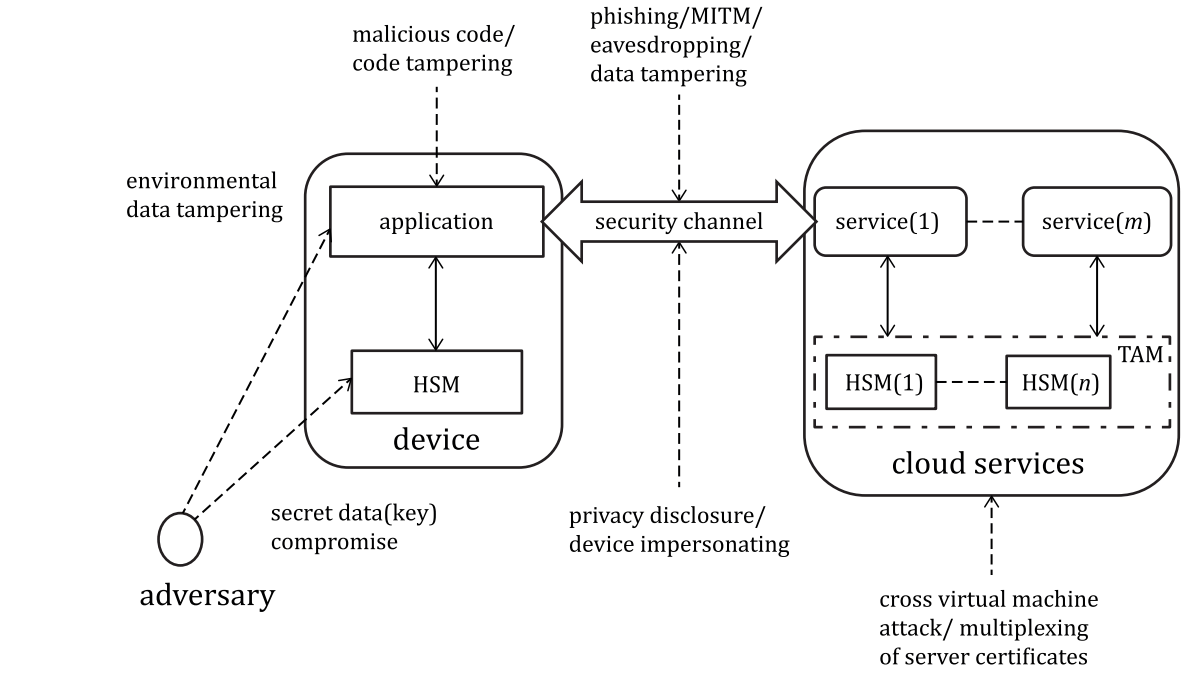
A.1.3 Scenario 3: Fictional data generated by AI

An AI entity can collect, edit and generate many kinds of data automatically, and the results (e.g. fictional image, fake article by automatic posting machine) are difficult to distinguish from the real data directly obtained and processed by the algorithms of the sensor. Currently, there is no reliable way to track the source of the data and the processing process.

A.2 Threat model

A.2.1 General

The threats faced by the trusted connection, as shown in Figure A.1, can be categorized into three types: threats against the device (A.2.2), threats against the service (A.2.3), and threats against channels (A.2.4) between the device and the service.



Key

---->	threats	⟷	data and control flow
▭	mandatory component	⬚	abstract component
m	number of services	*n*	number of HSM

Figure A.1 — Threats on trusted connection

A.2.2 Threats against the device

A.2.2.1 Malicious code attacks

— Description: The adversary injects code into the device or cheats the user to run a program with malicious code such as Trojans and virus.

— Consequences: The adversary runs the malicious code, obtains long-term secret data such as the key and credentials stored in the device, or gets the session data from the trusted connection.

— Possible solutions: The HSM in devices guarantees that an untrusted code cannot be executed, and the remote attestation ensures that the device satisfies the requirement of service.

A.2.2.2 Environmental data tampering

— Description: The adversary constructs a bypass between an environment data source (e.g. device sensor, cameras) and an application that uses the data.

— Consequences: The adversary injecting the environment data (e.g. biometric recognition information) to cheat the service.

— Possible solutions: Using an HSM in a device to build trusted environments and construct a trusted path from data sources to the application can prevent untrusted (spoofed) environmental data sources from running and prevent rivals from tampering with environmental data. Furthermore, the legitimate data processing (e.g. fingerprint feature extraction), can be recorded and included in the remote proof.

A.2.2.3 Code tampering

Description: The adversary modifies the trustee code running in the device. In contrast to malicious code attacks, the code tampering focuses on modifications to existing (trusted) application to enable them to perform actions required by the adversary, and the tampered codes usually do not have separate running entities. However, malicious code attacks can have its own (illegal) identity, which try to run by destroying the trusted environment detection mechanism.

Consequences: The tampered program is executed successfully and the adversary acquires the session content or secret data used within the program.

Possible solutions: Building a chain of trust based on the RoT and measuring the application can detect and block tampering with the trusted application code. In addition, an environment proof can enable the service to verify the integrity of the device (including hardware and software configuration), thereby preventing code tampering attacks.

A.2.2.4 Secret data (key) compromise

Description: The adversary obtains a legitimate device, attempts to obtain protected data stored in the device, such as keys and encrypted data. The adversary then performs the identity imitation or the crack attack.

Consequences: The adversary acquires long-term secret data, such as the secret key or the identity credential stored in the device, which means the adversary can masquerade as the device to the cloud service.

Possible solutions: By using HSM to store core keys, the device can resist crack attacks and protect secret keys and data.

A.2.3 Threats against the service

A.2.3.1 Multiplexing of server certificate

Description: The adversary uses the server certificate shared by multiple services to counterfeit other service and tries to establish a trusted connection to the device.

Consequences: The adversary successfully impersonates other services to communicate with the device, obtain data without permissions, or perform operations that do not have permission.

Possible solutions: In server certificates, an identifier is usually a domain name, which makes a multiplexing situation [such as multiple services running on the same content delivery network (CDN) server]. Each service holds a service identity that is bound to a RoT and can prevent such attacks.

A.2.3.2 Cross virtual machine attack

Description: In a cloud application scheme, the adversary may use virtual machines running on the cloud service to attack the access control vulnerabilities of the service.

Consequences: The adversary obtains the security privileges that were not previously available, such as obtaining secret data without access permission or performing an operation that is not privileged.

Possible solutions: Using the device identity that is bound to the RoT and building effective authorization solution can prevent this threat.

A.2.3.3 Environmental proof data forgery

Description: The adversary forges or replays the environmental proof.

Consequences: The adversary enables a device without a trusted environment to perform a cloud service operation that is not authorized.

Possible solutions: Constructing a chain of trust based on RoT and measuring the application can ensure the authenticity of the environment data and prevent the environment data from being forged. The replay of the environment data can be prevented by a remote attestation protocol.

A.2.3.4 Device impersonating

Description: The adversary uses a device that has the same model as a legitimate user or has similar environment properties.

Consequences: The adversary successfully forged into a legally identifiable device interacting with the service.

Possible solutions: Building a chain of trust based on the RoT in device's trusted environment and measuring the application can ensure the authenticity of the environment data and its binding to the RoT. Subsequent remote attestation binding with the device’s identity can prevent the environmental data and identity from being pieced together.

A.2.4 Threats against security channels

A.2.4.1 Phishing attack

Description: Entities other than cloud services interact with the device using message replay, forged certificate, or compromised certificate.

Consequences: The adversary pretends to be a service interacting with the device to obtain the user's secret data or perform an operation without permission.

Possible solutions: For well-designed phishing attacks, it is difficult for users to detect them. By using a service identity, the device can verify the authenticity of the service, thereby avoiding phishing attacks.

Social engineering and phishing: Insider attacks are on the rise, which are both intentional and non-intentional. Employees are often a weak link in the chain of security countermeasures. The goals of these attacks can vary from stealing intellectual property to causing physical harm. Employees are often used to get initial access to a network.

A.2.4.2 MITM attack

Description: The adversary forwards and tampers the message between the service and the device.

Consequences: The adversary pretends to be a service (device) communicating with the device (service), performs an unlicensed operation, or obtains a session key corresponding to the channels of trusted connection.

Possible solutions: Authentication of device and service identity, negotiation of session key and establishment of security channel can prevent a man-in-the-middle (MITM) attack.

MITM attacks are typically launched as a form of eavesdropping, whereby an attacker can tap into a conversation between two peers unbeknownst to the attacked endpoints. There are passive and active MITM attacks. With the latter, messages are typically modified. The absence of strong mutual authentication between the two communicating peers can lead to successful MITM attacks.

A.2.4.3 Eavesdropping attack

Description: The adversary eavesdrops the message between the service and the device.

Consequences: The adversary obtains the session key or the session data protected by the key in communication channels of trusted connection.

Possible solutions: The establishment of a secure channel can protect the confidentiality of communication data and prevent eavesdropping attacks.

A.2.4.4 Data tampering

Description: The adversary tampers the message between the service and the device.

Consequences: The adversary makes meaningful changes to session data without perception of the service and the device. For example, altering the parameters of an operation instruction.

Possible solutions: The establishment of a security channel can ensure the communication data is tamperproof and prevent potential data tampering attacks.

Annex B
(informative)

Solutions for components of a trusted connection

B.1 Hardware security module

There are several candidates that can be selected as HSM to establish a trusted connection, such as:

— The trusted platform module (TPM) which was written by trusted computing group (TCG) and standardized in the ISO/IEC 11889 series. The TPM can be embedded into a computing platform as a root of trust. TPM specification as contained in the ISO/IEC 11889 series gives the method of using a TPM chip to establish a trusted operating environment, as well as using a TPM chip key management and cryptography operations interface (signature, encryption, message digest, etc.).

— Trusted cryptography module (TCM) is one of the technologies which corresponds to TPM.

— Firmware-TPM (fTPM) is an end-to-end implementation of a TPM using Trust-Zone technology. fTPM provides security guarantees similar (although not identical) to a discrete TPM chip.

— A secure element (SE) is a tamper-resistant platform capable of securely hosting applications and their confidential and cryptographic data (e.g. cryptographic keys) according to the rules and security requirements set by well-identified trusted authorities.

B.2 Root of trust

B.2.1 General

Root of trust (RoT) is a component that performs one or more security-specific functions, such as measurement, storage, reporting, verification, and/or update. It is trusted always to behave in an expected manner, because its misbehaviour cannot be detected. In a practical environment, the functions of RoT may be executed by several HSM. In this document, RoT is based on HSM components that are inherently trusted, and can provide following functions:

— coot firmware protections;

— secure measurement of firmware;

— secure storage;

— device authentication;

— application and data isolation;

— remote attestation;

— data integrity and authenticity.

Depending on the functionality provided by the RoT, the RoT can be divided into root of trust for measurement (RTM), root of trust for reporting (RTR) and root of trust for storage (RTS), which can be implemented by single or multiple modules. To establish a trusted environment, the security of the key store and the cryptographic algorithm is protected by the physical root of trust.

B.2.2 Solution1: RoT in the device

The RoT can be directly constructed based on the HSM (TPM or CPU combined with cipher chip) embedded in the device.

B.2.3 Solution2: RoT in the service

For service, there may be multiple service instances running on a single physical server, which may not have enough physical HSM. To implement the binding of each service unique identity to the RoT instance, a vRoT can be used. Virtualized root of trust is based on the physical root of trust. Its computational security and IO security are protected by some security technology on the computing platform. Virtualized TPM is a kind of virtual root of trust and can be constructed both on TPM or TEE/SE module. ISO/IEC 27070 provides the security requirements for establishing vRoT.

B.3 Identity

B.3.1 General

In this document, identities are used to identify a device or a service instance. Identities can be categorized into several types by the identity issuer:

— identity issued by the manufacturer;

— identity issued by a third-party CA;

— identity generated by the cloud service provider.

The type of identity finally used is not confined by this document, but each identity used for trusted connection is bound to a RoT.

B.3.2 Identity generation

B.3.2.1 Identity issued by the manufacturer

Identity issued by the manufacturer is bound to a RoT and is generated during the HSM production procedure. Where a TPM is used, the identity corresponding to the attestation identity key of the TPM can be used as the identity for establishing a trusted connection. In other cases, the identity can be issued to a non-exportable key generate in the cipher module by signing a public key certificate.

B.3.2.2 Identity issued by a third-party CA

B.3.2.2.1 General

As HSM is built into a device and RoT is derived from HSM, CA (a trusted party or trusted third party) can issue an identity certificate for devices.

B.3.2.2.2 Solution 1: Device identity created by an internal CA

Device identity can be created from an internal CA. The following steps show an example:

— Device manufacturers can use internal CA issue certificates to the roots of trust.

— Device manufacturer stores the X.509 certificate[5] in the device.

B.3.2.2.3 Solution 2: Device identity issued by a third-party CA

Device identity can be created by a third-party CA after delivery. The following steps give an example:

— The identity of device should be bound to the roots of trust which are derived from the HSM.

— The device generates a certificate request for its root of trust, which is signed by the IK.

— The user sends the certification request to the third-party CA.

— The CA validates the certification request and issues a certificate to the device.

— The device stores the certificate.

B.3.2.3 Identity generated when the service is deployed

When a service is deployed, the server can create an identity that is bound to the service. Because the vRoT instance used by the service is based on the RoT with an HSM, it is possible to establish a trusted chain linked to the HSM, so the endorsement key (EK) of the vRoT can be used as a service identity, and a certificate can be issued to this key as well as information of the service. The following steps show an example:

— vRoT generates an identity key along with attestation for the linkage between the IK and its EK.

— The HSM deployed on the server uses its signing key linked to its EK to sign a signature (certificate) for the service's IK.

— The service instance can use the IK and the certificate to authenticate itself during trusted channel construction.

B.3.3 Identity revocation

For identity issued by the manufacturer and a third-party CA, the revocation can be done by revoking the corresponding certificate. For those devices using TPM, detailed steps of revocation are mentioned in the TPM specifications.

B.4 Authentication and key establishment

Secure authentication and key establishment can be implemented by using TLS 1.3[6] or other authenticated key agreement protocols that ensure authentication and key negotiation security.

B.5 Remote attestation

B.5.1 General

Remote attestation is an important step in establishing a trusted connection. By remote attestation, the device and the service can be mutually confirmed to:

— determine whether the communication target is trusted, that is, the availability of HSM, RoT and other credibility support;

— check the I/O of the platform application process and running state integrity;

— verify that the current running state of the platform conforms to the authenticator's security requirement.

In some cases, remote attestation can be used for validation of the operating environment and can include the platform (device) identity. The former can be attained by using the authentication key agreement protocol described in B.5. The latter can be supported by the solutions given in B.4.

B.5.2 Solution 1: Binary-based remote attestation

Binary-based remote attestation is the most basic proof method of platform integrity. In the binary-based remote attestation, the hardware security module digitally signs the platform integrity value represented by the binary hash value across the whole of the software or configuration data.

In device implementations, other than those with a tamper responsive envelope, the hardware security module (e.g. TPM) can only attest its own integrity. Outside the bounds of the hardware security module, various methods are used to implement the attestation calculations. These are then signed by the hardware security module, but that demonstrates only where the signature came from. Nothing is said about the integrity of the value that has been signed (that is, passed to the SE/TPM via an API call).

In the TPM specification, the remote attestation interface is already available for direct invocation, which can be invoked directly to implement binary-based remote attestation, which is signed by the platform attestation identity key. In addition, the prover can use the integrity measurement application running in a trusted environment to measure and generate metrics for the BIOS, BootLoader, operating system, and applications, and then call the cryptographic module to sign it to complete the proof. The sample procedure is as follows:

— RoT of measurement (e.g. BIOS module) firstly measures the platform status and computes the integrity value.

— The RoT uses a signing key linked to its identity to sign the integrity value and generate an attestation according to concrete specification.

— The verifier can verify the attestation.

— Strong attestation is possible by:

  — use of an approved strong cryptographic hash mechanism, and

  — use of shared secrets in the calculation or in the communication of that value between the prover and the verifier such that, replay and masquerade attacks are infeasible. By using a binary-based remote attestation, a verifier can verify the detailed running configuration of an offset entity. Since the attestation is signed by a signing key linked to a unique identity, the identity of platform and its environment are bounded, which prevents attacks such as face swap in remote face recognition.

More comprehensive measurement and attestation frameworks include integrity measurement architecture (IMA) and policy-reduced integrity measurement architecture (PRIMA). The IMA remote proof scheme is designed and implemented in conformity with the integrity proofs in the TCG specification and has become embedded in the operation system kernel. The IMA proof content includes platform integrity starting from the trusted computing platform, such as BIOS, BootLoader, operating system, and other applications.

B.5.3 Solution 2: Property-based attestation

The basic idea of a property-based attestation (PBA) is to convert a system attribute attestation request into an attribute logical expression of several components, which proves that each component satisfies the specific component attribute in turn. Attribute attestation is about some aspect of the device environment or operation that would not be evident from binary based attestation.

In most instances, PBA will use confidentiality and authentication measures to make the attestation, in which endpoint identity is shared. However, in some cases, the zero-knowledge proof method is used to prove that the commitment of platform component measurement results satisfies the attribute certificate requirements of component configuration. PBA using the zero-knowledge proof method can hide concrete platform configuration information to achieve more privacy protection. PBA can be used in the scenario of enhanced privacy requirement. Sample property-based attestation scheme using TPM and CL-proof scheme (proposed by Camenish and Lysyanskaya[7]) are described as follows.

— Firstly, the TPM undertakes and signs the component measurement results.

— Then, the host proves to the authority the integrity of component configuration with enhanced privacy.

— Finally, the verifier checks that the attribute is not revoked and the commitment signature is valid, with the assistance of the relevant authority of the property.

Typically, a property-based attestation scheme contains four main algorithms, which have the following purposes.

— Setup: to establish the public parameters of the system, and to issue the component attribute certificate.

— Attest: to generate evidence of the security attributes of the platform.

— Verify: to validate the component attribute.

— Check: to check that the security attribute has been revoked.

B.6 Data integrity and authenticity

Data integrity and authenticity are mainly used to ensure that the data obtained by the data source are not tampered with or altered without authorization (data integrity) and to prevent an adversary from forging sensor data (data authenticity). Attestation targets platform tampering detection while authentication is used to bind data transfers to source and destination parties and prevent in-channel manipulations of various kinds. The integrity and authenticity of the data can be guaranteed by HSM and trusted environment, and the reference scheme is described as follows:

— The data-source application should be run under a trusted environment protected by the HSM and the root of trust.

— After the data was generated or collected, the application requests a signature from the root of trust.

— The root of trust validates the request and signs the data using its private key; data may be divided into several blocks.

— When the data are being changed (e.g. image compression), the application requests a signature. The content of the signature should include the data and operation of the application.

Annex C
(informative)

Example of establishing a trusted connection

C.1 Establishing a trusted connection between the mobile device and the service

This annex describes an example where a mobile device has embedded a TPM, providing a trusted execution environment (TEE) and rich environment. The TPM acts as the root of trust and is used to uniquely identify the mobile device. The TEE delivers hardware isolation between the two operating systems, so that the secure operating system can provide a high level of assurance and security. The mobile device can securely access peripherals on the device.

After the mobile device is distributed to the holder, the mobile can generate its own unique private key and have unique identity issued from a CA.

While the mobile device starts to connect the service deployed by the cloud service provider, the mobile device and the service starts mutual authentication.

Then, the remote attestation is executed to ensure the integrity of the environment on the mobile device, so that the service can validate that an application and/or data are being delivered to the mobile device with specific properties.

After the trusted connection has been established between the mobile device and the service, the data to be delivered to the service is signed by the identity related key, in order to keep the integrity and authenticity of the data.

Bibliography

[1] ISO/IEC 11889 (all parts), *Information technology — Trusted platform module library*

[2] ISO/IEC 22123-1:2023, *Information technology — Cloud computing — Part 1: Vocabulary*

[3] ISO/IEC 19790, *Information technology — Security techniques — Security requirements for cryptographic modules*

[4] ISO/IEC 27000, *Information technology — Security techniques — Information security management systems — Overview and vocabulary*

[5] ITU-T X.509, *Information technology — Open Systems Interconnection — The Directory: Public-key and attribute certificate frameworks*

[6] RFC 8446, *The Transport Layer Security (TLS) Protocol Version 1.3*

[7] Camenisch J., Lysyanskaya A. *A Signature Scheme with Efficient Protocols*. 2002

ICS 35.030

Price based on 24 pages