HarmonyOS 2 Security Technical White Paper

Issue	**1.0**
Date	**2021-06-02**

HUAWEI

Contents

1 About This Document

Abstract

HarmonyOS is a brand-new operating system for smart devices. It is designed to provide a unified language for multi-device intelligence, interconnection, and collaboration, bringing a smooth, easy, and continuous interaction experience with reliability and security across all scenarios. Its typical technical features include:

- DSoftBus: Securely and reliably connects super devices.
- Virtual resource pool management for all devices: Allows any device or application to access cross-device resources through the distributed super device just as if it were using local resources.
- Distributed data management: Ensures centralized resource management on different devices, allowing any device or application to access cross-device files or data through the super device just as if it were accessing these locally.
- Distributed task scheduling: After conventional monolithic applications become service-oriented, the system builds a centralized distributed service management mechanism and supports remote startup, invocation, and connection as well as migration of applications across devices. As a result, right devices can be selected to perform distributed tasks based on device capabilities, location, running status, and resource usage, as well as user habits and intentions.

HarmonyOS benefits the entire ecosystem:

- For end users, HarmonyOS integrates various smart devices, implementing fast connection, capability collaboration, and resource sharing among them. This seamlessly transfers services to a right device for a smooth, all-scenario experience.
- For application developers, HarmonyOS uses distributed technologies to facilitate application development across various types of devices. Developers can focus on upper-layer service logic, making application development easier and more efficient.
- For device developers, HarmonyOS uses a component-based software design scheme to tailor itself to particular device forms based on their respective resource capabilities and service characteristics.

HarmonyOS is a convenient and efficient OS that benefits the entire ecosystem. However, it also requires a higher degree of user privacy and cyber security, especially in the following aspects:

- **DSoftBus:** As all devices are connected to one another to form a distributed HarmonyOS super device, a "trusted by default" security model is set up among them which is likely to lead to mutual pollution. Attackers can attack one of the devices and use it as a jump-off point to attack the other ones.

- **Distributed data management:** To ensure the seamless transfer of files and data between devices, a data protection mechanism is required for the whole system, rather than a single device. This increases the complexity of designing such a mechanism.
- **Intelligent atomic service/Distributed task scheduling:** Monolithic applications transform into distributed intelligent atomic services that can be invoked and run across devices, complicating application permission control, sandbox isolation, and other operations.

To meet these new security requirements, HarmonyOS proposes a security architecture based on the hierarchical security theory. This architecture is designed to "allow the right people to access right data through right devices" through a set of new, secure applications and a good, transparent ecosystem order. Consumers and developers enjoy new experiences in distributed collaboration along with strict privacy protection and data security.

This document describes the security technologies and functions of HarmonyOS 2, helping security practitioners understand the specific implementation of HarmonyOS. It also enables HarmonyOS developers to integrate the security capabilities provided by the HarmonyOS platform with their programs, ensuring the privacy and security of consumer data.

This document contains the following chapters:

- Chapter 1 About This Document: Introduces HarmonyOS, including its positioning, primary technical features, new value for the ecosystem, and security risks as well as their countermeasures.
- Chapter 2 HarmonyOS Overview: Briefly describes the technical aspects of HarmonyOS, including typical system architecture, technical schemes that differ from those used on traditional mobile and desktop operating systems, and major security risks.
- Chapter 3 HarmonyOS Security Theory Models: Describes the core security architecture models of HarmonyOS, including the access control model, BLP model for data privacy protection, and Biba model for system integrity protection.
- Chapter 4 Identity Management and Authentication for "Right People" in HarmonyOS: Describes the identity management and authentication mechanisms applied to subjects (which include developers, consumers, applications, and devices) throughout the lifecycle of applications. These mechanisms, based on the zero trust network architecture (ZTNA), constitute resilient security capabilities for the distributed HarmonyOS.
- Chapter 5 Hierarchical Security System Architecture for "Right Devices" in HarmonyOS: Describes the hierarchical security system of HarmonyOS, where corresponding security capabilities are provided for different types of devices with a consideration given to costs, risks, and capabilities. The contents cover chip, hardware, and kernel security, vulnerability anti-exploitation, secure isolation environment, formal verification, and other key technologies. Developers can flexibly select and assemble these technologies based on their system's security requirements.
- Chapter 6 Hierarchical Access Control Architecture for "Right Access" in HarmonyOS: Describes an access model set up based on the ZTNA and hierarchical security system architecture, which hierarchically manages users, applications, devices, and data in compliance with the requirements on access to sensitive personal data as well as those from security and privacy laws and regulations such as the GDPR. It also describes the BLP model (ensuring confidentiality) and Biba model (ensuring integrity) proposed by the hierarchical security theory.
- Chapter 7 HarmonyOS Ecosystem Governance Architecture: Describes the pure application- and device-oriented ecosystem governance architecture of HarmonyOS, which is designed to protect consumers' privacy and developers' interests to the

maximum extent. The content covers the concept of the least privilege, governance throughout the application lifecycle, and ecosystem construction for trusted HarmonyOS devices.

- Chapter 8 HarmonyOS Compliance with Security Standards and Certifications: Describes the compliance of HarmonyOS with privacy and security laws, regulations, and standards in various countries, and lists the mainstream security certifications obtained by HarmonyOS.
- Chapter 9 Typical Advanced Security Service Capabilities of HarmonyOS: Presents several examples (with specific services and scenarios) to describe how to build service systems with strict security for applications and services by using the security capabilities provided by HarmonyOS, thereby protecting consumers' privacy, property, and data to the maximum extent. The examples include Huawei Pay, secure keys, electronic ID (eID), and the car key.
- Chapter 10 Building the Resilient Security System Architecture of HarmonyOS: Describes how to build the resilient security system architecture of HarmonyOS, with reference to cutting-edge security architectures, such as the ZTNA and cyber resilience architecture. This chapter introduces the capabilities of HarmonyOS in security and trustworthy engineering, Singular Security Lab, bug bounty program, and security emergency handling process and mechanism. All these means serve to "minimize potential security vulnerabilities in the system, perform in-depth defense to make it hard to exploit vulnerabilities, and remediate vulnerabilities and restore services as soon as possible."
- Chapter 11 Ecosystem for Enabling and Opening Up HarmonyOS Security Capabilities: Introduces the basic security services, APIs, kits, and SDKs provided by HarmonyOS.

2 HarmonyOS Overview

2.1 Introduction to HarmonyOS

HarmonyOS is a brand-new operating system for smart devices. It is designed to provide a unified language for multi-device intelligence, interconnection, and collaboration, bringing a smooth, easy, and continuous interaction experience with reliability and security across all scenarios.

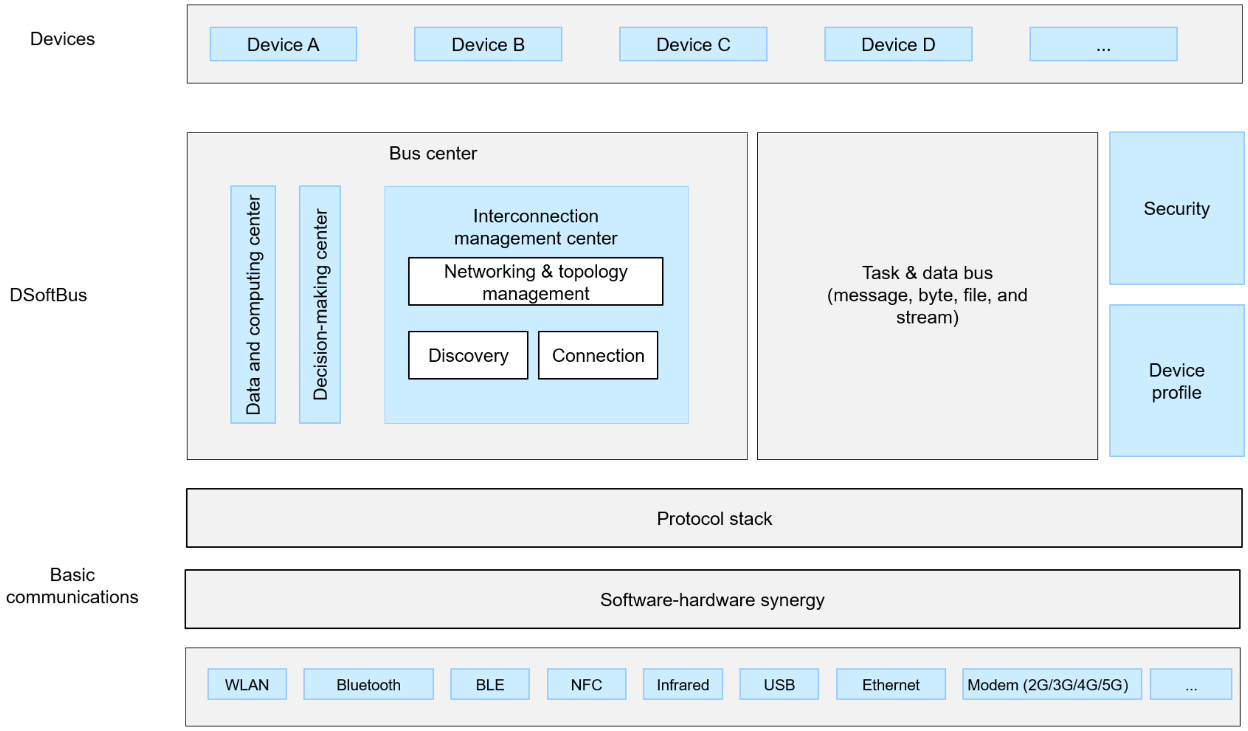
2.2 Technical Features of HarmonyOS

HarmonyOS is a distributed operating system tailored to the Internet of Things (IoT). It securely connects consumers' multiple devices, and builds an integrated distributed cross-device development platform. With HarmonyOS, different smart devices in distributed intelligent scenarios function as one whole, giving consumers the experience of using one super device. Beyond improving user experience, HarmonyOS also systematically helps developers become more efficient in multi-terminal scenarios.

DSoftBus

DSoftBus is a communication base for interconnecting devices, such as mobile phones, tablets, wearables, smart TVs, and head units. It powers devices with distributed communication capabilities, which allow devices to quickly discover and connect to each other, enabling efficient task distribution and data transmission. See Figure 2-1 Schematic diagram of DSoftBus.

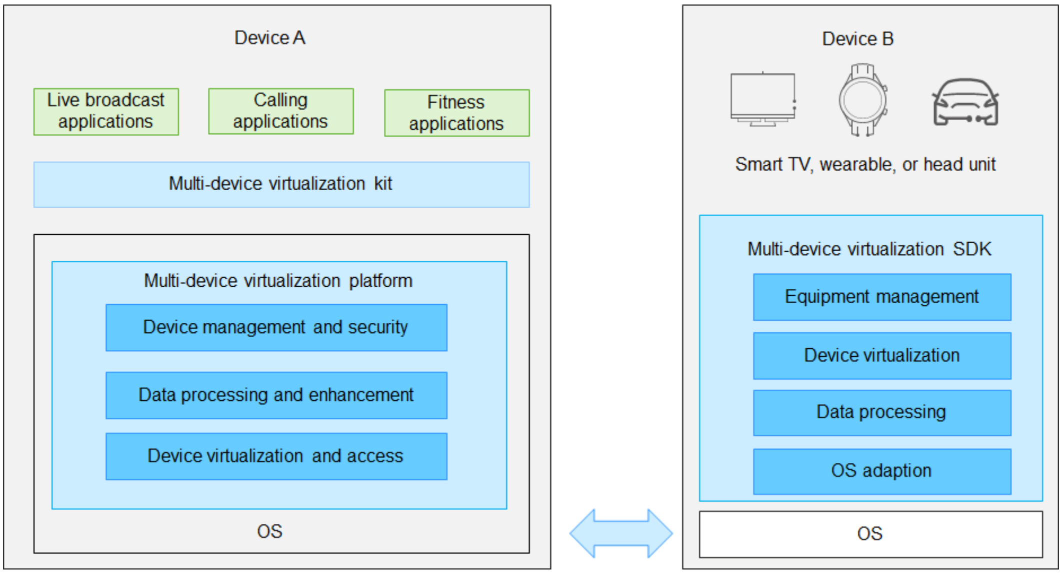
Figure 2-1 Schematic diagram of DSoftBus



Distributed Device Virtualization

The distributed device virtualization platform enables cross-device resource convergence, device management, and data processing so that multiple devices jointly function as a virtual super device. This platform virtualizes devices and fully utilizes their advantages by assigning the most appropriate hardware to execute particular user tasks. This ensures the continuity of services that have migrated between different devices. See Figure 2-2 Schematic diagram of distributed device virtualization.

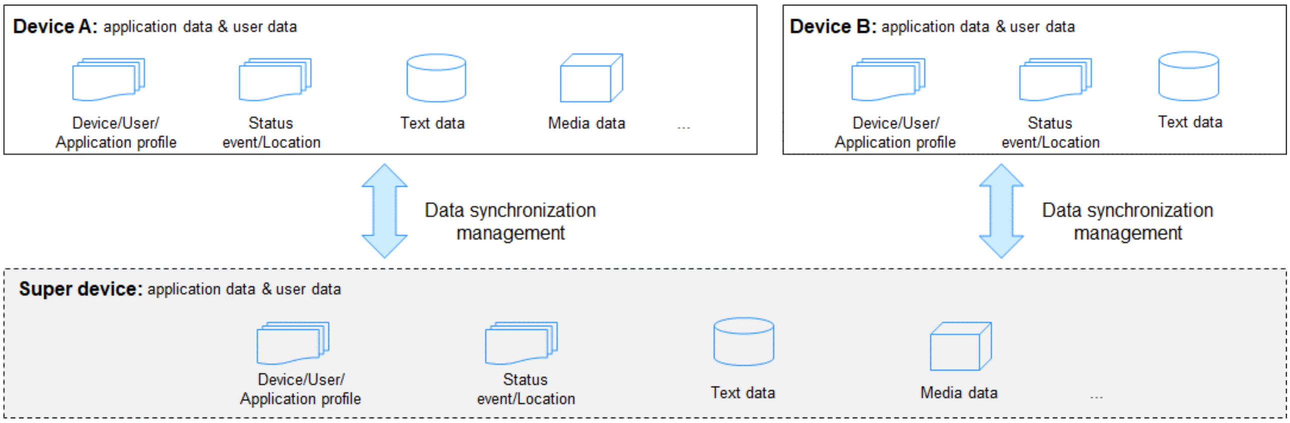
Figure 2-2 Schematic diagram of distributed device virtualization


Device A
Live broadcast applications
Calling applications
Fitness applications
Multi-device virtualization kit
Multi-device virtualization platform
Device management and security
Data processing and enhancement
Device virtualization and access
OS
Device B
Smart TV, wearable, or head unit
Multi-device virtualization SDK
Equipment management
Device virtualization
Data processing
OS adaption
OS

Distributed Data Management

Distributed data management uses the DSoftBus to manage the application and user data distributed on different devices. Under such management, user data is no longer bound to a single physical device, and service logic is separated from data storage. As applications are running across devices, their data is seamlessly transmitted from one device to another, which serves as the foundation for a consistent and smooth user experience. See Figure 2-3 Schematic diagram of distributed data management.

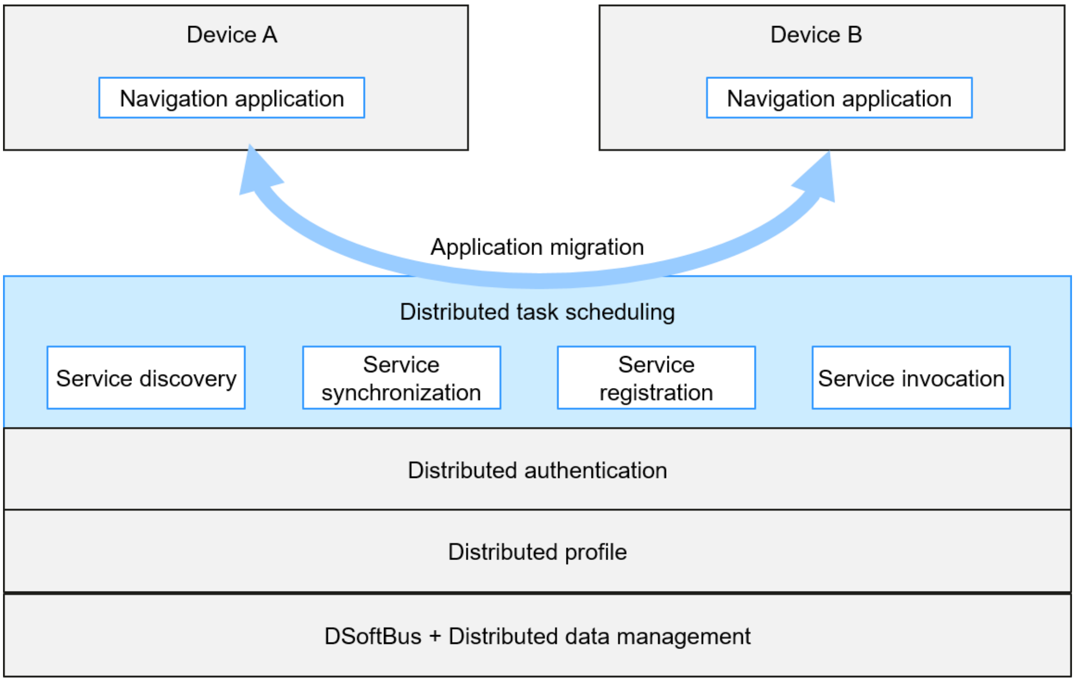
Figure 2-3 Schematic diagram of distributed data management


Device A: application data & user data
Device/User/ Application profile
Status event/Location
Text data
Media data
...
Device B: application data & user data
Device/User/ Application profile
Status event/Location
Text data
Data synchronization management
Data synchronization management
Super device: application data & user data
Device/User/ Application profile
Status event/Location
Text data
Media data
...

Distributed Task Scheduling

Distributed task scheduling is designed based on technical features such as DSoftBus and distributed data management. It builds an integrated distributed service management mechanism (including service discovery, synchronization, registration, and invocation), and supports remote startup, remote invocation, remote connection, and migration of applications across devices. In this way, applications can select a right device to perform distributed tasks based on the capabilities, location, running status, and resource usage of different devices, as well as user habits and intentions. See Figure 2-4 Schematic diagram of distributed task scheduling.

Figure 2-4 Schematic diagram of distributed task scheduling



2.3 HarmonyOS Security Risk Assessment

The following assesses HarmonyOS security risks based on the security risk assessment model (Risk = Asset x Threat) and the distributed architecture of HarmonyOS.

1. Key assets of HarmonyOS
   - Hardware and sensor resources after device resource pooling
   - Consumers' privacy-sensitive data
   - Data resources exclusively occupied by applications
   - Key data such as the device OS and firmware
2. Major threats to HarmonyOS
   - Personal privacy tracking and eavesdropping through the abuse of device resources, such as cameras, microphones, and location information
   - Personal data and privacy breaches caused by leaks in consumer data
   - Damage to the interests of developers, caused by application data leakage
   - Attacks on the operating system and firmware by tampering with program logic and data, implanting Trojan horses, and hijacking
3. Major security risks of HarmonyOS
   - **Risks due to imbalance in security capabilities of the distributed super device:** As all devices are connected to one another to form a distributed super device, a

"trusted by default" security model is set up among them which is likely to lead to mutual pollution. Attackers can attack one of the devices and use it as a jump-off point to attack the other ones.

- **Data security and privacy breach risks in distributed data management:** The distributed data management platform is designed to facilitate the seamless transfer of data and files between devices. However, it poses significant challenges to user privacy protection and the data security mechanism. Data security protection needs to be considered for the distributed system as a whole, which is quite different from security planning for traditional single devices. Any inadequacy of security protection may be exploited and become an entry point for attacks.
- **Risks associated with intelligent atomic service/distributed task scheduling:** Monolithic applications are changed into distributed intelligent atomic services that can be invoked and operated across devices, complicating the permission control, sandbox isolation, and other operations for applications.

3 HarmonyOS Security Theory Models

The Trusted Computer System Evaluation Criteria (TCSEC), better known as the *Orange Book* released by the USA Department of Defense in 1985, defines seven evaluation classes: D, C1, C2, B1, B2, B3, and A1. It has become the most widely accepted computer security rating criteria and method, with many countries integrating it into their national security standards.

With the development of security assessment technologies, the Common Criteria (CC) framework has been established, which provides a set of systematic security assessment standards and technical methods. Mappings can be set up between CC rating levels and TCSEC security classes. CC defines seven rating levels, EAL 1 to EAL 7, corresponding to the seven security classes, from D to A1, in the *Orange Book*.

Within the context of IoT, HarmonyOS connects all distributed devices, which in turn involves security and privacy protection for a large amount of user data and even the safety of personal life and property (such as the intelligent door lock service). In this sense, HarmonyOS requires a high security assurance level.

After a comprehensive evaluation of the system security, product usability, and user experiences, TCSEC B2/CC EAL 5 has been set as the goal for the HarmonyOS security architecture. The core security theory models of HarmonyOS use structured, hierarchical protection mechanisms, the two primary ones involved in the process of access to resource objects by subjects to resource objects being:

- Confidentiality model: Bell–LaPadula
- Integrity model: Biba

The following sections detail the security models of the HarmonyOS architecture.

3.1 Computer Security Evaluation Model

According to the *Orange Book*, computer security is classified as follows:

Class	Description
A1	Verified design. The security of systems in this class shall be strictly proven through formal techniques.

Class	Description
B3	Requires that user stations or terminals connect to network systems via trusted means, and hardware-based protection be provided for the storage area of the security system.
B2	Structured protection. Labels shall be assigned to all objects in the computer system, and security levels shall be specified for devices (such as home hubs, control devices, and IoT devices).
B1	Systems in this class support the multilevel security (MLS) model.
C2	Systems in this class enforce very detailed access control, for example, role-based access control (RBAC).
C1	Systems in this class have a certain protection mechanism in hardware and have the capability of full access control. However, they do not distinguish between permissions.
D1	Systems in this class do not authenticate users. This means that the systems can be used by anyone.

To match with the TCSEC security classes (D to A1), CC defines seven rating levels, EAL 1 to EAL 7.

EAL	Name	TCSEC
EAL 1	Functionally Tested	
EAL 2	Structurally Tested	C1
EAL 3	Methodically Tested and Checked	C2
EAL 4	Methodically Designed, Tested, and Reviewed	B1
EAL 5	Semiformally Designed and Tested	B2
EAL 6	Semiformally Verified Design and Tested	B3
EAL 7	Formally Verified Design and Tested	A1

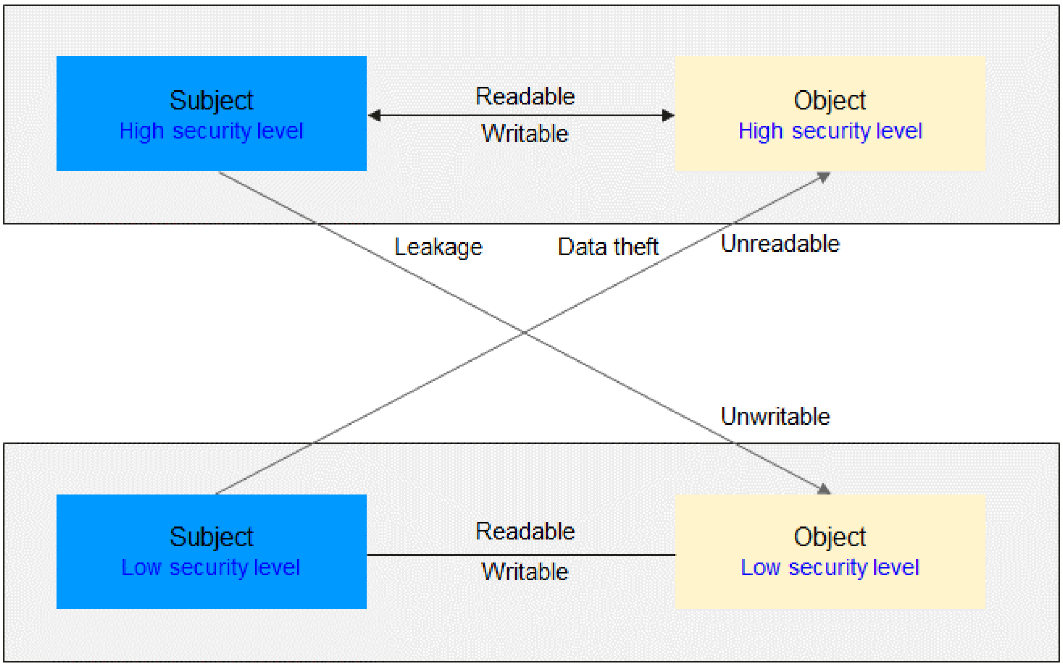
Among typical operating systems, MS-DOS is at level D, and Windows NT and UNIX are at levels C1 or C2. Systems in the B1 class use the MLS model and provide better protection for sensitive data, for example, the security level can be classified as confidential, secret, and top secret. Systems in the B2 class need to label all objects in the computer system, including subjects, environments, and objects, as well as protect confidentiality and integrity based on the strictly structured labels.

Aiming to assure protection for user data and privacy, HarmonyOS is designed to offer detailed protection for consumers' smart devices and protect key data from system attacks. HarmonyOS reaches B2-level security across the system; it uses B3-compliant dedicated security chips and processors to store and process key data such as biometric information for consumer authentication and payment, eID, and secure key data, and uses formal techniques to prove the security of the critical trusted execution environment (TEE) OS in compliance with the requirements of the A1 class.

3.2 BLP Model for Confidentiality Protection

In 1973, David Elliott Bell and Leonard J. LaPadula formalized the access control rules that are applied in military applications into a model named Bell–LaPadula (BLP model for short).

The following figure shows the access control model:



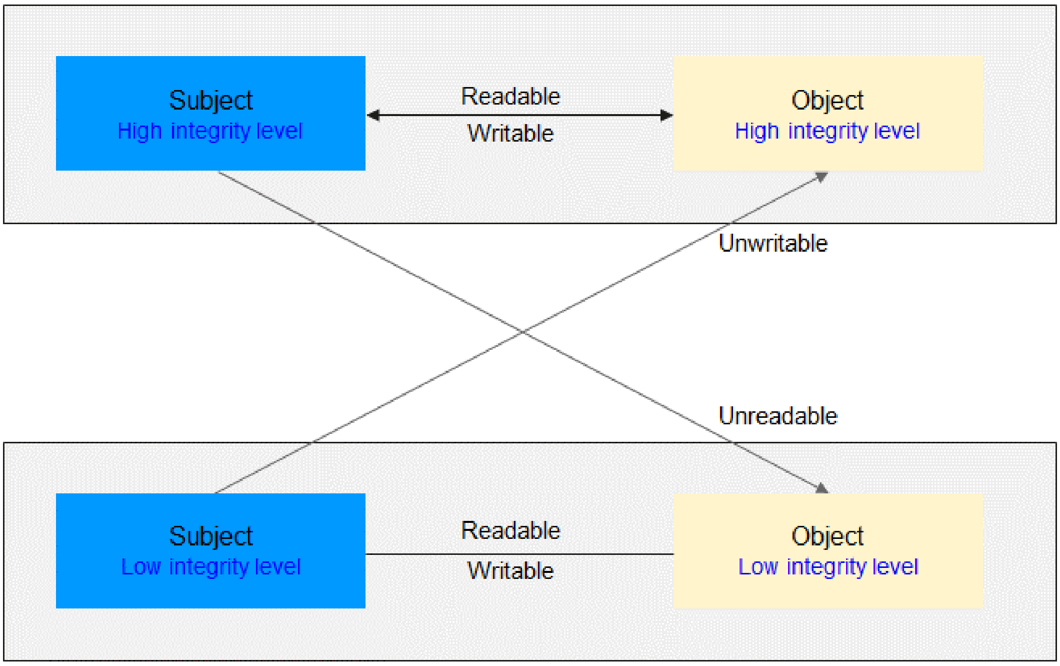
The BLP model has the following rules:

- No read up: A subject at a given security level may not read an object (data) at a higher security level.
- No write down: A subject at a given security level may not write to any object (data) at a lower security level.

HarmonyOS is intended to strictly implement the access control principles of the BLP model for confidentiality protection and ensure that user data and privacy are not leaked, high-security data is not leaked by high-security devices to low-security devices without consumers' awareness, and low-security devices are unable to obtain high-security data.

3.3 Biba Model for Integrity Protection

The BLP model mathematically proves that information privacy can be guaranteed, but it does not address data integrity. Therefore, in 1977, Ken Biba proposed the Biba model.



The Biba model has the following rules:

- No read down: A subject at a given level of integrity must not read data at a lower integrity level.
- No write up: A subject at a given level of integrity must not write to data at a higher level of integrity.

HarmonyOS is intended to strictly implement the access control logic defined by the Biba model. Applications, software, upgrade packages, and patches from untrusted sources are prevented from being installed on high-security devices. Only signed software that is officially recognized by HarmonyOS can be imported into the operating system. In addition, a low-security device is prohibited from sending a control instruction to a high-security device, such as a sports watch controlling a mobile phone to make large payments.

HarmonyOS adopts the security architecture models to reach TCSEC B2. It applies structured protection schemes and strictly assigns security labels to the subjects (developers, applications, natural persons, and devices), environments (IoT devices running HarmonyOS and network environments), and objects (data, files, and peripherals) in the system.

To make the structured security models take effect in the HarmonyOS security architecture, all subjects, environments, and objects must be trustworthy; and at the same time, authenticity, integrity, and being tamper-proof must be ensured for the security labels assigned to the said subjects, environments, and objects. Then, it is possible to "allow the right people to access right data through right devices". The following sections describe the models that implement this process.

3.4 Right People: Model for Subjects

HarmonyOS involves four types of subjects, and the system uses corresponding identity authentication measures for each one:

- Developers: Real-name authentication is performed on the HarmonyOS developer website to ensure that developers take corresponding responsibilities and obligations, and receive the corresponding rights and benefits.
- Consumers: HarmonyOS provides multiple authentication options to authenticate consumers (natural persons), such as the use of the PIN code, fingerprint, face, voiceprint, certificate, and real-name authentication, ensuring that the system is not deceived when the device is lost or when an attacker impersonates a consumer for authentication.
- Applications: All applications running on HarmonyOS are signed in to AppGallery to prevent fake or forged applications from running on HarmonyOS.
- Smart atomic services: HarmonyOS has a strict definition of permissions for each smart atomic service.

3.5 Right Devices: Model for Access Environments

After ascertaining the identities of subjects, HarmonyOS needs to ensure that it is running on a trusted hardware device that is right for service implementation. HarmonyOS provides the following capabilities to ensure IoT device security:

- Trusted device sources: All devices in the HarmonyOS ecosystem must have the security capabilities defined by HarmonyOS. The devices that pass the security check will receive a HarmonyOS operations platform certificate, officially signed by Huawei, as a proof of their security capabilities and level so as to ensure the trustworthiness of the device sources.
- Matching between device security levels and data privacy requirements: The security capabilities of IoT devices shall match the security and privacy requirements of the services and data running on the devices. Devices with low security levels are not allowed to process highly sensitive data, and strict security rating criteria must be complied with.
- Device authentication: In distributed, trusted interconnection, all devices that comprise the super device are allocated with signed identity certificates in advance. The certificates are the basis for device authentication, authorization, and signing, and their use ensures the confidentiality, integrity, and non-repudiation of the data, program, and instructions transferred between devices running HarmonyOS.
- Trustworthiness of device systems: In HarmonyOS, products of all series are required to support trusted boot and trusted operation as well as implement integrity protection throughout the product lifecycle to prevent devices from being tampered with.

3.6 Right Access: Model for Access Control

HarmonyOS strictly manages data by security level and label. During service data processing, it strictly complies with the BLP and Biba models for confidentiality and integrity protection to achieve structured protection for the whole system.

HarmonyOS strictly defines and classifies data based on security requirements defined in GDPR, GAPP, China's Personal Information Protection Law, and other related laws and regulations and manages it by security level and label.

International standards (FIPS 199 and NIST SP 800-122) are used as reference for data classification; Huawei's corporate standards and industry best practices are used as reference for data privacy categorization.

The data security classification criteria of HarmonyOS are as follows:

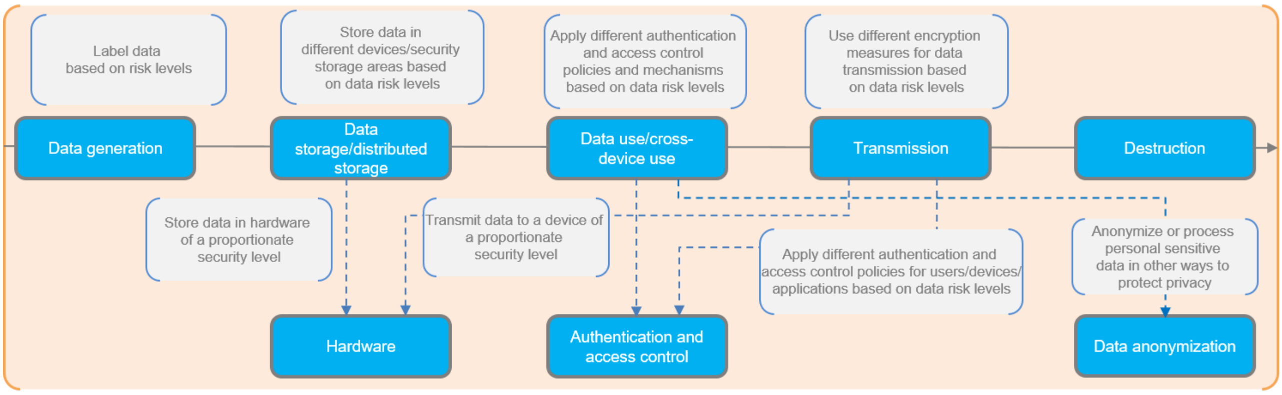
- Critical: special data types defined in industry laws and regulations, involving the most private personal information, or data that may cause significant negative impact on individuals or organizations if leaked
- High: data that could cause a serious negative impact on individuals or organizations if leaked
- Medium: data that could cause significant negative impact on individuals or organizations if leaked
- Low: data that could cause limited negative impact on individuals or organizations if leaked
- Public (no risk): data that can be made public without any adverse impact on individuals or organizations

Data Privacy Category	**Data Type**	**Data Security Class**	**Example**
Sensitive personal data	Identity authentication credential	Critical (S4)	Tokens and passwords used for identity authentication
	Personal racial and ethnic information		Racial or ethnic origin
	Negative reputation data		Negative records such as criminal records and disciplinary actions
	Health information		Data relating to body fat, blood pressure, blood glucose level, heart rate, blood oxygen level, ECG, medical records, sex activities, and sleep patterns
	Biometric features		DNA, fingerprint, facial features, iris, voiceprint, palm print, ear pinna, and behavioral features
General personal data	Workout data	High (S3)	Step count, workout distance, workout duration, calories consumed, climbing height, oxygen intake, running posture, and heart rate
	Personal multi-media data		Images, texts, audios, and videos on user devices
	Age and birth date	Medium (S2)	Age and birth date

Data Privacy Category	**Data Type**	**Data Security Class**	**Example**
	Social user identifiers		Socially recognizable user identifiers, such as Huawei accounts and social media app accounts, which can be discarded, replaced, and re-registered
	Name and nickname		Name and nickname
	Address information		Postal code, work address, and home address
	Basic personal information	Low (S1)	Gender, nationality, place of birth, education, professional background, etc.
	Positive reputation data		Professional achievements
Non-personal data	System key	High (S3)	System root key, working keys derived using the root key for encrypting system services and applications, and working keys generated by applications for encrypting system services and applications
	Miscellaneous non-personal data	Low/Open (S0)	Publicly released data attached to system and device information, such as software version number, engine version number, client version number, driver version number, SDK version number, and application category

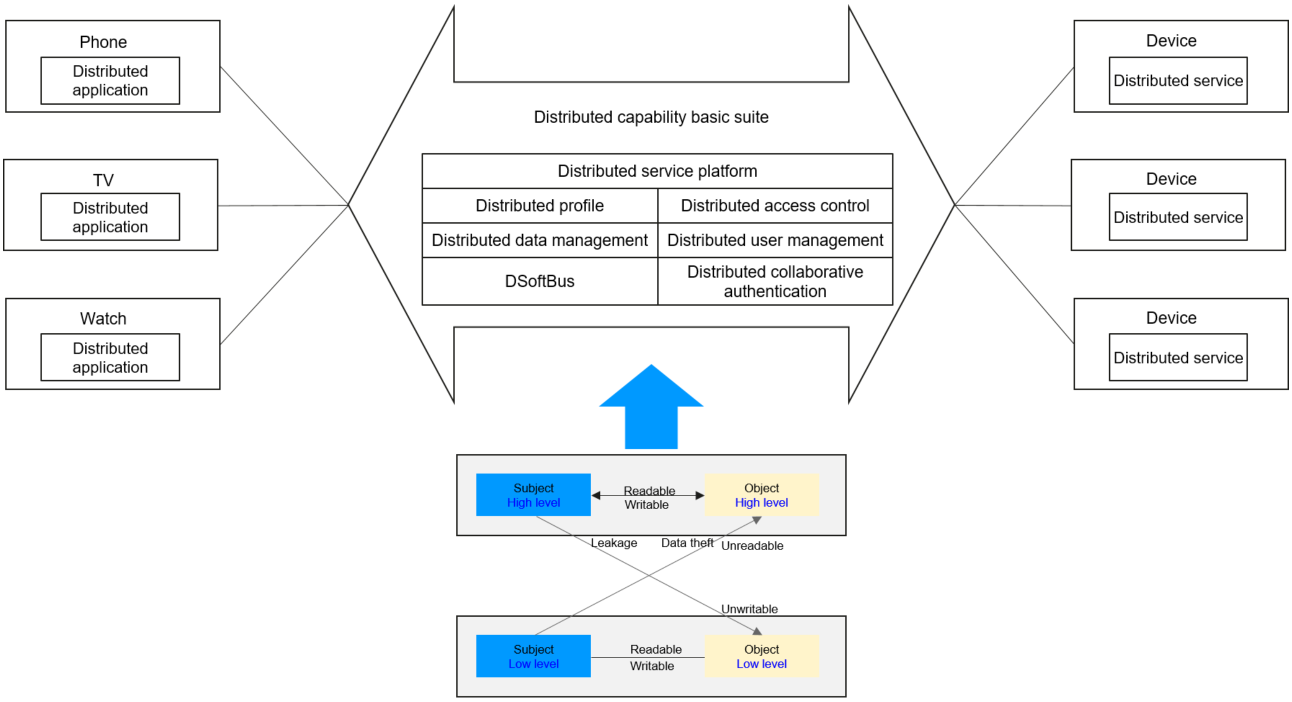
On the basis of data security classification, access to data is strictly controlled throughout the lifecycle of the data.

Figure 3-1 Data access control throughout the data lifecycle


Label data based on risk levels
Store data in different devices/security storage areas based on data risk levels
Apply different authentication and access control policies and mechanisms based on data risk levels
Use different encryption measures for data transmission based on data risk levels
Data generation
Data storage/distributed storage
Data use/cross-device use
Transmission
Destruction
Store data in hardware of a proportionate security level
Transmit data to a device of a proportionate security level
Apply different authentication and access control policies for users/devices/applications based on data risk levels
Anonymize or process personal sensitive data in other ways to protect privacy
Hardware
Authentication and access control
Data anonymization

Based on the classification and categorization of users, devices, services, and data, HarmonyOS implements distributed access control:

Figure 3-2 Distributed access control



4 Identity Management and Authentication for "Right People" in HarmonyOS

In addition to traditional identity authentication methods involving numerical and graphical passwords, HarmonyOS also provides biometric authentication methods such as fingerprint and facial authentication. Based on their respective security capabilities and features, these methods can be applied to a range of diverse scenarios such as device or application locking and mobile payment.

HarmonyOS also offers distributed collaborative authentication, enabling users in distributed service scenarios to perform authentication through the most convenient device.

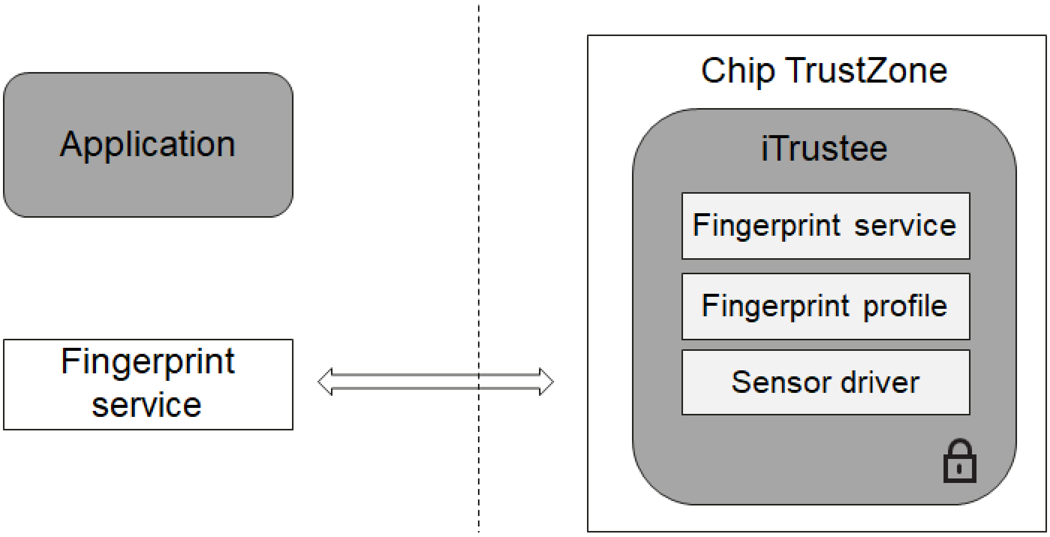
4.1 Biometric Authentication

Fingerprint Authentication

Currently, HarmonyOS supports capacitive, optical, and ultrasonic fingerprint authentication. The experiences brought by and security capabilities of the three fingerprint authentication technologies are basically the same. Different devices can select one of them for fingerprint authentication according to their own product positioning.

The following figure shows the HarmonyOS's fingerprint authentication security framework.

Figure 4-1 Fingerprint authentication security framework



HarmonyOS establishes a secure channel between the fingerprint sensor and iTrustee. Fingerprint images are transmitted to iTrustee over the secure channel. HarmonyOS extracts features, detects liveness, and compares features in iTrustee, and performs security isolation based on the TrustZone. The fingerprint framework of the rich execution environment (REE) is only responsible for fingerprint authentication initiation and authentication result data. It does not handle the fingerprint data.

Fingerprint feature data is stored in iTrustee's secure storage. Data encryption and integrity protection are implemented using high-strength cryptographic algorithms. The key for encrypting fingerprint data cannot be obtained externally, which ensures that fingerprint data cannot be leaked. No external third-party applications can obtain fingerprint data or transfer such data outside of iTrustee. HarmonyOS does not send or back up any fingerprint data to any external storage media including the cloud. After fingerprint features are stored (for template input) or compared (for identity authentication), fingerprint images are destroyed.

The probability of fingerprints that are not enrolled passing authentication is about one in fifty thousand. To provide extra protection, HarmonyOS's fingerprint authentication supports an anti-brute force cracking mechanism. If the fingerprints of a user fail to be recognized five consecutive times in the screen-on state or 10 consecutive times in the screen-off state, fingerprint authentication will be disabled for 30 seconds. If a user fails the fingerprint authentication 20 consecutive times, the user must enter the password to unlock the device.

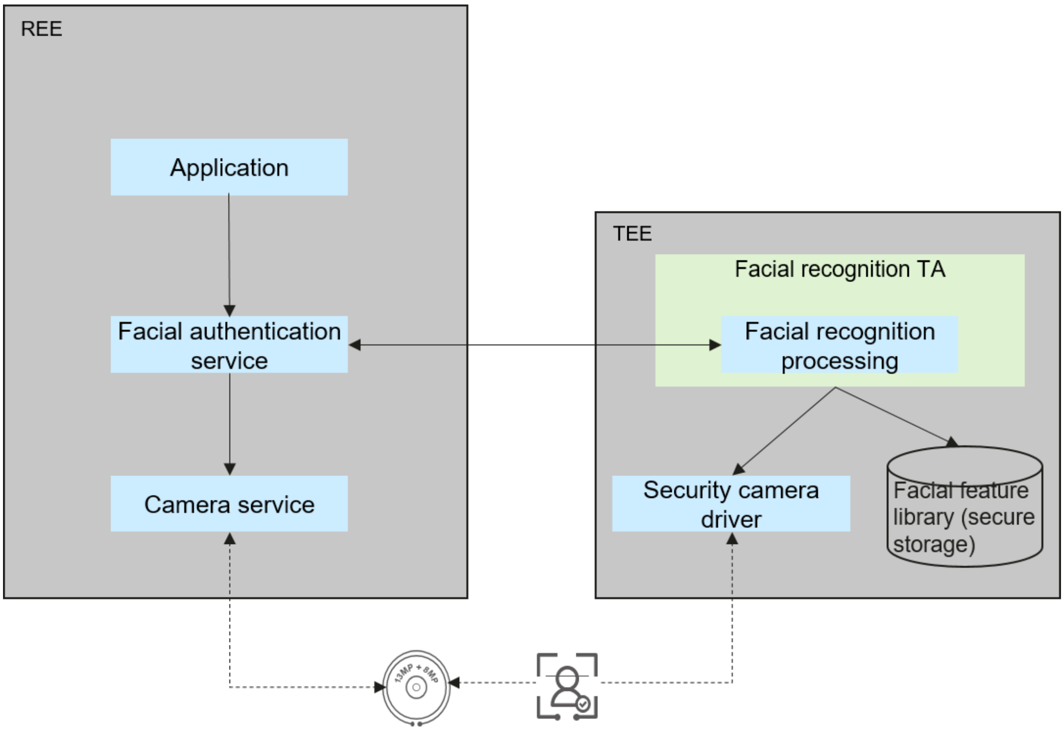
Fingerprint authentication facilitates identity recognition, but users may be more likely to forget their lock screen passwords. If a HarmonyOS user does not use his/her unlock password within 72 hours, the user has to enter the password when unlocking the screen. This is designed to reduce the likelihood of a forgotten password.

Facial Authentication

HarmonyOS provides both 2D and 3D facial authentication solutions. The 3D facial authentication solution depends on a special depth camera, and 2D facial authentication is implemented via a common front-facing camera. The authentication rate and anti-counterfeiting capability of 3D facial authentication are much better than those of 2D facial authentication. 3D facial authentication can be applied to payment scenarios, whereas 2D facial authentication cannot. Different terminals can select one of them for facial authentication according to their own product positioning.

The following figure shows HarmonyOS's facial authentication security framework (a general framework for some Qualcomm and MTK platforms):

Figure 4-2 Facial authentication security framework


REE
Application
Facial authentication service
Camera service
TEE
Facial recognition TA
Facial recognition processing
Security camera driver
Facial feature library (secure storage)
13MP + 8MP

HarmonyOS establishes a secure channel between the camera and iTrustee. Face images are transmitted to iTrustee over a secure channel. HarmonyOS extracts features, detects liveness, and compares features in iTrustee, and performs security isolation based on the TrustZone. The external facial framework is only responsible for facial authentication initiation and authentication result data, and does not handle facial data.

Facial feature data is stored in iTrustee's secure storage. Data encryption/decryption and integrity protection are implemented using high-strength cryptographic algorithms. The key for encrypting facial feature data cannot be obtained externally, which ensures that facial feature data cannot be leaked. No external third-party application can obtain facial feature data or transfer such data outside of iTrustee. HarmonyOS does not send or back up facial data (either encrypted or unencrypted) to any external storage media including the cloud.

The probability of faces that are not enrolled passing authentication is about one in a million for a 3D solution and between one in fifty thousand and one in one hundred thousand for a 2D solution. HarmonyOS's facial authentication supports an anti-brute force cracking mechanism. If the face of a user fails to be recognized five consecutive times, the user must enter his/her password to unlock the screen. Facial authentication may not work as effectively for twins, siblings with similar looks, and children under 13 years of age.

In addition, because facial authentication is mainly based on facial image data collected by a camera, it may be impossible to accurately distinguish a recaptured photo or a well-made head model.

If users are concerned about the preceding risks, they can use password authentication.

4.2 Distributed Collaborative Authentication

HarmonyOS provides distributed identity authentication capabilities between trusted devices that constitute a distributed system. It breaks device boundaries and provides flexible identity authentication capabilities based on user operations and service requirements. When a user operates multiple trusted devices on the same local area network (LAN), the user can use the most handy device with the same security level as the access and identity authentication portal.

The collaborative user identity authentication (hereinafter referred to as collaborative authentication) of HarmonyOS is based on traditional secure data channels between trusted devices and a distributed user identity authentication framework.

- Distributed authentication based on what the user knows (protected information)

  If trust relationships are established between HarmonyOS devices, the end collecting the lock screen password can be decoupled from the authentication end on these devices to support distributed authentication based on that password. The collection end anonymizes the user's collected lock screen password, while the authentication end compares the authentication credentials. The two ends work together using a password authenticated key exchange (PAKE) protocol to complete distributed authentication. In this way, remote authentication is carried out without the need to transmit the user's protected information to the peer end.

  To ensure that the information on the collection and authentication ends comes from authentic security modules, HarmonyOS's distributed authentication service generates an identifier (an Ed25519 public/private key pair) in the iTrustee of both the collection and authentication ends. During remote authentication based on the user′s protected information, the identifier information is used to sign the local data to be sent to the peer and verify the data received at the local end.

  After the lock screen password is collected and anonymized on device A, a PAKE protocol field is generated in the iTrustee and signed with a collector identifier (a private key). The information is then sent to device B through a secure E2E encryption channel based on the trust relationship between the two devices. Finally, the signature is validated in the iTrustee of device B, completing the authentication based on the PAKE protocol.

  During this process, the REE cannot tamper with the signature. In the PAKE-based authentication mechanism, the plaintext lock screen password and intermediate computing results are not transmitted between devices, and the password cannot be reversely deduced from the transmitted protocol authentication field.

  An anti-brute force cracking mechanism is supported in HarmonyOS's collaborative authentication using the lock screen password. This mechanism functions similar to that protecting the local lock screen password.

- Distributed authentication based on what the user has (trusted possessions)

  A Bluetooth accessory connected by the user can serve as a "trusted possession" for user identity authentication. HarmonyOS provides multi-factor authentication as an enhanced feature for collaborative authentication. When a Bluetooth accessory is connected to a mobile phone and is worn, the mobile phone delivers a wearing token to the accessory after the user unlocks the screen on the mobile phone. If the Bluetooth accessory is disconnected or removed by the user, the token becomes invalid and cannot conduct continuous wearing detection with the mobile phone.

  When the Bluetooth accessory is used to initiate user authentication, the collaborative authentication service determines its connection status and distance with the mobile phone and also initiates the continuous wearing detection between it and the mobile phone based on the wearing token. The authentication is considered successful only when all the three factors meet the requirements.

5 Hierarchical Security System Architecture for "Right Devices" in HarmonyOS

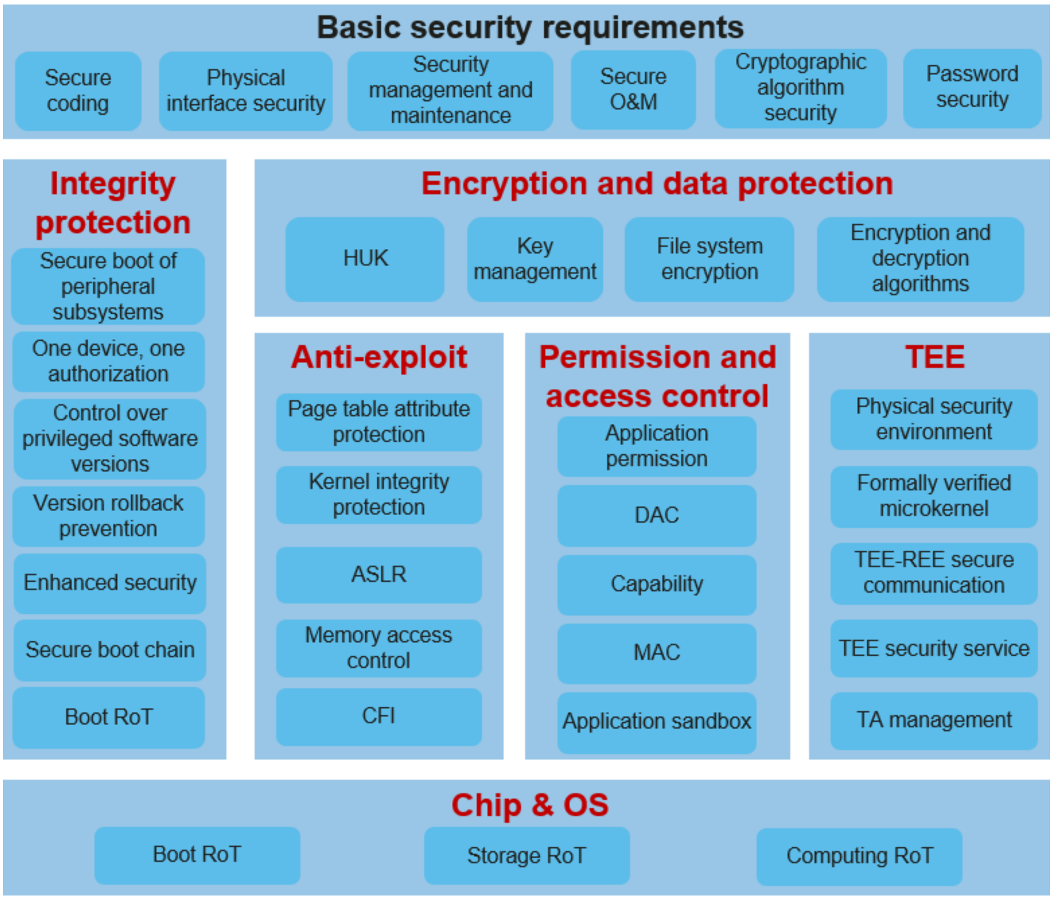
HarmonyOS provides a reference architecture for hierarchical system security based on a wide range of security classification models, including the TCSEC (known as the Orange Book), CC, Federal Information Processing Standards (FIPS) on security requirements for cryptographic modules, and the security classification model of computing devices defined by the Internet of Things Security Foundation (IoTSF). The reference architecture provides the basis on which the security classification specifications for HarmonyOS devices are formulated. This chapter describes typical security technologies in the HarmonyOS system security architecture and security classification requirements.

5.1 HarmonyOS Device Security Classification Specifications

System security in HarmonyOS is built on three hardware roots of trust (RoTs): boot RoT, storage RoT, and computing RoT. Leveraging its basic security engineering capabilities, HarmonyOS integrates security technologies and capabilities that focus on device integrity protection, data confidentiality protection, and attack defense.

Figure 5-1 shows the typical security architecture of individual devices running HarmonyOS.

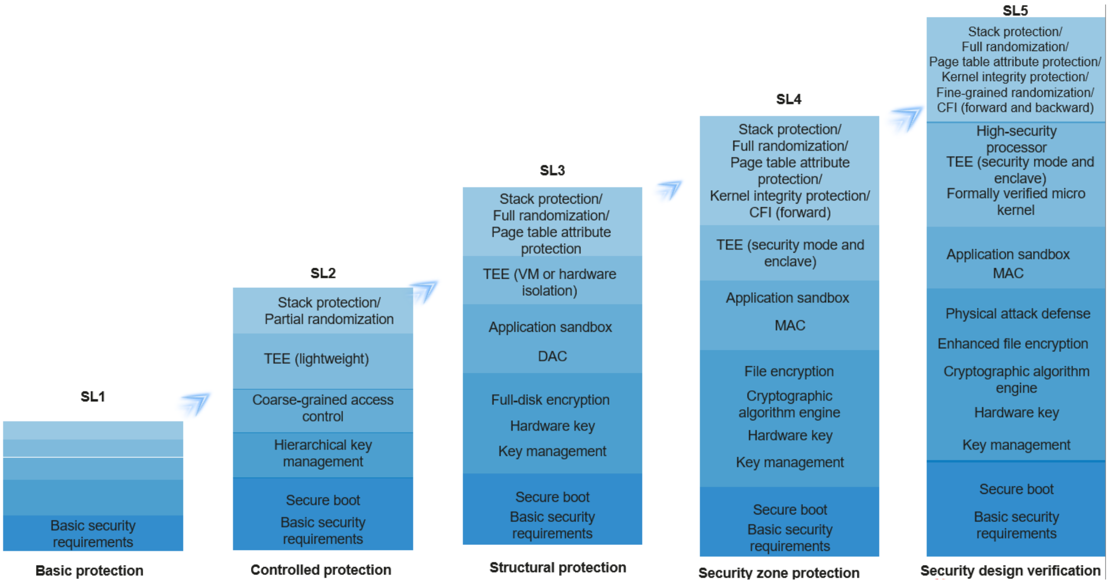
Figure 5-1 System security architecture


Basic security requirements
Secure coding
Physical interface security
Security management and maintenance
Secure O&M
Cryptographic algorithm security
Password security
Integrity protection
Secure boot of peripheral subsystems
One device, one authorization
Control over privileged software versions
Version rollback prevention
Enhanced security
Secure boot chain
Boot RoT
Encryption and data protection
HUK
Key management
File system encryption
Encryption and decryption algorithms
Anti-exploit
Page table attribute protection
Kernel integrity protection
ASLR
Memory access control
CFI
Permission and access control
Application permission
DAC
Capability
MAC
Application sandbox
TEE
Physical security environment
Formally verified microkernel
TEE-REE secure communication
TEE security service
TA management
Chip & OS
Boot RoT
Storage RoT
Computing RoT

The implementation of the security architecture varies depending on the threat analysis results (risk levels) and the resources (both software and hardware) of different devices running HarmonyOS. The security capabilities of such devices are classified into five security levels (SL1 to SL5) based on the Orange Book, real scenarios, and device types. Each level builds upon its preceding level (for example, SL3 includes all capabilities defined in SL2).

Figure 5-2 shows the device security levels.

Figure 5-2 HarmonyOS device security levels


SL1
Basic security requirements
Basic protection
SL2
Stack protection/ Partial randomization
TEE (lightweight)
Coarse-grained access control
Hierarchical key management
Secure boot
Basic security requirements
Controlled protection
SL3
Stack protection/ Full randomization/ Page table attribute protection
TEE (VM or hardware isolation)
Application sandbox
DAC
Full-disk encryption
Hardware key
Key management
Secure boot
Basic security requirements
Structural protection
SL4
Stack protection/ Full randomization/ Page table attribute protection/ Kernel integrity protection/ CFI (forward)
TEE (security mode and enclave)
Application sandbox
MAC
File encryption
Cryptographic algorithm engine
Hardware key
Key management
Secure boot
Basic security requirements
Security zone protection
SL5
Stack protection/ Full randomization/ Page table attribute protection/ Kernel integrity protection/ Fine-grained randomization/ CFI (forward and backward)
High-security processor
TEE (security mode and enclave)
Formally verified micro kernel
Application sandbox
MAC
Physical attack defense
Enhanced file encryption
Cryptographic algorithm engine
Hardware key
Key management
Secure boot
Basic security requirements
Security design verification

SL1 is the lowest security level. HarmonyOS devices at this level typically run a lightweight operating system and low-end microprocessors, provide simple services, and do not process sensitive data. Such devices are free from common errors and support software integrity protection. Devices that do not meet the requirements of SL1 can only function as accessories under the control of HarmonyOS and cannot control HarmonyOS devices or perform complex service collaboration.

HarmonyOS devices at **SL2** allow you to perform data annotation and define access control rules to implement discretionary access control (DAC). Such devices provide basic attack defense capabilities and can support a lightweight, secure isolation environment in which some required security services can be deployed.

HarmonyOS devices at **SL3** provide comprehensive security protection capabilities. They run an operating system that integrates comprehensive security semantics and mandatory access control (MAC). The operating system can be structured into critical and non-critical protection elements, with a clearly defined security policy model used to protect the critical elements. Devices at this level defend against common vulnerability exploits.

HarmonyOS devices at **SL4** provide the trusted computing base (TCB), which can prevent tampering and perform adequate authentication and arbitration to control access to critical protected objects. Such devices can defend against most software attacks.

SL5 is the highest security level. HarmonyOS devices at this level perform formal verification for core software modules of the system. Key hardware modules, such as the RoTs and cryptographic computing engine, are capable of defending against physical lab attacks. Such devices use tamper-resistant hardware security modules (a dedicated security chip, for example) to enhance the system's RoTs.

The following sections describe the typical security technologies used at each SL.

5.2 SL1

Secure Boot

The digital signatures of boot objects are verified at each step of the HarmonyOS device boot process, ensuring that only successfully verified image files can be loaded and executed. These files include the bootloader, kernel, baseband image, and firmware of other modules and co-processors. If a file fails the signature verification, the boot process is terminated.

When a device is started, the first program to be executed is the ROM SoC Bootloader, which is written into the chip's ROM during the manufacturing process and cannot be modified. The ROM SoC Bootloader functions as the root of trust for device boot, loading the Flash Device Bootloader from the flash storage chip after performing basic system initialization. It uses the public key hash in the eFuse space (using the fuse technique and cannot be changed once the fuse blows) of the main chip to verify the public key, and then uses the public key to verify the digital signature of the Flash Device Bootloader image. The Flash Device Bootloader, executed only if verification is successful, then loads, verifies, and executes the next image file. A similar process is repeated until the entire system is booted, eventually establishing a chain of trust that prevents unauthorized programs from being loaded during boot.

For greater security, some boot processes use encrypted images.

Enhanced Security

HarmonyOS devices ensure the integrity and authenticity of both system software during boot and platform software during OTA upgrade. The devices verify the signature of an upgrade package during system software updates, allowing only legitimate upgrade packages to be installed.

Additionally, HarmonyOS provides software update control. Before commencing an OTA upgrade using a downloaded software package, HarmonyOS applies for upgrade authorization by sending the digest information of the device identifier, the version number and hash value of the upgrade package, and the device upgrade token to the OTA server. The OTA server signs the digest and returns it to the device, authorizing the upgrade to proceed, only if the digest is successfully verified. If the verification fails, the device displays a message indicating that the upgrade has failed. This process prevents unauthorized software updates, especially updates using vulnerable software, from being installed on HarmonyOS devices.

5.3 SL2

Hardware Unique Key (HUK)

A HUK is a unique identifier immutably written in hardware during the manufacturing process. In HarmonyOS, it is typically used as the root key for key derivation.

HarmonyOS devices at SL2 and higher strictly control access to the HUK. For example, the HUK is accessible only through the hardware cryptographic engine (and not from software).

Key Management

HarmonyOS provides applications with the HarmonyOS Universal Keystore Service (HUKS), including key and certificate lifecycle management as well as encryption/decryption. The HUKS enables developers of HarmonyOS applications to manage the lifecycle of keys and certificates and call encryption/decryption algorithms.

HarmonyOS strictly controls access to keys, allowing only the applications that generate the keys to access them. When an application generates a key, the HUKS records its information (such as the UID, signature, and package name of the application). This information is then used for identity authentication when the application accesses the key. HarmonyOS applications can use the identity authentication function (including biometric authentication and PIN codes) to strengthen access control to the keys. The HUKS allows key access and relevant operations only after confirming the identity authentication result.

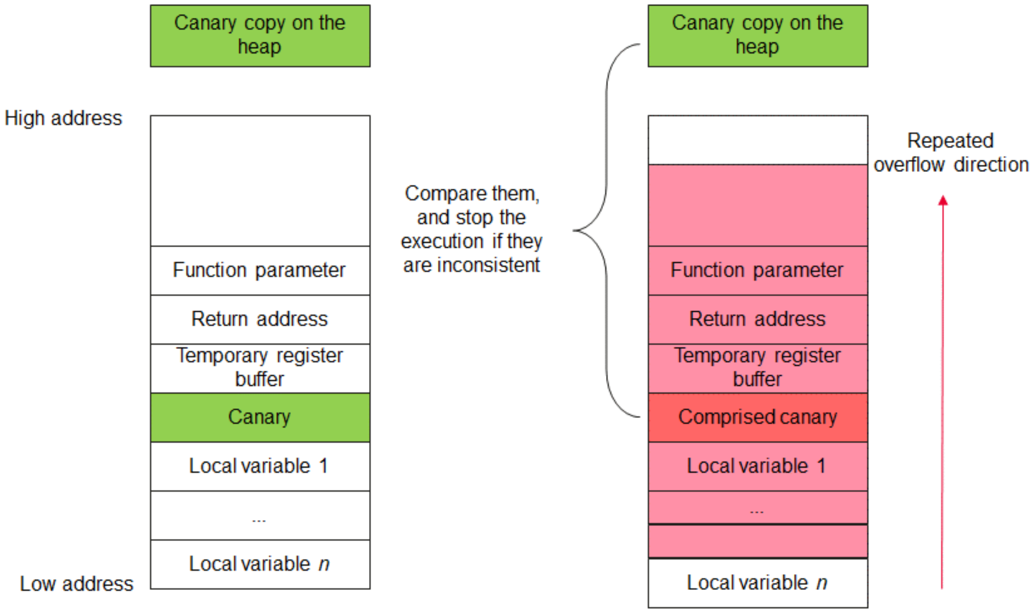
In addition, HarmonyOS provides the key attestation function to authenticate keys based on the device certificate, which is injected into and unique to each device. HarmonyOS also provides the ID attestation function, which offers trusted device identifier authentication capabilities for the cloud, covering device identifiers such as the SN and IMEI.

HarmonyOS devices at SL2 and higher implement the software architecture of the key management service in different ways. For example, they implement the architecture based on TEE or hardware security modules.

Stack Protection

Stack protection is the most cost-effective solution against stack overflow vulnerabilities. Most stack overflow attacks typically leverage continuous overwriting to destroy other data on the stack before the overwriting function returns addresses. During compilation, a canary variable is inserted between the local variable and the function return address. This allows the system to determine whether the return address is compromised by comparing the canary on the stack with the canary copy on the heap before the function returns. Stack protection ensures security while causing little impact on performance. HarmonyOS devices at SL2 and higher support stack protection.

Figure 5-3 Stack protection principle



DAC

In 1985, TCSEC proposed two well-known access control models: DAC and MAC.

DAC is widely used in the design of file system permissions. It includes classic UNIX style permission checks and ACL, which are implemented based on users, groups, and permissions. The owner of a file can grant other users access to the file, and the controls are discretionary. HarmonyOS devices at SL2 and higher support the DAC model.

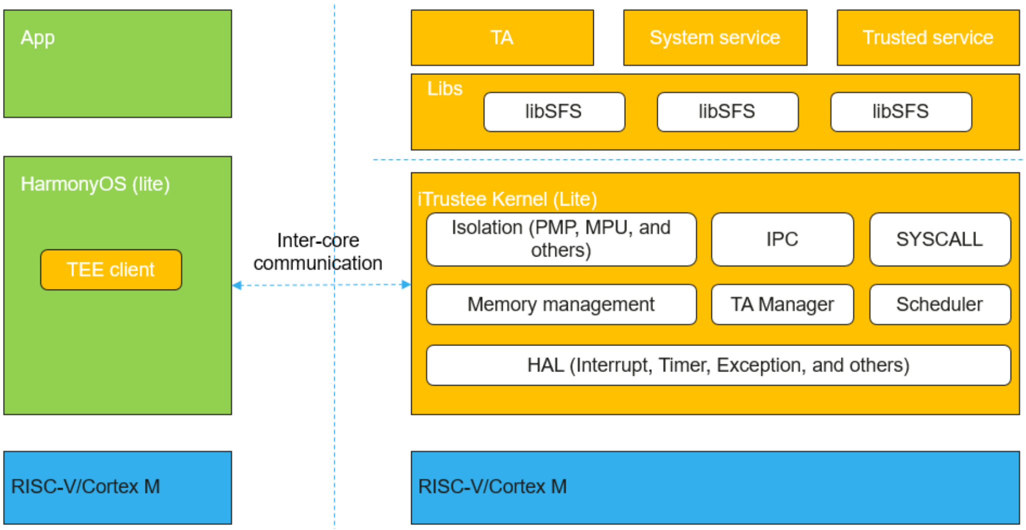
DAC gives subjects full control over objects. For example, subjects can grant resource access permissions to other subjects and withdraw the permissions. DAC is widely used in the design of file system permissions but lacks centralized management. DAC alone is not secure enough for HarmonyOS devices at SL4 and SL5.

Lightweight TEE

Lightweight devices may have certain security service and data protection requirements. Due to hardware resource limitations, however, the conventional TEE may not be a feasible option on these devices.

To address the resource limitations, lightweight HarmonyOS devices can deploy a small number of security services (such as key management and secure connection) by using the lightweight TEE, which involves fewer resource overheads. Figure 5-4 shows a typical lightweight TEE of HarmonyOS, which is implemented using low-end processors (such as RISC-V/Cortex-M) and inter-core isolation. This lightweight TEE is deployed on a specified physical core of a device.

Figure 5-4 Lightweight TEE



After Arm TrustZone for Cortex-A is successfully implemented, the TrustZone-M technology is delivered on the Arm Cortex M33 (low-end processor) for IoT devices. The TEE built on TrustZone-M also applies to the architecture shown in Figure 5-4 and is considered a lightweight TEE.

5.4 SL3

Address Space Layout Randomization (ASLR)

In the early stack overflow vulnerability, an attacker can change the return address to point to the stack itself, giving rise to shellcode execution. One method to mitigate this issue is to change the start location of the stack. This makes it difficult to predict the address space layout, making the attack more difficult and thereby improving security. Shellcode-like attacks can also be implemented using the Return Oriented Programming (ROP) technology by combining existing code snippets in the system. Therefore, full-space ASLR should be supported in addition to stack ASLR. HarmonyOS devices at SL3 therefore support both full-space ASLR and stack ASLR. They also support randomization of core protected objects, including stacks, shared libraries, mmap, and VDSO.

HarmonyOS devices at SL4 and SL5 support ASLR for code segments and heaps.

Note that the random entropy of 32-bit devices is too low for address space randomization to produce a sufficiently high value — only 64-bit devices provide adequate random entropy.

Data Execute Never (XN)

One way to prevent buffer overflow exploits is to block the execution of injected code. This can be achieved by preventing the CPU from executing injected shellcode in the data region as code.

HarmonyOS devices use data XN and ASLR together to achieve desired security protection.

Privileged Access Never (PAN) and Privileged eXecute Never (PXN)

HarmonyOS uses PAN and PXN to prevent the kernel from accessing user-space malicious data and executing user-space malicious code, thereby protecting the kernel against certain attacks.

Using some kernel attack methods, an attacker tampers with the data pointer in the data structure used by some kernels so that it points to the data structure that the attacker prepared in user mode, which launches an attack by affecting kernel behavior. PAN prevents the kernel from accessing user-mode data, thereby preventing such attacks.

Using some kernel attack methods, an attacker tampers with the code pointer in the data structure used by some kernels so that it points to the attack program in user mode, and then triggers execution of the attack program code by using a system call. PXN prevents the kernel from directly executing user-mode code, thereby preventing such attacks.

Hardware Encryption/Decryption Engine

Some cryptographic instructions, such as hash and symmetric algorithm instructions, are integrated in modern processors to deliver high computing performance. The dedicated cryptographic hardware engine refers to the encryption/decryption unit in modern SoCs. Compared with the cryptographic algorithms that are implemented based on processor-specific instructions, this engine has notable advantages in both energy efficiency and security, and supports more algorithm types.

HarmonyOS devices at SL3 provide a hardware engine for data encryption/decryption and key derivation. The hardware encryption/decryption engine supports the following algorithms:

- Symmetric algorithms: AES-128, AES-256, SM4, etc.
- Hash algorithms: SHA256, HMAC-SHA256, SM3, etc.
- Public key algorithms: RSA2048, ECDSA-P256, ECDH-P256, SM2, etc.

True Random Number Generator (TRNG)

Random number generation is required to generate keys, IVs, and salt values, and cryptographically secure random numbers are required to ensure unpredictability.

HarmonyOS devices provide the CTR_DRBG RNG compliant with NIST SP800-90A and the TRNG whose hardware entropy source is compliant with NIST SP800-90B.

5.5 SL4

Control over Privileged Software Versions

Using privileged software versions is a common method that attackers and black/gray industries adopt to obtain more permissions and compromise device security. HarmonyOS can control privileged software versions by separating the signatures of commercial and R&D software versions and using eFuse to ensure that commercial signatures are used after factory delivery. In this way, only commercial software versions can be run on commercial HarmonyOS devices, and no privileged software versions can be started. In scenarios where privileged software versions need to be run (for example, during maintenance), the authorization certificate issued by the original equipment manufacturer (OEM) must be obtained. The privileged software versions can be run only after passing verification.

In scenarios where information (related to countries/regions, carriers, commercial/demo devices, or temporary/permanent unlocking) needs to be changed, the online dongle mechanism must be used to perform authentication before permissions are granted.

TEE

HarmonyOS devices at SL4 support TEE. Huawei develops iTrustee based on TrustZone, which provides hardware-level security while striking a balance between performance, security, and cost. This technology allows CPUs to operate in a TEE or REE and switch between the two using special instructions in order to provide hardware isolation. A TEE protects and isolates hardware resources, such as memory and peripherals. End-to-end security is achieved by protecting the execution process, key confidentiality, data integrity, and access permissions, thereby preventing malware attacks from an REE.

HarmonyOS TEE provides multi-core and multi-thread capabilities to create multiple security tasks, which can be run on multiple CPUs to significantly improve the computing power of the TEE. In addition, HarmonyOS TEE supports the basic function library, mathematical library (C library and POSIX API), and dynamic library, facilitating TA development and deployment.

HarmonyOS TEE supports the following capabilities:

Basic security hardening

- TEE ensures authenticity and integrity throughout the entire lifecycle, including boot and upgrade.
- Reverse analysis of image files offers attackers valuable insight into a potential target. HarmonyOS TEE supports anti-reverse protection of image files, implementing it using mainly image encryption and symbol table obfuscation.
- HarmonyOS TEE provides attack defense capabilities, including secure compilation (PIC/PIE and REOLO), ASLR, stack protection, data XN, and read-only code segments and function pointers.

Security management

- Lifecycle management of TAs is supported, including certificate signature and revocation, integrity verification during installation, and session management.
- To ensure effective isolation between TAs and prevent attackers from exploiting TA vulnerabilities (which would allow them to continuously attack and compromise the TEE), HarmonyOS TEE supports fine-grained resource access and permission control.
- The TEE has multiple TAs to serve different tasks in an REE. HarmonyOS supports fine-grained access control over TAs to ensure that each one is dedicated to a specific service. Additionally, HarmonyOS uses trustlists to control which processes can access a specified TA, and supports authentication of process code segments to prevent spoofing.
- TEE is responsible for sensitive data processing and occupies certain system resources (such as memory). To improve the utilization of these resources, HarmonyOS TEE

supports dynamic resource management and reduces the static resource usage. For example, common memory can be dynamically converted into secure memory.

Security service

- The **trusted storage service** of HarmonyOS stores key information and ensures data confidentiality and integrity. Trusted storage supports device binding and isolation between different TAs, which can access only their own storage content and cannot open, delete, or tamper with that of other applications. Trusted storage is classified into two types: secure file system (SFS) and replay protected memory block (RPMB). An SFS stores ciphertext in a specific secure storage partition, whereas an RPMB stores ciphertext in a specific storage area of the embedded multimedia card (eMMC). The RPMB supports anti-deletion and anti-rollback protection.
- In terms of **encryption and decryption services**, HarmonyOS TEE supports multiple symmetric and asymmetric encryption/decryption algorithms and key derivation algorithms, along with derivation of the device group key, HUK, and standard encryption algorithms. This provides key storage and use services for third-party TAs. Furthermore, HarmonyOS TEE complies with the GlobalPlatform TEE standard, and uses independent hardware chips to perform key generation and calculation, thereby enhancing security.
- The **trusted time service** of HarmonyOS provides trusted reference time, which cannot be changed by malicious TAs or REE applications.
- HarmonyOS provides the **Trusted User Interface (TUI)**, which disables screenshots to protect the content displayed by TAs and prohibits access from the REE side. In this way, the TUI prevents the displayed data and input from being hijacking or tampered with by malicious applications, ensuring that such applications cannot view information on the screen or access the touchscreen. The TUI supports basic controls such as controls over PNG images, texts, buttons, and text entry boxes; display of Chinese characters, English letters, symbols, and digits using a consistent font size; custom UI; randomized keypad keys; and various controls and window management. Furthermore, the style of the TUI matches that used throughout the device UI.

HarmonyOS provides developers with the TEE, rich APIs, comprehensive SDKs, related reference manuals, and reference designs. It also provides security certificate management, application signature, TA lifecycle management, and application release services. In addition, Huawei DevEco Studio provides a unified development UI. These HarmonyOS capabilities enable third-party applications to be developed and commissioned.

Hierarchical Encryption of the File System

HarmonyOS devices at SL4 support file-level encryption, which uses the kernel encryption file system module and hardware encryption/decryption engine to implement the XTS-AES-256 algorithm.

To ensure the security of user data and application experience, HarmonyOS provides two encryption solutions: (1) Credential Encryption (CE), Sub Enhanced Credential Encryption (SECE), and Enhanced Credential Encryption (ECE) that work with the device lock screen password; and (2) Device Encryption (DE) that is irrelevant to the lock screen password. By default, the first solution is used. In this solution, class keys are relevant to the lock screen password and are protected by using both the lock screen password and HUK.

Control Flow Integrity (CFI)

ROP and jump-oriented programming (JOP) are attack methods that attempt to exploit program vulnerabilities by redirecting program control flows to the code snippets of existing programs. Attackers combine these code snippets to implement a complete attack.

A common way to implement ROP/JOP attacks is to overwrite a function pointer stored in memory, allowing a targeted check to be performed. In order to mitigate such attacks, CFI adds additional checks that confirm whether control flows stay within the preset scope. If undefined behavior is detected in a program, an exception handler will be invoked to process the behavior. Although CFI cannot prevent attackers from exploiting known vulnerabilities or even rewriting function pointers, it can strictly limit the scope of targets that can be called, making it more difficult for attackers to exploit vulnerabilities.

HarmonyOS uses Clang CFI and stack protection to mitigate ROP/JOP threats to the kernel.

CFI adds a check before each indirect branch to verify the validity of the target address and prevent such branches from jumping to an arbitrary code location.

Mandatory Access Control (MAC)

HarmonyOS supports MAC, whereby MAC policies are loaded to the kernel and cannot be dynamically changed upon device startup. MAC applies to all processes that attempt to access resources such as directories, files, and device nodes, and applies root-capability-based MAC to local processes with the root permission. This prevents malicious processes from reading and writing protected data or attacking other processes and limits the system impact of processes that are maliciously tampered with to a local scale, thereby enabling upper-layer applications to implement security protection.

HarmonyOS also provides the seccomp feature to restrict the system calls that can be invoked by processes, preventing malicious applications from using sensitive system calls to compromise the system.

Kernel Integrity Protection*

Although secure boot and verified boot ensure the authenticity and integrity of software during the boot, attackers may still be able to exploit vulnerabilities in authentic code.

The kernel integrity protection technology of HarmonyOS uses the virtualization extension model provided by ARMv8 processors to prevent vital registers, page tables, and code from being tampered with, thereby protecting the kernel. In this way, integrity protection and privilege escalation prevention are implemented during system uptime.

In addition to protecting static data (such as code and read-only data segments), this technology also protects some dynamic data by using the Write-Rare protection mechanism, which safeguards data that is mostly read but seldom modified in the kernel. Attackers cannot modify this data even if they obtain the kernel-level memory write permission by exploiting vulnerabilities.

The kernel integrity protection technology supports the following security protection mechanisms:

- Code snippets of the kernel and driver module cannot be tampered with.
- Read-only data of the kernel and driver module cannot be tampered with.
- Non-code snippets of the kernel cannot be executed.
- Critical dynamic kernel data cannot be tampered with.
- Key system register settings cannot be tampered with.

*Note: Only some HiSilicon chip models in China support this function.

5.6 SL5

Secure Element (SE)*

An SE is a subsystem that provides a highly secure execution and storage environment. HarmonyOS supports the deployment of an SE, which is used to ensure the security of core services and data, such as mobile payment and identity ID. Unlike TEE, the SE solution provides both software and hardware protection through high security design and software algorithms. This solution not only delivers software security protection capabilities, but also defends against physical attacks. It provides improved protection and ensures the security of core security services provided by HarmonyOS devices.

*Note: The SE used by equipment vendors must be certified by related industries and organizations to support mobile payment and financial services.

Independent Security Chip

The SE is mainly used to deploy a specific security service, whereas the independent security chip can enhance the system security capability of HarmonyOS devices.

HarmonyOS uses the highly secure environment of the independent security chip (physical security) to provide security services, such as lock screen password protection, file encryption, biometric protection and identification, key management, RoTs, and anti-rollback. In this way, the basic security capabilities of HarmonyOS devices are ensured at the hardware level. These security services rely on the specific security chips deployed on devices.

Formal Verification + Microkernel

HarmonyOS TEE uses the microkernel architecture, which simplifies kernel functions and adopts a modular design to implement more system services outside the kernel and reduce the attack surface. Thanks to this architecture, HarmonyOS TEE delivers good scalability and high security. By providing stable underlying library interfaces, HarmonyOS TEE simplifies application development and portability, and supports the development of the security service ecosystem.

In addition, HarmonyOS TEE is trialing the use of formal verification, which significantly improves the security levels of the TEE kernel and key modules to build trustworthiness and security. Formal verification effectively verifies system correctness (without vulnerabilities) from the source by using mathematical theorems. Traditional verification methods, such as function verification and attack simulation, can only be used in limited scenarios, whereas formal verification can be used to verify all software running paths through data models. The correctness of core modules, core APIs, and high-level mechanisms (such as process isolation and permission management) is verified to prevent data race and memory access errors.

With this method, HarmonyOS microkernel for TEE has passed CC EAL 5+ and will achieve a higher security level in the future.

Physical Attack Defense

The core security modules of HarmonyOS devices at SL5 are capable of defending against physical attacks, including side-channel attacks and fault injection. The core security modules include independent security chips (including SEs), hardware encryption/decryption engines, and fuse storage modules.

The independent security chip uses digital sensors and the active shield layer technology to prevent fault injection and physical attacks. When a digital sensor circuit detects fault injection, it generates an alarm that triggers the system to take appropriate protective measures. The active shield layer covers the key circuits of an independent security chip. If a physical attack is launched against the chip, an alarm on the active shield layer is generated, making it difficult for attackers to steal sensitive information from the key circuits by bypassing the active shield layer.

The public key cryptographic engine provides protection against advanced side-channel attacks and fault injection. Security protection measures are mainly reflected in algorithm scheduling design. Point multiplication for elliptic curve cryptography (ECC) aims to reduce physical attack risks and prevent common attacks such as SPA, DPA, and Zero Point by introducing random number key masking, coordinate randomization, random masking, intermediate value masking during calculation, validation of core parameters, online elliptic curve verification, and other security designs. RSA modular exponentiation uses security designs such as multiple modular multiplications of per-bit keys, intermediate value masking during calculation, random insertion of pseudo keys, and CRC to defend against physical attacks.

The symmetric algorithm engine also defends against advanced side-channel attacks and fault injection. To achieve this, the engine uses full-path balanced mask protection algorithm, high-security Sbox mask protection design, CRC on vital intermediate values for integrity protection, and time-redundancy-based fault injection prevention.

eFuse is used to protect the confidentiality and integrity of sensitive information, such as root keys. Sensitive data comes with CRC values to prevent fault injection. Digital sensors can be deployed in strategic areas of the eFuse to bolster defense against fault injections, and redundancy design can be implemented for some key control registers to enhance the robustness of the eFuse.

5.7 Trusted Interconnection of Distributed Devices

To ensure the security of user data flows between devices in a distributed scenario, the devices must trust each other. Specifically, they must establish a trust relationship, which they need to verify before establishing a secure channel. The trust relationship can be established between devices with the same HUAWEI ID or between devices in point-to-point (P2P) trust relationships.

Interconnection Security for Devices with the Same HUAWEI ID

HarmonyOS provides authentication services for devices that are signed in using the same HUAWEI ID. Each HarmonyOS device that is signed in using a HUAWEI ID generates a public/private key pair using elliptic curve cryptography as the authentication credential, and applies for public key attestation from the Huawei Cloud server. The private key credential is stored only on the device — it is never sent to the server.

When devices with the same HUAWEI ID are discovered and connected by DSoftBus in the near field, device authentication and session key exchange will be performed based on the public and private key pairs of the two devices. After the authentication is successful, the security channel of DSoftBus uses the session key provided by the device authentication service to encrypt the transmitted data using AES-GCM. In this way, even if vulnerabilities exist in Bluetooth or Wi-Fi connection, the data transmitted over the channel is encrypted in an end-to-end manner, and only the devices with the same HUAWEI ID can decrypt the data. The session key is valid only for the current session.

Device Connection Security Based on P2P Trust Relationship

To initiate distributed services between two devices that use different HUAWEI IDs, it is necessary to establish a P2P trust relationship between the two devices. This ensures that the connected device is not under the control of an attacker. HarmonyOS provides the device authentication service based on P2P trust relationship.

To ensure that the P2P trust relationship is authentic and trusted, the user needs to manually create shared secret information between the two devices. This can be achieved by, for example, scanning the other device's QR code or entering the random PIN code displayed on the other device.

The device authentication service of HarmonyOS executes the password authenticated key exchange (PAKE) security protocol based on the shared secret information created by the user. A secure communication channel is established only after the protocol authentication is complete. In addition, the devices generate their own public/private key pairs using elliptic curve cryptography as authentication credentials, exchange the public key credential with the peer device over the secure communication channel, and store the credentials. Because the secure communication channel is protected by the shared secret information created by the user, the exchanged public key credentials cannot be hijacked or replaced, even if vulnerabilities exist in the Bluetooth or Wi-Fi connection, thereby preventing implantation of forged identities.

When a device for which the P2P trust relationship is established is discovered and connected by DSoftBus in the near field, device authentication and session key exchange will be performed based on the peer-end public key credentials that are exchanged and stored when the P2P trust relationship is established. After the authentication is successful, the security channel of DSoftBus performs E2E encryption of the data transmitted over the channel (similar to how it performs encryption on other devices using the same HUAWEI ID) so that only the device in P2P trust relationships can decrypt the data.

6 Hierarchical Access Control Architecture for "Right Access" in HarmonyOS

HarmonyOS protects the security of consumer and developer data throughout the data lifecycle using protection measures aligned with the sensitivity of personal data, the importance of system data, and the value of application data assets.

It adopts a hierarchical data access control architecture which draws core ideas from the confidentiality protection and integrity protection policies of the BLP and Biba models, respectively. In short, data is assigned labels indicating risk levels during creation and associated with full-lifecycle access control permissions and policies accordingly. During data storage, different encryption measures are required based on risk levels. During data transmission, highly-sensitive data must not be transferred to devices with low security capabilities, and highly-sensitive resources and peripherals must not send control instructions.

Right access to data will be implemented throughout the data lifecycle based on the BLP and Biba models.

6.1 Data Classification Specifications

Data is classified based on the potential impact of data breaches or damages on individuals, organizations, or the public. Consequently, protection requirements are proposed based on data risk levels.

FIPS 199 provides a mechanism to determine data risk levels through risk assessment based on three security objectives — data confidentiality, integrity, and availability — which primarily consider the impact on individuals, organizations, or the public. The higher the impact, the higher the risk level, as shown in the following table.

Risk assessment formula: Risk level = F{Confidentiality, Integrity, Availability}

Security Objective and Potential Impact	**Low**	**Medium**	**High**
Confidentiality Protecting information (including personal privacy and proprietary information) during access and disclosure by means of encryption, access control, and other measures	The unauthorized disclosure of information is expected to have a limited adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized disclosure of information is expected to have a serious adverse effect on organizational operations, organizational assets, or individuals. For example, fines and damage to reputation.	The unauthorized disclosure of information is expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals. For example, significant business losses, reputational losses, and exit from specific industries.
Integrity Guarding against improper modification or destruction of information, and ensuring non-repudiation and authenticity of information	The unauthorized modification or destruction of information is expected to have a limited adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized modification or destruction of information is expected to have a serious adverse effect on organizational operations, organizational assets, or individuals.	The unauthorized modification or destruction of information is expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals.
Availability Ensuring timely and reliable access to and use of information	The disruption of access to or use of information, or an information system, is expected to have a limited adverse effect on organizational operations, organizational assets, or individuals.	The disruption of access to or use of information, or an information system, is expected to have a serious adverse effect on organizational operations, organizational assets, or individuals.	The disruption of access to or use of information, or an information system, is expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals.

HarmonyOS classifies data into high, medium, and low risk levels based on the potential impact of data breaches with reference to good industry practices. (References: ISO/IEC 27005, FIPS 199, and NIST SP 800-122). For non-personal data, the "public" risk level is added; for sensitive personal data (for example, special categories of personal data defined in

EU GDPR and sensitive personal data defined in GB/T 35273-2020 on Information Security Technology - Personal Information Security Specification), the "critical" risk level is added. A label is assigned to each risk level, as shown in the following table.

Data Privacy Category	**Data Type**	**Risk Level**	**Example**
Sensitive personal data	Identity authentication credential	Critical (S4)	Tokens and passwords used for identity authentication
	Personal racial and ethnic information		Racial or ethnic origin
	Negative reputation data		Negative records such as criminal records and disciplinary actions
	Health information		Data relating to body fat, blood pressure, blood glucose level, heart rate, blood oxygen level, ECG, medical records, sex activities, and sleep patterns
	Biometric features		DNA, fingerprint, facial features, iris, voiceprint, palm print, ear pinna, and behavioral features
General personal data	Workout data	High (S3)	Step count, workout distance, workout duration, calories consumed, climbing height, oxygen intake, running posture, and heart rate
	Personal multi-media data		Images, texts, audio, and video on user devices
	Age and birth date	Medium (S2)	Age and birth date
	User social identifiers		Social identifiers that can be discarded, replaced, and re-registered by users, such as HUAWEI IDs and social accounts
	Name and nickname		Name and nickname
	Address information		Postal code, work address, and home address
	Basic personal information	Low (S1)	Gender, nationality, place of birth, education background, and professional background
	Positive reputation data		Professional achievements

Data Privacy Category	Data Type	Risk Level	Example
Non-personal data	System key	High (S3)	System root key, working keys derived from the root key for encrypting system services and applications, and working keys generated by applications for encrypting system services and applications
	Miscellaneous non-personal data	Low/Public (S0)	Publicly released data attached to system and device information, such as software version number, engine version number, client version number, driver version number, SDK version number, and application category

6.2 Overview of Data Security and Privacy Protection Management Throughout the Data Lifecycle

HarmonyOS protects data according to its risk level throughout the data lifecycle. The lifecycle of data on a smart device is comprised of the following phases:

Creation: a process in which data is generated on the smart device or application software, or collected by, received by, or otherwise transferred to the smart device or application software from another device

Storage: a process in which data is stored on a smart device

Usage: a process in which data is accessed, processed, or otherwise used on a smart device

Transmission: a process in which data is transmitted from the source smart device to the destination device

Destruction: a process in which data on smart devices is destroyed to ensure that it cannot be retrieved or accessed

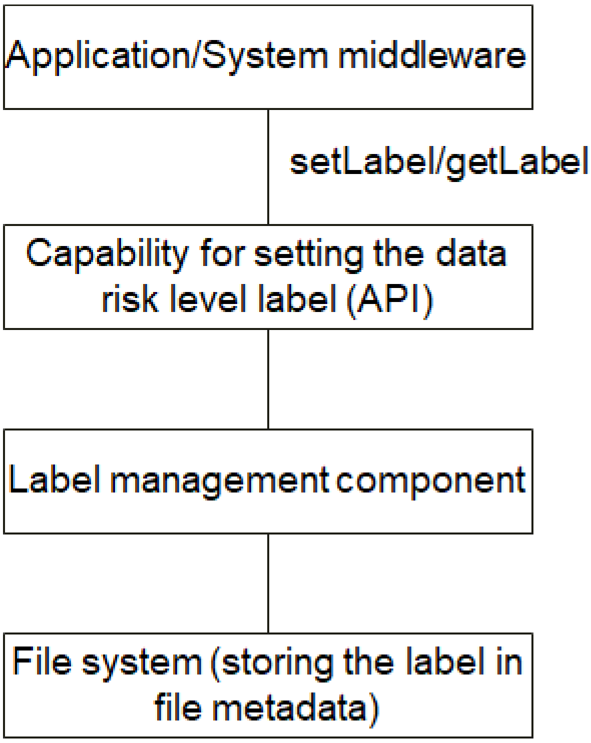
6.3 Security Mechanisms for Data Creation

HarmonyOS allows risk level labels to be set for data during file or data creation using the following APIs:

API	API Function
public int setLabel(Context context, String filePath, String labelName, String labelValue, int flag)	Sets a risk level and a protection policy for a file. **labelName** indicates the risk level label, and **labelValue** indicates the risk level, which ranges from S0 to S5.

public String getLabel(Context context, String filePath, String labelName)	Obtains the risk level of a file.
public int getFlag(Context context, String filePath, String labelName)	Obtains flag information relating to the file risk level.

As shown in the following figure, a service application can call the API to set risk level labels for stored application files, and this risk level information is ultimately stored in the metadata of the corresponding files.


Application/System middleware
setLabel/getLabel
Capability for setting the data risk level label (API)
Label management component
File system (storing the label in file metadata)

The service application needs to set risk levels for files or data based on the service risk level definitions provided by HarmonyOS. In addition, the application must assess the security level of the device and verify that it is proportionate to the risk level of the data being stored, thereby ensuring that data is properly protected throughout its lifecycle.

Device Security Level	**SL5**	**SL4**	**SL3**	**SL2**	**SL1**
Supported data risk levels	S0–S4	S0–S4	S0–S3	S0–S2	S0–S1

6.4 Security Mechanisms for Data Storage

HarmonyOS leverages the kernel encryption file system module and the hardware encryption/decryption engine to provide file-level encryption for data storage. In this regard, AES-XTS-256 encryption is used.

Based on data risk levels, HarmonyOS provides the following encryption schemes (using the mobile phone system as an example):

- Data encryption coupled with lock screen password (CE/SECE/ECE): The class key of the encrypted data is protected by both the lock screen password and HUK. Details are as follows:
  - CE: CE-protected data only becomes accessible after the lock screen password is entered following power-on, and remains accessible when the screen is locked. Such data includes the gallery, contacts, SMS, calendar, and call records.
  - SECE: This scheme enhances CE. When the device is locked, SECE-protected files cannot be opened, but file creation and write operations can be executed. For example, email attachments can be downloaded and written at the backend.
  - ECE: This scheme enhances SECE. When the device is locked, ECE-protected files cannot be opened and file creation is not allowed until the device is unlocked.
- DE: DE-protected data can be accessed upon power-on, regardless of whether the device is locked. Such data includes wallpapers, alarm clocks, and ringtones. The DE key is protected by the HUK and has no relation to the lock screen password.
- Non-encryption (NE): Data is not encrypted at all, which is rare. The OTA upgrade package is an example.

Figure 6-1 File encryption levels

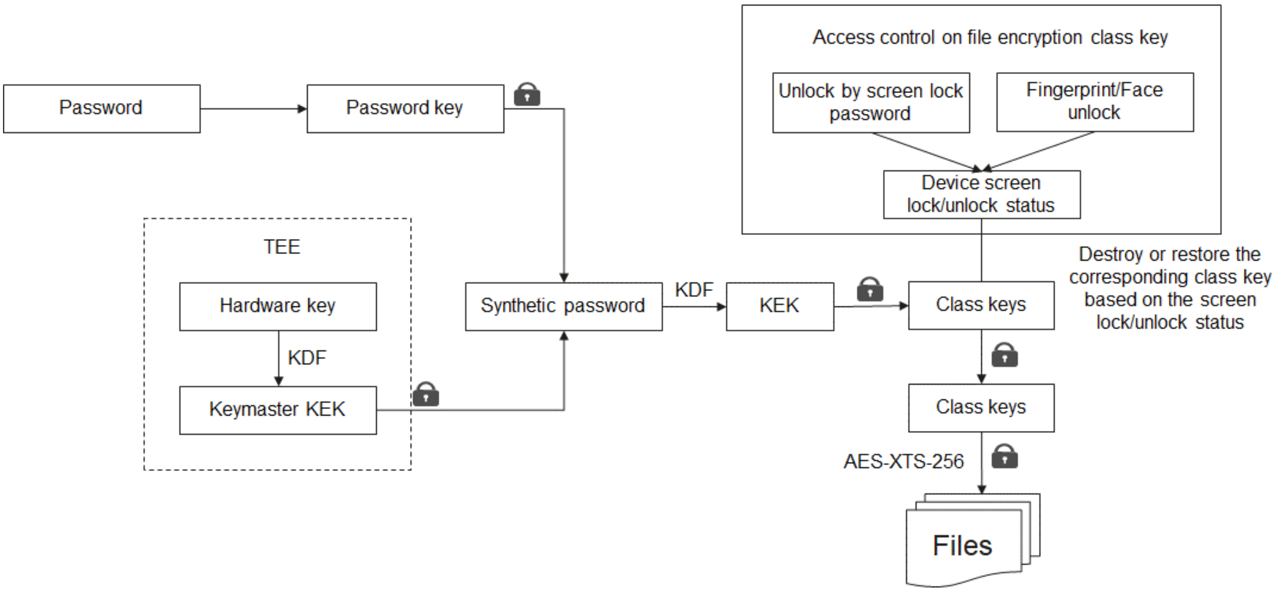

Access control on file encryption class key
Unlock by screen lock password
Fingerprint/Face unlock
Device screen lock/unlock status
Password
Password key
TEE
Hardware key
KDF
Keymaster KEK
Synthetic password
KDF
KEK
Class keys
Destroy or restore the corresponding class key based on the screen lock/unlock status
Class keys
AES-XTS-256
Files

Figure 6-1 shows the key hierarchy of file-level encryption for HarmonyOS mobile devices.

If the chip platform provides hardware-level encryption, the plaintext class keys and file keys related to file encryption are all created, stored, used, and destroyed inside the TEE, but not the REE, which enhances the security of file-level encryption.

HarmonyOS provides different levels of data protection based on data risk levels:

For critical (S4) data, HarmonyOS offers up to the ECE scheme.

For high-risk (S3) data, HarmonyOS offers up to the SECE scheme.

For medium-risk (S2) and low-risk (S1) data, HarmonyOS offers up to the CE/DE scheme.

6.5 Security Mechanisms for Data Use

In the data use phase, HarmonyOS provides DAC and MAC to ensure that only the right applications can access data.

The HarmonyOS distributed file system features a distributed sandbox capability to ensure that only the right applications can access data in cross-device data access scenarios.

HarmonyOS manages file encryption keys and provides ECE/SECE for enhanced access control over high-risk data, ensuring that only users who have access to the device (those who can unlock the device) can use the high-risk data.

HarmonyOS uses various protection methods for file encryption class keys, so that different encryption schemes (DE/CE/SECE/ECE) provide different file protection capabilities.

Data Encryption	Class Key Lifecycle	File Protection Mode
DE	The class key is available after device power-on.	The corresponding file can be used after device power-on.
CE	The class key is available after the device is powered on and unlocked using a correct lock screen password.	The corresponding file can be used after the device is powered on and the screen is unlocked using a correct lock screen password.
SECE	The class key is available after the device is powered on and the screen is unlocked using a correct lock screen password. When the screen is locked, the class key is temporarily cleared from the system. In scenarios where an application opens an existing file, the class key is unavailable; in scenarios where an application creates a file, the system temporarily restores the class key of the file. When the device is unlocked by the user again, the class key is restored.	When the device is locked, SECE-protected files cannot be opened, but file creation and write operations can be executed.
ECE	The device is in use, and the class key is available after the device is unlocked. When the device is locked by the user, the class key is temporarily cleared from the system. When the device is unlocked by the user again, the class key is restored.	When the device is locked, ECE-protected files cannot be opened and file creation is not allowed until the device is unlocked.

6.6 Security Mechanisms for Data Transmission

To protect user data and privacy, high-risk data must not be transferred to a device with a lower security level than that of the source device without a user's knowledge.

With this in mind, the HarmonyOS distributed system provides a cross-device access control mechanism based on data risk levels, ensuring that data is only transferred to devices that provide a security level proportionate to the risk level of the data.

Security Level of the Receiving Device	SL5	SL4	SL3	SL2	SL1
Supported data risk levels	S0–S4	S0–S4	S0–S3	S0–S2	S0–S1

Data can only be transferred to a device which fails to provide a security level proportionate to the data risk level following explicit authorization by the user on the source device.

The access control mechanism is implemented in the HarmonyOS distributed database and distributed file system, and a service may use the distributed capability to securely transmit data between devices that have established a trust relationship within the HarmonyOS distributed system.

6.7 Security Mechanisms for Data Destruction

Standard factory restoration operations usually do not erase all data stored on physical storage. While logical addresses are usually deleted for efficiency, this method does not clear the physical address space, and the data can often be restored.

Using the following mechanism, the factory restoration function of HarmonyOS can securely erase all stored data: An overwrite command is sent to the physical storage to erase the data, with all erased data represented by 0s or all 1s. This ensures that sensitive user data cannot be restored using software or hardware techniques, protecting data security in cases of abandoned or resold devices.

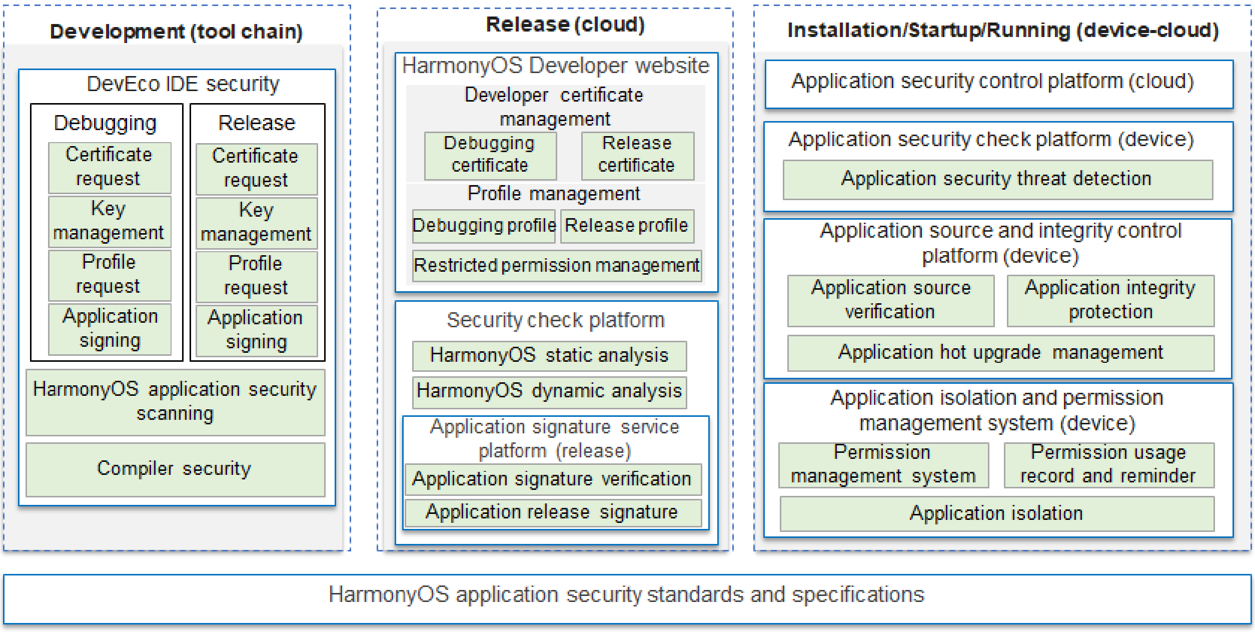
7 HarmonyOS Ecosystem Governance Architecture

HarmonyOS provides a pure application- and device-oriented ecosystem governance architecture, which ensures that applications and IoT devices running on the HarmonyOS super device comply with HarmonyOS security standards and specifications along with data security and privacy protection requirements, thereby protecting consumer rights and interests.

7.1 Overview of the HarmonyOS Application Lifecycle Governance Architecture

The HarmonyOS application lifecycle governance architecture manages applications throughout their lifecycle, encompassing development, release, installation, running, and uninstallation. It ensures that applications are developed in compliance with security and privacy specifications, originate from trustworthy sources, and are integrity-protected throughout their lifecycle. It also ensures that applications in running state are trustworthy, protect consumer privacy and data security, do not exhibit harassment or other malicious behavior, and are both traceable and controllable.

Figure 7-1 HarmonyOS application lifecycle governance architecture


Development (tool chain)
DevEco IDE security
Debugging
Certificate request
Key management
Profile request
Application signing
Release
Certificate request
Key management
Profile request
Application signing
HarmonyOS application security scanning
Compiler security
Release (cloud)
HarmonyOS Developer website
Developer certificate management
Debugging certificate
Release certificate
Profile management
Debugging profile
Release profile
Restricted permission management
Security check platform
HarmonyOS static analysis
HarmonyOS dynamic analysis
Application signature service platform (release)
Application signature verification
Application release signature
Installation/Startup/Running (device-cloud)
Application security control platform (cloud)
Application security check platform (device)
Application security threat detection
Application source and integrity control platform (device)
Application source verification
Application integrity protection
Application hot upgrade management
Application isolation and permission management system (device)
Permission management system
Permission usage record and reminder
Application isolation
HarmonyOS application security standards and specifications

7.2 Pure Application Development in HarmonyOS

HarmonyOS provides developer registration, account management, real-name authentication, developer certificate management, and other management capabilities related to application development and debugging.

Development tools offer security capabilities to help developers perform code-level and binary-related security and privacy checks, enabling them to quickly develop highly secure HarmonyOS applications.

In addition, the DevEco IDE provides application source control and integrity protection capabilities for developers, enabling the automatic generation and management of keys, signatures, debugging certificates, and debugging devices, and allowing any developed applications or services to be quickly released.

Real-name authentication requirement: According to the Provisions on the Administration of Mobile Internet Applications Information Services issued by Cyberspace Administration of China on June 28, 2016, a HarmonyOS developer needs to register an account and real-name authentication can be performed during account registration; together this promotes the healthy and orderly development of the ecosystem and protect legitimate rights and interests of developers and users. Real-name authentication includes that for individual developers and that for enterprise developers, and traceability is ensured so that application developers can be held accountable. Real-name authentication is required before applications are released. Therefore, developers are advised to pass real-name authentication during account registration.

7.3 Pure Application Release in HarmonyOS

When receiving an application release request from a developer, the release system checks whether the integrity of the application is compromised during upload, and examines and reviews it based on the HarmonyOS application check specifications. Applications that pass these checks and meet release criteria are re-signed by the system. This process ensures that the right developers release the right applications.

7.4 Pure Application Running in HarmonyOS

During application installation, HarmonyOS uses the Public Key Infrastructure (PKI) to verify the validity and integrity of applications.

HarmonyOS designs a new security and privacy protection mechanism for applications.

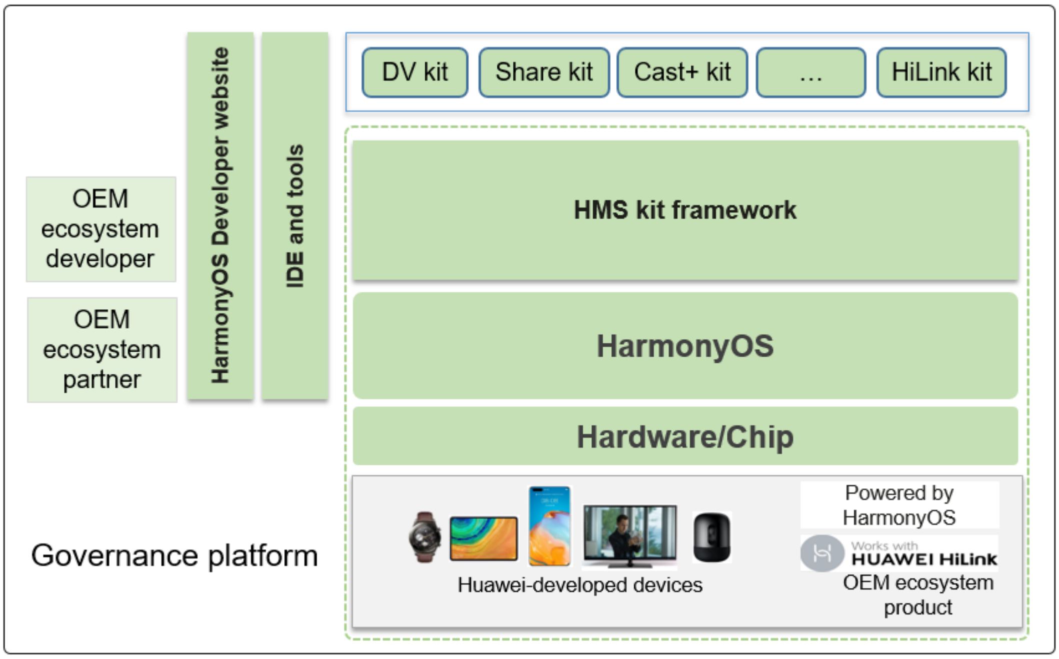
- Pure source: Applications are installed in pure mode. For example, control mechanisms are implemented for Android applications not sourced from application markets, and hot updates are not allowed.
- Pure permission: Permissions on SMS, phone calls, and call records which may include personal data are not open to ecosystem applications, and permissions on contacts and more are strictly controlled using permission certificates. In addition, System Picker is forcibly applied to the gallery and contacts to prevent permission abuse.
- Storage sandbox: The storage access permission is not open to ecosystem applications, ensuring application data security.

7.5 Overview of HarmonyOS Device Ecosystem Governance Architecture

Huawei has built a basic device ecosystem framework based on HarmonyOS and HMS kit framework, and provides kits for device vendors — such as the DV kit for device virtualization and the Cast+ kit for device casting.

In addition, Huawei has established a partner management platform to ensure optimal experiences for OEM devices, allowing only qualified vendors and devices that have passed the corresponding security certifications and tests to access the HarmonyOS ecosystem. Huawei also provides a complete set of security specifications for device development, helping ecosystem partners develop devices that meet ecosystem experience and security requirements.

Figure 7-2 HarmonyOS device ecosystem governance platform



The HarmonyOS device ecosystem governance architecture provides the following capabilities:

- OEM ecosystem developer certification
- Device security testing and certification
- Authorization credential management for device security

7.6 HarmonyOS Device Ecosystem Partner Qualification

To ensure that devices entering the HarmonyOS device ecosystem meet the associated experience and security requirements, vendors must be traceable and undergo real-name authentication. As such, they must first register a HUAWEI ID on Huawei's official website and complete real-name authentication and enterprise qualification. Next, they must agree to and sign the HUAWEI Smart Device Partner Service Agreement on the Device Partner ecosystem management platform. Once these steps have been completed, the vendor can become a Huawei ecosystem partner.

An ecosystem partner can select a partner type based on the product certification type on Device Partner, create a product entry based on the cooperation plan provided by the management center of the platform, and register product information. The corresponding development guide and specifications will then become available on the partner management platform.

7.7 Security Certification for HarmonyOS Ecosystem Devices

After development of a device has been completed, the partner is required to perform security self-checks and rectify any security issues according to device security classification specifications. The device can then be sent to Huawei's ecosystem certification lab for security testing. If it meets the security level certification requirements and passes testing, the lab will award the corresponding certificate and logo files.

7.8 Hierarchical Control Mechanism for HarmonyOS Ecosystem Devices

To ensure that device data flows with a certain level of security, partners should specify the minimum security level required for the selected functions to be integrated and provide the technical requirements for the corresponding security level, which device vendors must abide by when developing devices. Once development is complete, the device's security level will be certified.

A device can only be assigned a security level after it has met the minimum security level test requirements. Huawei registers the device's security level information on Device Partner, and this information can be obtained as follows: The security level information is digitally signed with the corresponding credential issued on the cloud. The device initiates a request for the security level information, and a challenge message is sent to the cloud. Upon receiving the request, the cloud issues the credential signed using the challenge message and security level information. The device then imports the security level information to the trust zone where an anti-rollback mechanism is provided to prevent a rollback from occurring.

In cases where ecosystem devices interoperate with the service kit capabilities provided by Huawei, they must first evaluate each other's security levels and only transmit data to those that meet security level requirements. For example, as a service requires data to be transmitted only to devices ensuring security level SL2, it needs to confirm that the peer device meets the SL2 requirements before transmitting data to that device. Consequently, Huawei provides a hierarchical control mechanism for devices. For details, see chapter 6 "Hierarchical Access Control Architecture for "Right Access" in HarmonyOS."

8 HarmonyOS Compliance with Security Standards and Certifications

HarmonyOS is designed and implemented with reference to public standards on cyber security, system security, data security, and others, and in compliance with local privacy protection laws, regulations, and standards.

HarmonyOS 2 has obtained the following certifications:

Certification Name	**Certification Object**	**Issued By**	**Description**
CC EAL 5+	TEE secure operating system microkernel	Netherlands NSCIB	The CC certification is an authoritative security certification that comprehensively assesses the security functions and security assurance capabilities of IT products, covering product design and development, security functions, and delivery management, based on ISO/IEC 15408, the information technology security standard. CC lists seven levels, with EAL 1 being the most basic and EAL 7 being the most stringent. The TEE secure operating system microkernel is CC EAL 5+ certified.
CC EAL 2+	iTrustee 5.0 TEE	Netherlands NSCIB	Huawei's TEE secure operating system iTrustee 5.0 is CC EAL 2+ certified.
IT Product Information Security Certification EAL 4+	HarmonyOS 2	China Cybersecurity Review Technology and Certification Center	The certification evaluates devices according to the Security Technical Requirements for Operating System on Smart Mobile Terminal (EAL 4+), which is based on the Chinese counterpart of CC — GB/T 18336. HarmonyOS 2 has been comprehensively evaluated using

Certification Name	Certification Object	Issued By	Description
			the CC approach. The certificate demonstrates that HarmonyOS 2 has been verified to better protect the security of customer data.
ISO/IEC 27701	Device software of Huawei Device Co., Ltd.	British Standards Institution (BSI)	The ISO/IEC 27701 privacy information management system provides a complete set of personal data processing methods and privacy information management framework, covering organizational governance, legal compliance, process specifications, information technology, supervision, and audit. This certification indicates that the software of Huawei devices has a comprehensive personal information protection management system in its design, R&D, and O&M activities.

9 Typical Advanced Security Service Capabilities of HarmonyOS

This section describes the advanced security capabilities of HarmonyOS, including Huawei Pay, transport card, secure keys, electronic identity (eID), and car key. It provides examples and scenarios to systematically describe how applications and services use these security capabilities to maximize protection of consumers' privacy, property, and data.

9.1 Huawei Pay

Huawei Pay has enhanced security in both hardware and software design, allowing users to make payments on supported Huawei devices conveniently and securely.

Huawei Pay Components

SE: is an industry certified and recognized chip that complies with the requirements of digital payment in the finance industry.

Near Field Communication (NFC) controller: processes NFC protocols and supports communication between the application processor and SE, and between the SE and POS terminal.

Huawei Pay application: refers to "Wallet" on devices that support Huawei Pay. This application enables users to add, manage and make payments using credit and debit cards. Users can also query their payment cards and other information about the card issuers, and add a new payment card to Huawei Pay.

Huawei Pay server: manages the status of bank cards registered with Huawei Pay and the device card number stored in the SE. The server communicates with devices and payment network servers at the same time.

How Huawei Pay Uses the SE

Encrypted bank card data is sent from a payment network or card issuer to the SE. It is then stored in the SE and protected by the security functions provided by the SE. During a transaction, a device directly communicates with the SE using a dedicated hardware bus through the NFC controller.

How Huawei Pay Uses the NFC Controller

The NFC controller functions as the gateway to the SE and ensures that all contactless payments are conducted through POS terminals in close proximity to payment devices. The NFC controller only marks the payment requests from devices in the field as contactless transactions.

Once a cardholder authorizes payment through fingerprint or password authentication, the controller sends the contactless response of the SE to the NFC chip. This way, detailed information about payment authorization for contactless transactions is saved only in the local NFC chip and will not be disclosed to the application processor.

Bank Card Binding

When a user links a bank card to Huawei Pay, Huawei securely sends the bank card information and other information about the user account and device to the card issuer. The card issuer then determines whether to allow the user to add the card to Huawei Pay.

Huawei Pay uses commands invoked on the server to send and receive packets exchanged with the card issuer or network. The card issuer or network uses these commands to verify, approve, and connect more bank cards to Huawei Pay. The sessions between clients and servers are encrypted using TLS.

Adding Bank Cards to Huawei Pay

To manually add a bank card, users must enter their name, card number, card expiration date, and card verification value (CVV) code. Users can enter card information manually in the Wallet application or use the camera function to automatically recognize card details. If the camera is used to capture card details, the Wallet application will attempt to automatically fill in the card number. Once all information is input, it is verified (excluding the CVV code) with the card issuer for security purposes. Huawei will not retain or use information such as the CVV code.

If the card issuer returns any terms and conditions during the card verification process, Huawei downloads the terms and conditions and displays them on the user's device.

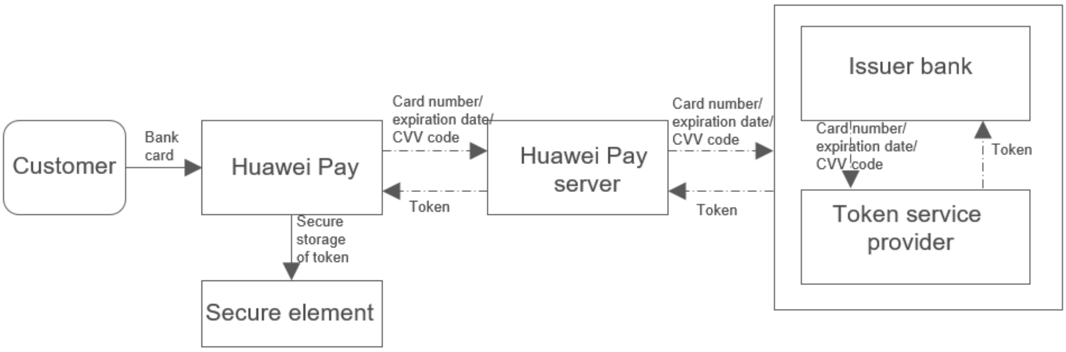
If the user accepts the terms and conditions, Huawei sends the accepted clauses and CVV code to the card issuer and carries out the binding process. The card issuer determines whether to allow the user to add the bank card to Huawei Pay according to the user's device information, such as the name, device model, Huawei mobile phone to which Huawei Pay is bound, and approximate location when the user adds the bank card (if GPS is enabled).

The following operations are performed in the binding process:

- The mobile phone downloads the credential file for the bank card.
- The mobile phone binds the bank card to the SE.

To ensure the security and privacy of cardholders' data, both international organizations and the People's Bank of China have issued relevant standards stipulating that bank card information stored on devices must be replaced with a token. In other words, when the user adds a bank card to Huawei Pay, the card information will be transmitted to the card issuer through the security control measures provided by the issuer. The issuer will then send back an authorized token to Huawei Pay as a substitute for the user's card number, protecting the bank card stored on the phone. The binding process also requires real-name verification by Huawei and the card issuer to ensure that the HUAWEI ID and bank card belong to the same user.

Figure 9-1 Adding bank cards to Huawei Pay



Additional Verification

Card issuers determine whether to further verify bank cards. Depending on the functions supported by card issuers, users can select text message verification as a means of additional verification.

Users can select the contact information archived by their card issuers to obtain text message notification and enter the verification code sent by your card issuer in the Wallet application.

Payment Authorization

The SE permits payment only after receiving authorization from the Huawei Pay-capable mobile phone and determining that the user has passed fingerprint or device password authentication. Fingerprint authentication, if available, is the default authentication mode for payment. Users can opt to use a password at any time. If fingerprint authentication fails once, the system automatically prompts the user to enter the password.

Using Huawei Pay for Contactless Payment

If a Huawei mobile phone is powered on and detects an NFC signal, it displays connected bank cards. The user can access Huawei Pay and select a bank card, or use a specific fingerprint sensor to invoke the payment page when the device is locked.

If the user is not authenticated, no payment information will be sent. Payment information, including the device card number and dynamic security code are used only after the user is authenticated.

Suspending and Removing Payment Cards

Card issuers or payment networks can suspend the payment function of Huawei Pay connected cards or remove the cards from devices even if the devices are not connected to cellular or Wi-Fi networks.

Payment with Biometric Features

Huawei Pay users can authenticate payments with fingerprints and facial authentication, which are stored securely on the device and will not be synchronized to Consumer Cloud. For further security, the payment information is protected by digital certificate signatures.

International Authoritative Financial Certification

Huawei Pay has obtained international PCI-DSS certification as well as VISA PCI-CP and CDCVM security certifications, and complies with authoritative security standards in the payment industry.

9.2 Transport Card

After users add a transport card to HUAWEI Wallet on their Huawei mobile phone, the issuer of the transport card loads their transport card application to the SE of the mobile phone over the air, associates the application with a Supplementary Security Domain (SSD), and then downloads and stores personal data associated with the transport card to the transport card application in the SE. This allows the associated SSD to provide security assurance for the personal data. After a transport card is set up in HUAWEI Wallet, users can top up, query card details and balance, upload the transport card to the cloud, download it from the cloud to the mobile phone, and return the transport card for a refund.

Set Up a Transport Card

After a user adds a transport card to HUAWEI Wallet and pays the service fee, the mobile phone initiates a card registration request. Under the protection of the Secure Channel Protocol (SCP) of the Issuer Security Domain (ISD), Huawei's Trusted Service Manager (TSM), namely, the Secure Element Issuer (SEI) TSM, creates an independent SSD for the transport card, and converts operations on the transport card application to the Application Protocol Data Unit (APDU) instructions according to the GlobalPlatform Card (GP Card) specifications. Under the protection of the ISD SCP, the mobile phone downloads the APDU instructions to the security chip, instantiates the transport card application, and transfers the card instance to the created SSD. The TSM of the transport card issuer, namely, the Service Provider (SP) TSM, manages the SSD key. The SP TSM uses the SSD key to provide SCP encryption protection while downloading personal data, such as the card key (one key per card), to the transport card application in the SE. The transport card is then successfully set up on the mobile phone.

Top Up a Transport Card

After a user adds money to the transport card, the mobile phone initiates a balance top-up request. Once the SP TSM has confirmed that the payment has gone through, it sends a top-up initialization instruction to the card application in the SE, initiating a random number challenge. After receiving the challenge, the card application uses its key to calculate and return the calculation result, which the SP TSM verifies by using the card key. If successful verified, the SP TSM considers the card valid. Then, the SP TSM uses the card key to perform another calculation and encapsulates the calculation result into the card application in the SE downloaded in the top-up instruction. The card also verifies the calculation result. If successfully verified, the card considers the SP TSM valid. Then, the card adds the top-up amount to the card balance. The card key is stored in both the card application in the SE and the hardware encryptor of the SP TSM, with hardware-level security. It is not available to any third party and only the SP TSM of the transport card company can complete the top-up.

Swipe a Transport Card

The NFC controller of the mobile phone allows contactless communication between the transport card application in the SE and the card reader of the transport company. After the transport card application and the card reader are mutually authenticated, the card application deducts an amount from the balance as instructed by the card reader.

Move a Transport Card to the Cloud

If a user temporarily does not need a transport card already set up in HUAWEI Wallet, the user can move it to the cloud. Relevant card data will be stored in the SP TSM. When the card data is backed up to the cloud, the SP TSM delivers a migration instruction to the card application in the SE. The card application then obtains data as instructed, encrypts it and adds the message authentication code (MAC) in the card, before returning the data to the SP TSM. Once received, the SP TSM verifies the MAC, decrypts the data, and stores it. The card data is encrypted and the MAC is added within the card, ensuring confidentiality and integrity during data transmission.

Return a Transport Card

If a user no longer needs a transport card, the user can initiate a card return request in HUAWEI Wallet. During this process, the SP TSM obtains the card balance, and then the SEI TSM deletes the card from the SE. The SP TSM returns the card balance to the user's bank card.

9.3 Secure Keys

The second-generation U key (such as USB key and audio key) is the main network transaction security solution for banks. As a U key is external hardware, it is prone to damage and loss, is not suitable for carrying around, and is not particularly user-friendly. For applications with a mobile payment function, the main security strategy is to bind with mobile phones during transactions through bank payment channels. The transactions are confirmed through SMS messages and are highly vulnerable. Users are therefore concerned that their money may be stolen when making a payment. Huawei secure keys are combined with an independent internal SE, which is an authenticated chip widely accepted in the industry that supports banks' mobile phone certificate services. Huawei secure keys combine traditional plug-in U keys with phones to form portable secure keys in order to provide financial-grade hardware protection for electronic payment.

When a user enables secure keys, the TSM of HarmonyOS establishes an SCP channel with the SE to create a trusted, independent, and secure running space within the SE. The bank application then generates an independent public and private key pair and a certificate in the secure space, requiring the user to enter the PIN on the TUI to protect the generated key data.

When using secure keys, the user has his/her identity authenticated on the TUI first, and then the SE signs the transaction request of the user with the private key generated during the enabling process. When processing the transaction request, the bank verifies and signs the transaction.

When a user deregisters (disables) secure keys, the system directly destroys the public and private key pair stored in the SE. This operation is irreversible.

The private key is stored in the SE throughout its entire lifecycle, from public and private certificate key generation to certificate destruction, and is therefore secure.

View Secure Keys Applications

Secure keys can check application package names and signatures. Only official applications will appear on the management screen to avoid fake and malicious applications. One-click query and management of secure keys applications is allowed using the HUAWEI Wallet APK setting interface.

Secure Keys Switch

The operating system has a switch to avoid backend programs and applications from maliciously invoking the bank certificate. Turning the switch on or off is identical to inserting or removing a traditional USB key. When the switch is turned off, no certificate-related transactions can be conducted. Therefore, secure keys offer a similar experience of being able to control hardware security.

9.4 eID

eID is an ID card application jointly developed by Huawei and the Third Research Institute of the Ministry of Public Security (MPS) of China. eID can be used in the same way as a physical ID card in scenarios recognized by the MPS. It also applies for online identity authentication based on the encrypted information provided by the MPS without disclosing the plaintext ID card information of users. Furthermore, users can swipe their eID to use public transport services if the card reader supports this function. eID also provides authentication interfaces for other third-party mobile phone applications, offering quick and trusted identity authentication.

Users can set up eID in HUAWEI Wallet. During this process, they can use the mobile phone NFC to read the physical ID card and enroll facial information. After liveness detection is performed, the mobile phone encrypts the facial image and uploads it to the server of the MPS. Once it is verified, the server delivers the eID information to the mobile phone. The collection and encryption of face images are implemented in the iTrustee secure operating system, ensuring data security. After eID is set up, the eID information delivered by the server of the MPS is stored in the Integrated Secure Element (inSE) and is accessible only to specific programs. Any intermediate data, such as facial images, will be deleted from the mobile phone after the process is complete.

Huawei mobile devices comply with eID standards and specifications throughout the process, perform full lifecycle management of eIDs, and provide convenient and secure network digital identity services for users. To sum up, Huawei's eID solution leverages the inSE, security camera, and iTrustee secure operating system, thereby providing end-to-end high security protection while users set up, download, use, and deregister eID.

9.5 Car Key

Huawei mobile phones support car keys that comply with the Digital Key Specification released by the Car Connectivity Consortium (CCC).

After a car key is set up, users can use an NFC-equipped mobile phone to unlock the car and start the engine.

Users can also use the application of the car manufacturer to share the car key with relatives and friends. Once authorized by the car owner, users can download the car's digital key and start the engine.

Car owners can withdraw authorization at any time.

When the car owner sets up the car key in their mobile phone, the HarmonyOS TSM establishes an SCP channel with the SE to create a trusted, independent, and secure running space.

Then, the car owner can request the car manufacturer to help download the car's digital key to the mobile phone by using a TSM, turning the mobile phone into a car key.

The car's digital key is stored in an industry-certified and recognized independent SE, which has financial-grade security.

After a user restores factory settings, the mobile phone automatically disables and deletes the car key for security purposes.

10 Building the Resilient Security System Architecture of HarmonyOS

This chapter describes how to build the resilient security system architecture of HarmonyOS, with reference to cutting-edge security architectures such as the ZTNA and cyber resilience network architecture. It introduces HarmonyOS's security and trustworthy engineering capabilities, Singular Security Lab, bug bounty program, and the security emergency handling process and mechanism. All these means serve to "rid the system of security potential vulnerabilities, make it hard through in-depth defense to exploit existing vulnerabilities, and remediate vulnerabilities and restore services as soon as possible."

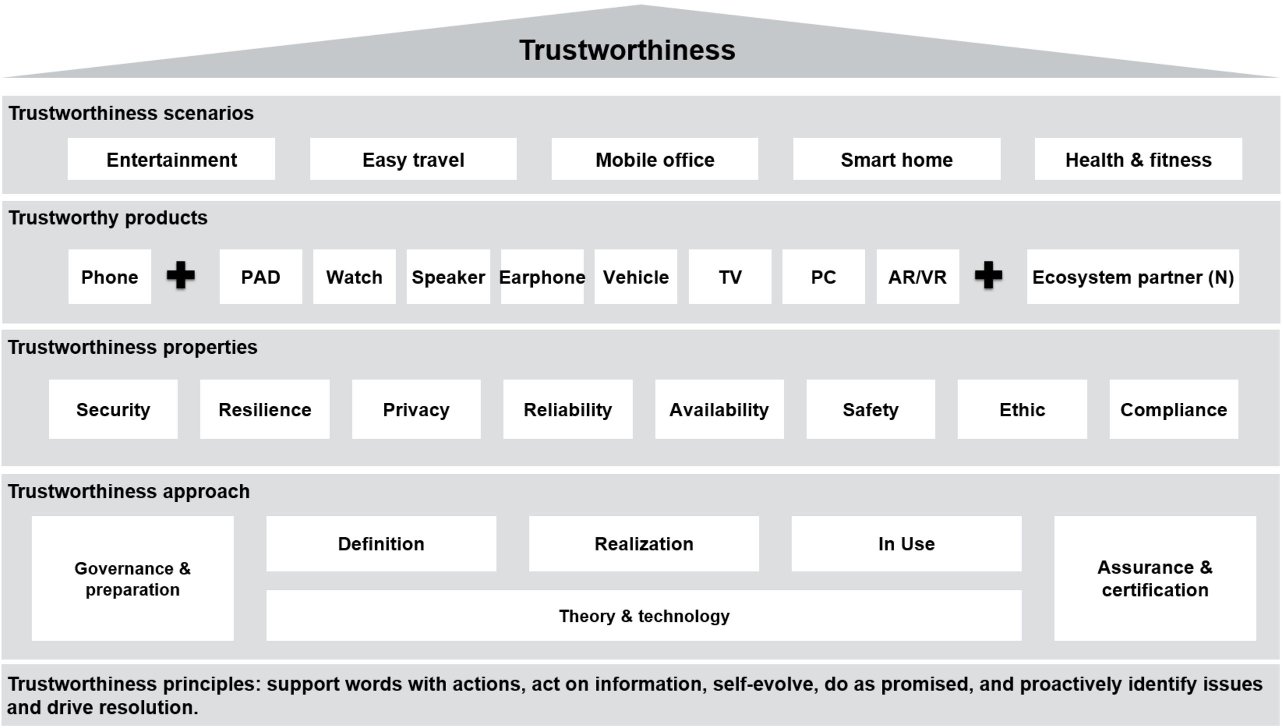
10.1 Trustworthiness Engineering of HarmonyOS

Society is evolving towards a fully-connected smart world. Trends such as full-scenario, full-connection, and intelligence have put forward unprecedented requirements for consumer product trustworthiness. Trustworthiness is a basic concern for customers when purchasing a product. Trustworthiness is not just about producing quality results. It is about the entire quality process, including designing and creating products in an inherently quality way. Trustworthiness stems from verifiable quality – both in process and results.

Cyber security and privacy protection have been and will always be Huawei's top priorities. Huawei will build a stronger management system based on the ISO 9000 quality management system and ISO/IEC/IEEE 15288 and 12207 standards system engineering and software development processes. This helps ensure that employees keep trustworthiness in mind, collaborate and innovate based on Huawei's trustworthy processes, provide customers with trustworthy and high-quality products, and make continuous improvements.

Huawei has conducted research across 150 documents covering mainstream security standards, processes, specifications, and guidelines in the industry as well as laws, regulations, white papers, and academic papers. The research shows that the standards have respective focuses and that one individual standard may not be comprehensive. What trustworthiness standards are needed for the development of devices and services favored by consumers in the future digital world? To build a "bridge" between design and trust so that personnel defining, designing, using, and operating products can have a common understanding of trustworthiness, Huawei outlines its trustworthiness framework based on its large-scale development and operation experience, systemic knowledge in designing complex products, and capabilities in system architecture design. Based on the industry's understanding of system engineering, Huawei defines its trustworthiness framework using four principles: explainable, implementable, verifiable, and embracing industry consensus.

Figure 10-1 Huawei's trustworthiness framework



Trustworthiness shall be integrated into every product and solution provided for consumers to ensure high quality. This consists of:

Security: The product can protect the confidentiality, integrity, and availability of services and data.

Resilience: The system has the ability to stay in a known state, even if in a degraded state, while under attack and to rapidly recover and continue its evolution after being attacked.

Privacy: Protecting privacy is a regulatory requirement, and also a reflection of values. Users are able to appropriately control how their data is used. Information use policies are transparent to users. Users are able to appropriately control whether and when to receive information based on their own needs. In addition, there is a comprehensive privacy protection mechanism and corresponding capabilities.

Safety: System failures do not cause any unacceptable risk of physical injury or of damage to the health of people, either directly, or indirectly as a result of damage to property or to the environment.

Reliability and availability: Long-term, fault-free operations are ensured throughout the lifecycle of products. They can rapidly recover and self-manage, and provide consistent services as expected.

Ethic: AI shall be used to benefit human beings and serve for the good of society and environment. AI systems shall not increase the prejudice and discrimination against disadvantaged and marginalized groups, and diversity in AI data sets is ensured through process requirements.

Compliance: Countries' or regions' requirements regarding taboos, accessibility, and protection for minors are complied with.

In the definition and complete implementation phases of each product, trustworthiness and its controls are integrated into innovation. This enables trustworthiness to be injected from the very beginning. In addition, the process from a product's concept phase to product delivery to customer sites is complete, consistent, and bidirectionally traceable. Proper confidentiality protection (permission separation, trust, and behavior monitoring) can be provided, if needed, to ensure that the product is not forged or tampered with, its deployment, maintenance, disposal process and tools are trustworthy, and sensitive data is not leaked.

10.2 Singular Security Lab

The security research and penetration test team led by top security researchers in the industry continuously assesses cyber security and privacy risks of HarmonyOS products and solutions through the following activities to identify and eliminate risks in advance and ensure privacy and security:

1. Continuously conduct research on security technologies, follow up the technology dynamics in the academia, industry, and security researcher communities, and understand state-of-the-art security technologies.
2. Introduce new security technologies during product and solution development in a timely manner and apply them to security tests of products and solutions.
3. Conduct penetration testing on HarmonyOS products and solutions from the perspective of penetration testers, develop systematic solutions for improvement and push for their implementation in products, and assist product teams in building in-depth security defense systems.

10.3 Bug Bounty Program of HarmonyOS

Huawei attaches great importance to the security of its products and services and is constantly enhancing product and service security by working with security communities and industries. In this context, Huawei launches the Bug Bounty Program. Huawei promises that issues reported by each reporter will be tracked, analyzed, handled, and replied by dedicated personnel in a timely manner.

The program covers Huawei phones and tablets running HarmonyOS, related smart devices developed by Huawei, as well as HarmonyOS and the distributed features for interaction between devices. The following table lists the products involved:

Product Type	Category and Model
Mobile phone	Mate/P series
Tablet	MatePad
Wearable	Smart watch
IoT	Smart TV, router, speaker, etc.

Note: The list is subject to update.

For detailed reward rules, visit https://device.harmonyos.com.

10.4 Security Emergency Response of HarmonyOS

Huawei sets up the Security Response Center (SRC) to quickly mitigate HarmonyOS security risks and issues. The SRC handles vulnerabilities in HarmonyOS in compliance with ISO/IEC 30111 (standard for vulnerability handling processes) and ISO/IEC 29147 (standard for vulnerability disclosure). All potential vulnerabilities are handled in line with the vulnerability response process.

The following is a brief description of the vulnerability response process carried by the HarmonyOS team:

1. Report collection: Actively monitor and receive external reports on security vulnerabilities and issues and initiate the vulnerability response process.
2. Issue verification: Coordinate resources to check if the reported issues are security vulnerabilities and assess the risk level.
3. Solution development: Develop solutions to mitigate the risks of vulnerabilities and remediate the vulnerabilities.
4. Vulnerability disclosure: After the vulnerabilities are fixed, coordinate the disclosure of security issues with security researchers.
5. Feedback collection: Collect and summarize feedback from internal and external customers and provide important cases to improve the product development process.

To prevent early disclosure from bringing harm to consumers and industry partners, Huawei will strictly control the scope of vulnerability information during the entire vulnerability handling process and will require vulnerability reporters to keep the information confidential until vulnerabilities are remediated and officially disclosed.

11 Ecosystem for Enabling and Opening Up HarmonyOS Security Capabilities

HarmonyOS provides security capabilities for applications in the ecosystem in the forms of APIs, kits, and SDKs. Within these capabilities are dedicated CBBs, security modules, and independent chips for ecosystem devices.

11.1 Opening Up of Security Capabilities to Applications in the HarmonyOS Ecosystem

Currently, HarmonyOS provides the following capabilities:

Secure Storage of Short Data

Applications usually contain some short data (64 bytes or less) that is sensitive and important, such as user names, passwords, credit card information, application tokens, etc. Such data has high security requirements.

HarmonyOS's key short data (asset) storage function can securely store, delete, update, and query sensitive short data.

Its advantages are as follows:

1. Convenient: The function is applicable to a broad range of scenarios where sensitive data needs to be stored or managed.
2. Secure: The function stores keys in a TEE, adopts optimal encryption and decryption algorithms, and encrypts stored data to ensure key, algorithm, and data security.
3. Efficient: Using the function can decrease the time required for developing security features, such as the selection and iteration of encryption and decryption algorithms, and the storage and management of key asset data.

Classified File Protection

Based on data classification standards, HarmonyOS establishes lifecycle security control policies and protection requirements for data. Developers shall adopt corresponding security protection measures in line with policies and requirements.

Classified file protection provides the capability for classifying and labeling data, as well as setting and reading files at the S0 to S4 risk levels. The following table lists the data risk levels and the corresponding definitions.

Risk Level	Risk Label	Definition	Example
Critical	S4	Special data types defined in industry laws and regulations, involving the most private personal information, or data that may cause significant negative impact on individuals or organizations if leaked	Health status and personal credit card information
High	S3	Data that could cause a serious negative impact on individuals or organizations if leaked	Personal real-time precise positioning information and workout routes
Medium	S2	Data that could cause significant negative impact on individuals or organizations if leaked	Detailed personal addresses, names, and nicknames
Low	S1	Data that could cause limited negative impact on individuals or organizations if leaked	Nationalities, places of birth, and educational information
Public (no risk)	S0	Public data that has no adverse impact on individuals or organizations	Publicly released product introductions, public meeting information, and open-source code

Application scenario: Classified file protection can be used to set the risk level for data, after which the terminal device performs corresponding encryption safeguards on the data. Developers can adjust the protective measures within the scope of the rules.

The advantages of this function are as follows:

1. Secure: Risk levels can be set for files, based on which corresponding protection solutions are provided. Identity authentication solutions also vary according to the risk level.
2. Efficient: Using the function can decrease the time required for developing security features; classifying and labeling data can improve data security. This capability also reduces the amount of work required to adapt bottom-layer security features when migrating phone applications to other devices.

Local Facial Authentication

HarmonyOS provides a local facial authentication capability, which can perform facial comparisons using images taken with the front camera and running facial feature algorithms.

Application scenario: This function can be used for login, payment, or other scenarios where facial authentication is required. 3D facial authentication, supported by certain device models, is high in security and can be used for large payments.

The advantages of the function are as follows:

1. Fast: It allows for quick facial comparisons using deep neural network algorithms.
2. Efficient: Using the function can decrease the time required for developing algorithms and can make the application more lightweight.
3. Versatile: The function can be used across a broad range of scenarios.

Adoption of Opening-Up HarmonyOS Capabilities by Ecosystem Partners

HarmonyOS provides **trusted, accurate, efficient, and versatile** security capabilities for applications in the ecosystem and continuously outputs end-to-end professional solutions and system services to "facilitate the development and use of these applications". As an important part of HarmonyOS security, the **data security** and **local authentication** capabilities are widely used by many ecosystem applications and are highly recognized:

The **data security** capabilities (including "secure short data storage" and "classified file protection") offer protection for user data security and system data access. **Secure short data storage** can securely store, delete, update, and query sensitive short data, which is applicable in a broad range of scenarios where sensitive short data needs to be stored and managed.

Classified file protection sets data security control policies and protection requirements throughout the data lifecycle based on data risk classification, while corresponding security protection measures need to be adopted by developers. **Data security capabilities are playing increasingly important roles in a large number of applications (such as banking, wealth management, and payment applications) where strict data security and privacy protection are required.**

Local authentication is developed based on deep neural networks and 3D structured light technologies to provide secure and reliable facial authentication capabilities on local ends. This includes high-precision facial comparison and liveness detection at high security levels. By now, **it has been used in the applications of Alipay, China Minsheng Banking Corp., Ltd (CMBC), Industrial and Commercial Bank of China (ICBC), etc., invoked hundreds of millions of times for quick user login, authentication, and more. In addition, the number of monthly active users has reached tens of millions**.

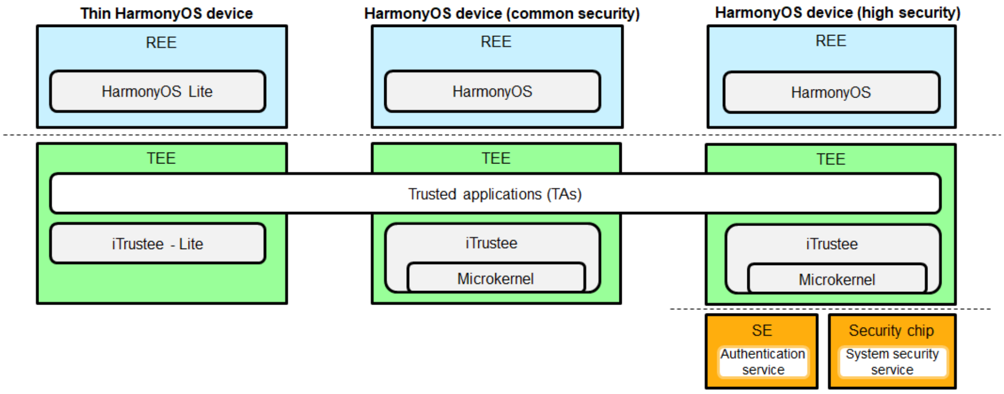
The security capabilities provided for ecosystem applications are actively used in financial and wealth management, social communication, media & entertainment, and mobile office fields. They will be applied to more service scenarios, constantly protecting security.

11.2 Opening Up of Security Capabilities to Devices in the HarmonyOS Ecosystem

Opening-Up of HarmonyOS's TEE Security Capabilities

As described in chapter 5 Hierarchical Security System Architecture for "Right Devices" in HarmonyOS, HarmonyOS builds a powerful TEE solution, bringing great value to the development and deployment of security services.

Figure 11-1 HarmonyOS TEE solution



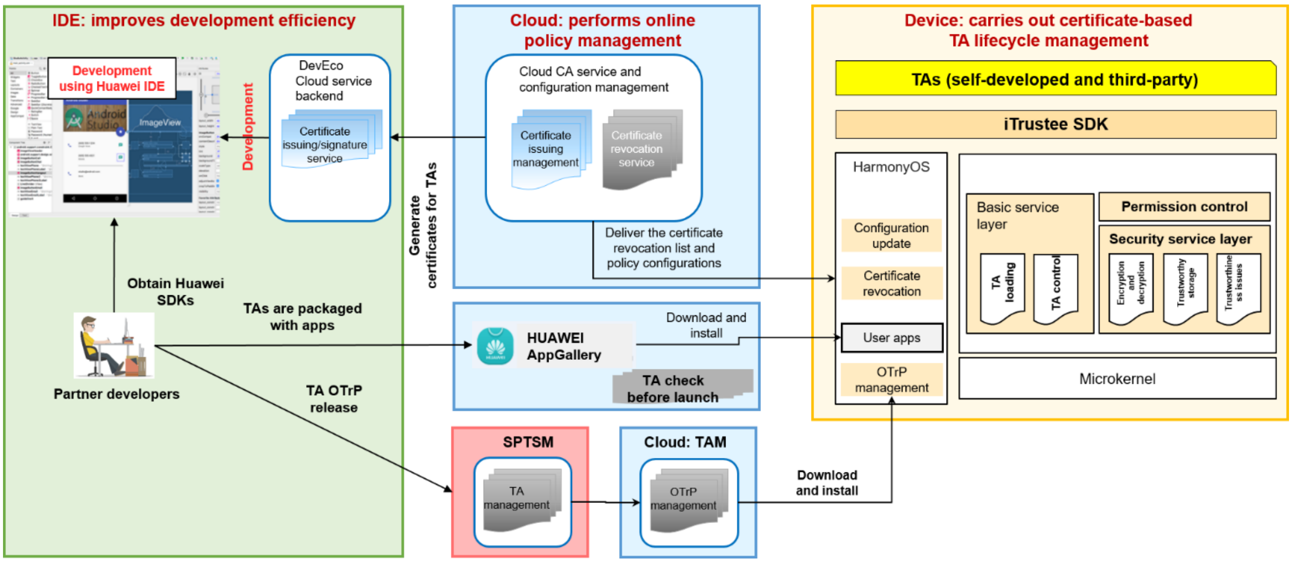
HarmonyOS builds a TEE solution that accommodates various chip architectures and product forms, and shields underlying differences. Trusted applications developed based on the HarmonyOS TEE can be deployed across products, realizing "one-end development and multi-end deployment."

To help ecosystem partners develop secure applications and services on HarmonyOS, HarmonyOS has its TEE capabilities opened to ecosystem partners. There are two ways to use TEE capabilities:

- Use the Security kit (on the REE side) for basic security capabilities, which are applicable to common security scenarios.
- Deploy trusted applications in HarmonyOS TEE for enhanced security capabilities, which are applicable to security scenarios of vertical industries.

In the second way, HarmonyOS provides the matching IDE and deployment service for trusted applications. Partners can use them to implement the development and deployment of their applications on HarmonyOS. The following figure shows the logic.

Figure 11-2 Architecture of HarmonyOS's TEE capabilities opened to the ecosystem


IDE: improves development efficiency
Development using Huawei IDE
Development
DevEco Cloud service backend
Certificate issuing/signature service
Obtain Huawei SDKs
TAs are packaged with apps
Partner developers
TA OTrP release
Generate certificates for TAs
Cloud: performs online policy management
Cloud CA service and configuration management
Certificate issuing management
Certificate revocation service
Deliver the certificate revocation list and policy configurations
HUAWEI AppGallery
Download and install
TA check before launch
SPTSM
TA management
Cloud: TAM
OTrP management
Download and install
Device: carries out certificate-based TA lifecycle management
TAs (self-developed and third-party)
iTrustee SDK
HarmonyOS
Configuration update
Certificate revocation
User apps
OTrP management
Basic service layer
TA loading
TA control
Permission control
Security service layer
Encryption and decryption
Trustworthy storage
Trustworthiness issues
Microkernel

A Acronyms and Abbreviations

Table A-1 Acronyms and abbreviations

Acronym/Abbreviation	**Full Spelling**
3D	three dimension
AES	advanced encryption standard
AI	artificial intelligence
API	application programming interface
APK	Android package
ARM	advanced RISC machines
CE	credential encryption
CFI	control flow integrity
DE	device encryption
ECC	elliptic curve cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
eMMC	embedded multimedia card
HarmonyOS	HarmonyOS
GP	GlobalPlatform
HMAC	hash-based message authentication code
HUK	hardware unique key
HUKS	HarmonyOS universal keystore service
ID	identifier
IMEI	international mobile equipment identity
inSE	integrated Secure Element
IoT	Internet of Things

Acronym/Abbreviation	**Full Spelling**
IT	information technology
JOP	jump-oriented programming
LTO	link-time optimization
MAC	Media Access Control
NFC	near-field communication
NIST	National Institute of Standards and Technology
OS	operating system
OTA	over the air
PAN	Privileged Access Never
PIN	personal identification number
PKI	public key infrastructure
POS	point of sale
PXN	Privileged eXecute Never
REE	rich execution environment
ROM	read-only memory
ROP	return-oriented programming
RSA	Rivest-Shamir-Adleman
RPMB	replay protected memory block
SD	secure digital memory card
SDK	software development kit
SHA	secure hash algorithm
SN	serial number
TA	trusted application
TEE	trusted execution environment
TLS	Transport Layer Security
TUI	trusted user interface
UID	user identifier
mmap	memory mapped
VDSO	virtual dynamic shared object
OEM	original equipment manufacturer
CE	Credential Encryption

Acronym/Abbreviation	Full Spelling
SECE	Sub-Enhanced Credential Encryption
ECE	Enhanced Credential Encryption
SCP	secure channel protocol
SSD	Supplementary Security Domain
SE	secure element
SP	select partner
TSM	trusted service manager
APDU	application protocol data unit

Change History

Date	Description
	First release.