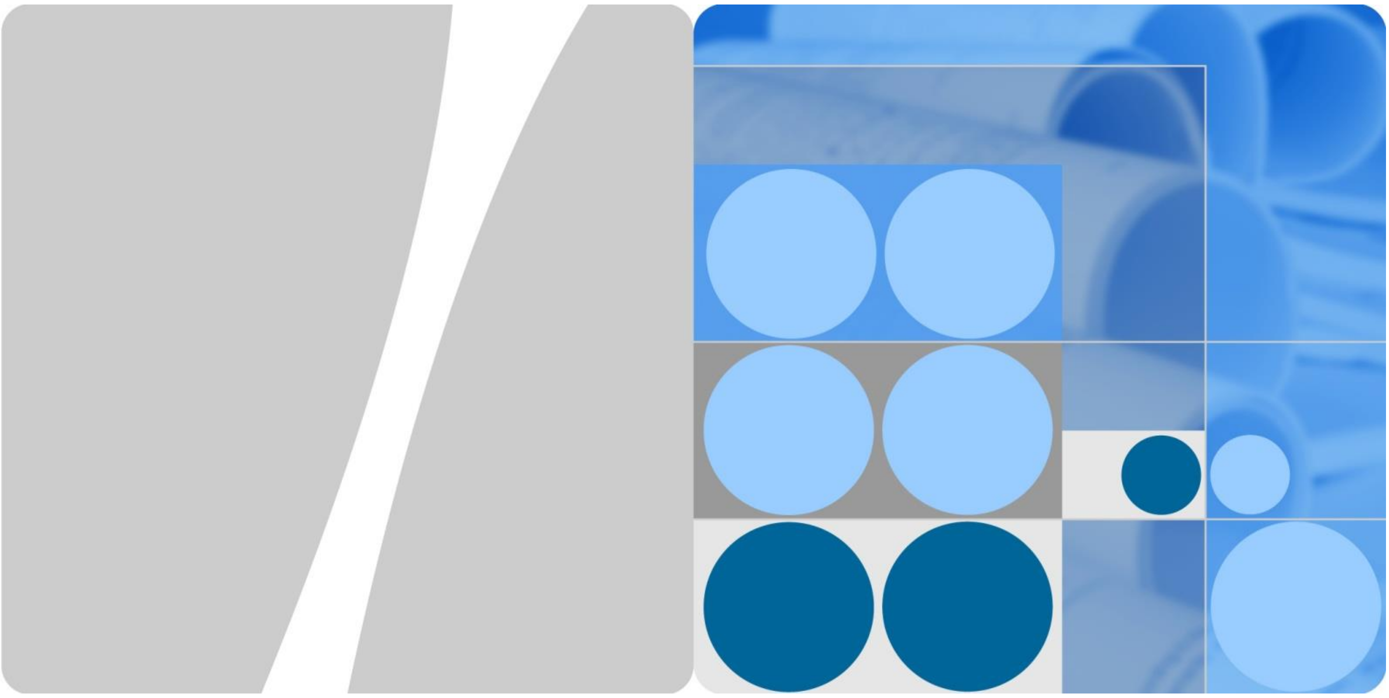


EMUI 8.0 Security Technical White Paper

Issue 1.0

Date 2017-10-31

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: http://www.huawei.com

PSIRT Email: PSIRT@huawei.com

Contents

Note: * indicates a feature not supported by all devices. Supported features vary depending on device models or market features in difference countries. For details, refer to specific product description.

Figures

1 Overview

As the mobile Internet develops, mobile smart devices become major network access devices and store much user data, including users' personal information. In addition, an increasing number of apps from uncontrollable sources are installed on the devices. Due to these reasons, privacy and security issues become increasingly prominent, and security issues in mobile smart devices become consumers' major concern.

Apps on mobile smart devices come from various channels, some pre-installed by vendors and some possibly from third parties. Users may download malicious apps. Malicious apps may infringe upon users' privacy or steal users' assets, bringing various potential security threats.

Huawei attaches much importance to the security of mobile smart devices to provide chip-level security assurance while ensuring good user experience. This white paper systematically describes the security and privacy protection solutions delivered by the Emotion UI (EMUI), with a focus on the enhancement and supplementation made by EMUI on the basis of Android.

EMUI is a deeply customized mobile device system based on Android. It is applied to products running difference hardware chip platforms. Therefore, the security implementation methods may differ depending on hardware and chips. For the actual specifications of different devices, refer to corresponding product manuals.

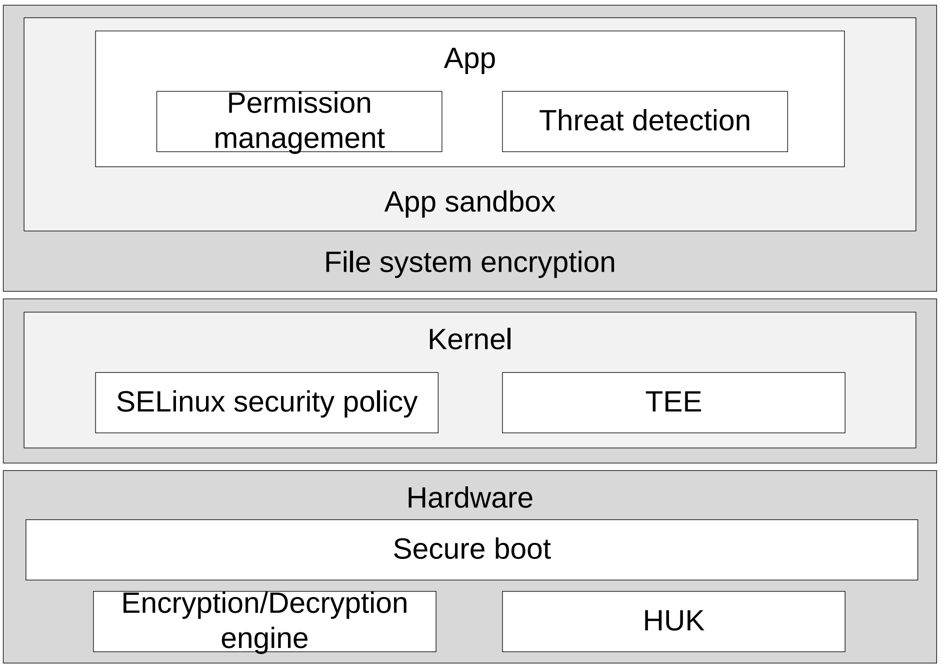
Security is a systematic project. EMUI provides end-to-end security protection from the hardware, system, and app to the cloud (as shown in Figure 1-1), including security and privacy protection for the hardware chip, system kernel, data, app, network, payment, cloud service, and device management.

EMUI provides a secure boot mechanism from the underlying hardware chip to prevent the EMUI ROM image from being tampered with. The ROM image can normally run on a device only after passing signature verification, which ensures secure boot for the boot loader, recovery, and kernel image. In addition, the Android native system provides verified boot to ensure the secure boot of the Android system and prevent tampering and malicious code implantation, thereby ensuring system security from the hardware chip to the Android boot.

To ensure data security, user data is encrypted using a hardware-based hardware unique key (HUK) and user lock screen passcode. Data files of different apps are stored in the directories of the corresponding apps, so that the files of one app cannot be accessed by other apps. The data erasure function is provided for permanently erasing data during device recycling or factory default restoration to prevent illegitimate data restoration. The combination of EMUI with cloud services helps users back up and synchronize data to ensure data security.

For app security, apart from the Android security sandbox and permission management mechanisms, EMUI pre-installs the Phone Manager to provide virus scanning, block and filter, traffic management, notification management, and other functions. With these functions, the system can automatically detect viruses and Trojan horses in apps and provide fine-grained permission, traffic, and notification management functions.

Figure 1-1 EMUI security architecture



- Hardware chip: secure boot, hardware encryption/decryption engine, HUK, device attestation key pair, Trusted Execution Environment (TEE), secure storage, and fingerprint authentication
- System security: integrity protection, SELinux access control, kernel address space layout randomization (KASLR), and system software upgrade
- Data security: lock screen passcode protection, file system encryption, SD card lock and encryption, and secure erasure
- App security: app signature, app sandbox, runtime memory protection, secure input, app threat detection, malicious website detection, and traffic management
- Network security: VPN, SSL/TLS, WPA/WAP2, and secure Wi-Fi detection
- Communications security: defense against rogue base stations, block and filter, short message encryption, and device interconnection security
- Payment security: Huawei Pay, Secure keys, payment protection center, and verification code short message protection
- Internet cloud service security: Huawei ID, code scanning login, account protection, Huawei ID message, HiCloud, account-based keys, and HiCloud cloud backup
- Device management: Find My Phone, activation lock, mobile device management (MDM), MDM API, and device certificate authorization
- Privacy protection: permission management, location service, notification management, app lock, file safe, and Private Space

2 Hardware Security

This chapter describes Huawei devices' hardware chip security, including the following security features:

- Secure boot
- Hardware encryption/decryption engine
- HUK
- Device attestation key pair
- Hardware Random Number Generator (RNG)
- TEE
- Secure storage
- TUI
- Fingerprint authentication

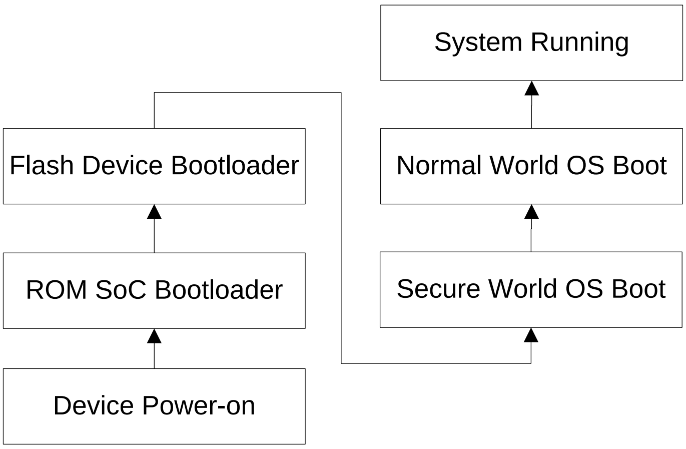
Secure Boot

Secure boot prevents the loading and running of unauthorized apps during boot. The bootstrap uses a public key to verify the digital signatures of software, ensuring the trustworthiness and integrity of the software. Only image files that pass the signature verification can be loaded. Examples of these files include the boot loader, kernel image, and baseband firmware. If the signature verification fails during boot, the boot process is terminated.

The bootstrap is a boot program in the hardware chip and is called the ROM SoC Bootloader. This code snippet is written into the ROM inside the chip during chip manufacturing, not modifiable after delivery, and first executed after device power-on.

The ROM SoC Bootloader performs basic system initialization and then loads the Flash Device Bootloader from the flash storage chip. The ROM SoC Bootloader uses the public key in the eFuse space (fuse blowout protects public key data from being tampered with) of the main chip to verify the digital signature of the Flash Device Bootloader image. Flash Device Bootloader is executed only after the verification succeeds. Then the Flash Device Bootloader loads, verifies, and executes the next image file. The similar process is repeated until the entire system is booted, thereby ensuring trust chain transfer and preventing unauthorized programs from being loaded during the boot.

Figure 2-1 Secure boot



Hardware Encryption/Decryption Engine

The chip provides a high-performance hardware encryption/decryption acceleration engine which supports the following algorithms:

- 3DES
- AES128 and AES256
- SHA1 and SHA256
- HMAC-SHA1 and HMAC-SHA256
- RSA1024 and RSA2048
- ECDSA-256

HUK

The HUK is the hardware trust root stored in the eFuse space in the chip, accessible only to the hardware encryption/decryption engine, and variable depending on the device.

Device Attestation Key Pair

To prove that the device is trusted, EMUI derives Rivest-Shamir-Adleman (RSA) and elliptic curve cryptography (ECC) public-privacy key pairs that are bound with the device hardware and service information to prove the device identity in different scenarios.

Hardware RNG

Random numbers used for generating session keys and initialization vectors (IVs) and those for anti-replay require high entropy values. To reach an acceptable security level, Huawei mobile phone chips provide a NIST SP 800-90A-compliant RNG named CTR_DRBG. The seeds of this RNG come from NIST SP 800-90B-compliant hardware entropy sources.

TEE

EMUI supports the TEE secure operating systems of various chip platforms. On the HiSilicon platform, TrustedCore is a TEE system designed on the basis of ARM TrustZone. The

TrustZone-based hardware isolation of the TEE isolates the memory, running environment, and screen from the external Android system to prevent attacks by malicious software.

Secure Storage*

The secure storage function is a TEE-based secure file system (SFS) for the secure storage of keys, certificates, personal privacy data, and fingerprint templates.

The trusted application (TA) running in the TEE uses a secure storage API to encrypt and store data in the SFS. The encrypted data is accessible only to the TA.

The AES256 hardware encryption/decryption used by the secure storage function is compatible with the GlobalPlatform TEE standard. The secure storage keys are derived by the HUK and not sent outside of the TrustZone. Data encrypted using the keys cannot be decrypted outside of the TrustZone.

EMUI further provides a Flash-based Replay Protected Memory Block (RPMB) to prevent system data from unauthorized deletion and access. The RPMB is directly managed by the TEE and bound with the keys derived by the HUK. Only the TEE can access the RPMB-protected data, and the external Android side does not provide any interface for accessing the RPMB. The RPMB uses built-in counters, keys, and the HMAC verification mechanism to defend against replay attacks and prevent data from being malicious overwritten or tampered with.

TUI*

In app environments in Android, the payment amounts or input passwords displayed by apps may be hijacked by malicious apps. For this reason, the TEE Trusted UI (TUI) display technology (compliant with GlobalPlatform standards) that can prohibit screenshots is provided for Android to protect content displayed by TAs by prohibiting access from Android. In this way, the TUI prevents the hijacking and tampering of displayed data and input by malicious Android apps.

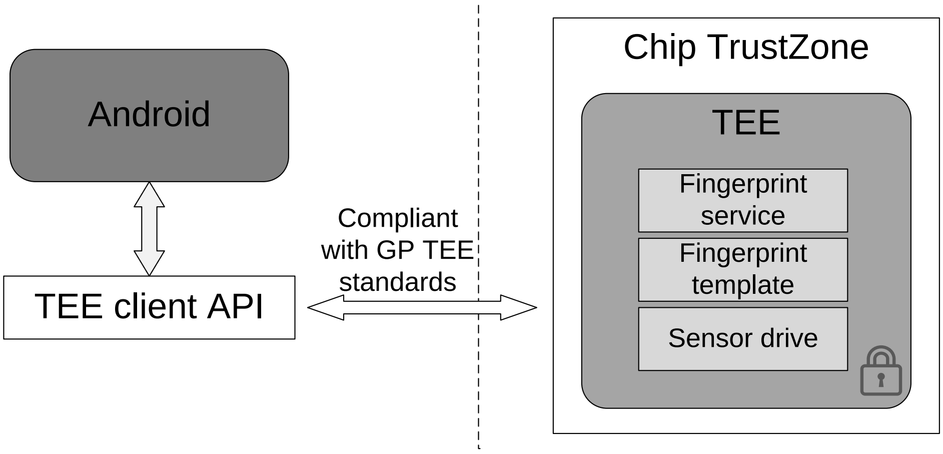
Fingerprint Authentication

A fingerprint is a person's inherent physiological characteristic mainly used for identity authentication and other important occasions.

EMUI implements all fingerprint-related processing, such as image pre-processing, characteristic extraction, template generation, entry, and authentication, in the TEE based on the chip hardware isolation of the TrustZone. The external Android fingerprint framework is only responsible for fingerprint authentication initiation and authentication result, but does not handle the fingerprint data. Any external third-party Android app cannot obtain the fingerprints or send the fingerprint data to the outside.

The fingerprint template data is stored in the TEE secure storage or RPMB using AES256 encryption to prevent the leak of the fingerprint encryption key and users' fingerprint data.

Figure 2-2 Fingerprint security framework



3 System Security

This chapter describes device security. EMUI enhances the security of the Android system based on Android's own security mechanisms, such as Linux Kernel LSM, permission control, and process protection.

Integrity Protection

Verified Boot

EMUI supports the verified boot function of Android and provides block device–based integrity check to prevent permanently-resident rootkits from holding root permissions. This function helps users ensure that the device status at startup is the same as the last time it is used.

Huawei Kernel Integrity Protection (HKIP)*

To better protect kernel integrity, EMUI uses hardware Hypervisor virtualization technologies to realize real-time kernel integrity protection in order to protect key positions through measures such as preventing kernel code fields and important system registers from being tampered with and preventing malicious code injection under privilege mode.

Kernel Security

- SELinux access control

  EMUI supports the Android-native SELinux feature and implements mandatory access control (MAC) on all processes, files, and operations. The access control policy cannot be modified by any third party and is protected during boot. The SELinux can prevent processes from writing and reading protected data, bypassing kernel security mechanisms, or attacking other processes.

- KASLR

  EMUI supports KASLR to randomize the loading address of the loadable kernel module (LKM) and the initial addresses of the kernel stack.

  The KASLR technique makes the memory address space unpredictable and makes it impossible for attack code to hard code addresses in the memory, which further enhances system kernel security.

System Software Upgrade

EMUI supports over the air (OTA) upgrade to fix possible vulnerabilities in a timely manner. The upgrade package signature is verified during system software upgrade. Only verified upgrade packages are considered legitimate and can be installed.

In addition, EMUI provides software upgrade control. At the beginning of OTA upgrade after a software package is downloaded, EMUI applies for upgrade authorization by sending the digest information of the device identifier, the version number and hash of the upgrade package, and the device upgrade token to the OTA server. The OTA server verifies the digest before authorization. If the digest verification succeeds, the OTA server signs the digest and returns it to the device. The upgrade can be implemented only after the device passes the signature verification. If the device fails the signature verification, an upgrade failure is displayed to prevent unauthorized software upgrade, especially upgrade using vulnerable software.

4 Data Security

This chapter describes EMUI data security protection. The EMUI file system is divided into the system partition and user partition. The system partition is read-only, isolated from the user partition, and inaccessible to common apps. For data stored in the user partition, the system provides file-based data encryption and directory permission management to restrict data access between apps. Some Huawei devices support SD cards which store much user data. Once a device gets lost, the SD card can be plugged out and inserted into another device for data reading, which causes data leak. To resolve this problem, EMUI provides the SD card lock and encryption functions to secure the SD card data.

Lock Screen Passcode Protection

The lock screen passcode is protected by the HUK. All processing of lock screen passcodes created or verified by users is done in the TEE. EMUI restricts users' consecutive failed password attempts to prevent brute-force cracking of lock screen passcodes. All key processing is done in the TEE, and the Android keystore module is used to provide secure services externally. To prevent malicious use of keys, the system implements access control mechanisms, such as identity authentication and lifecycle management, as early as key creation, allowing only authenticated callers to use keys.

File System Encryption

In the case of the loss of a mobile phone, to prevent unauthorized users from launching physical attacks (for example, directly reading the Flash) to obtain device data and cause data leak, EMUI provides data encryption protection for the user file system.

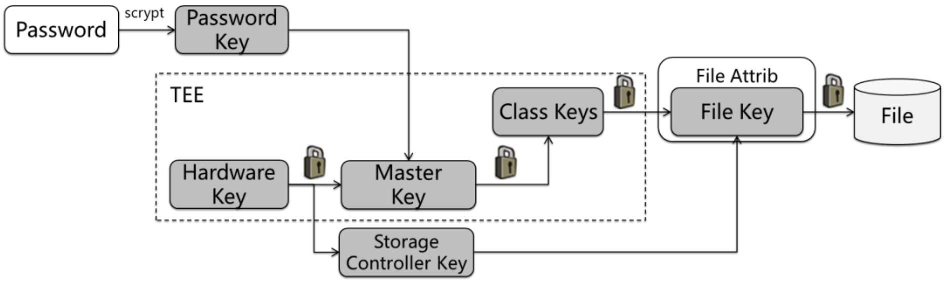
EMUI uses the kernel encryption file system module and hardware encryption/decryption engine to deliver Android-based (Android *N* or later versions) file-level encryption by utilizing the AES256 algorithm in XTS mode. In addition, the key for encrypting user password is protected using the user lock screen passcode and a key derived by the HUK to prevent unauthorized access to stored data. During boot, a device is encrypted and locked by default, allowing only specific apps, such as phone call and alarm, to run. To use other functions or access user data, users must unlock the device. Moreover, the device provides the dictionary attack prevention mechanism to prevent brute-force password cracking.

Data encryption keys are only processed in the TEE security quarantine area on the main chip. After the screen is unlocked, only the cyphertexts of the keys appear outside of the TEE, and the cyphertexts must be used with storage controller hardware keys derived from HUK at the time of power-on. As a result, nobody can obtain the keys illegally on the Android side.

To secure user data and app experience, the storage area of the Android system (Android *N* or later versions) is divided into the device encrypted (DE), credential encrypted (CE), and non-encrypted (NE) partitions. App data is stored in the CE partition by default to ensure app security.

- DE partition: Data in this partition can be accessed when the screen is locked after power-on. Such data includes data related to Wi-Fi authentication, Bluetooth match, alarm, ring, and others.
- CE partition: Data in this partition can be accessed only after the user is verified (by entering the lock screen passcode). Such data includes accounts/passwords, contacts, short messages, calendar, call history, and location information.
- NE partition: Data in this partition is not encrypted, which is a rare case. Such data includes the OTA upgrade package.

Figure 4-1 File encryption



SD Card Lock and Encryption*

The SD card stores a great deal of personal information, such as photos, diaries, voice and video data. It is a pluggable storage device and can be inserted in another device to read SD card data. Therefore, securing the SD card is vital.

EMUI provides the SD card lock function based on standard SD card protocols and allows users to set, change, or cancel passwords for external SD cards. The password is a string of no more than 128 bits, consisting of digits, uppercase letters, lowercase letters, or any of their combinations.

The SD card provides static data protection. Specifically, the SD card is automatically locked once it is removed. Then, the data stored in the SD card can be accessed only after the correct SD card password is entered. To improve the ease of use, the SD card lock supports the remember password function to simplify the use on the same device. If the user forgets the password of the SD card, the user can forcibly erase the password and SD card data.

EMUI has the SD card encryption function and can encrypt files in SD cards to prevent private information leakage due to phone or SD card loss and allow users to lend their SD card to others care-free. Encrypted files can only be viewed and used on the owner's device. Others can use the SD card for storage but cannot view encrypted files.

SD card encryption uses the kernel encryption file system module and hardware encryption/decryption engine to deliver file-level encryption by utilizing the AES256 algorithm. In addition, the key for encrypting user password is protected using the user lock screen passcode and a key derived by the HUK to prevent unauthorized access to stored data.

Secure Erasure

Common factory default restoration operations do not ensure that all data stored on a storage device is deleted. For efficiency, users usually use the logical address deletion method. However, this method does not clear the physical address space and the data can be restored.

EMUI allows users to thoroughly erase user data from devices when restoring factory settings. EMUI uses the internal storage space formatting method to overwrite file encryption keys to totally erase the keys, so that ciphertext user data cannot be decrypted. This data erasure method ensures data security for users who want to resale or dispose of devices.

5 Application Security

This section focuses on the security of application programs. Application programs can be obtained from various channels, so users may download malicious applications at any time. If not properly handled, the malicious applications may bring security risks to the security and stability of the system, and may bring damages to user personal data and even personal property.

Therefore, EMUI provides functions like threat detection and mitigation, and malicious website detection to ensure security of the application programs.

Application Signature

Only application programs with complete signatures can be installed in EMUI. Application signatures can be used to verify the integrity and legitimacy of applications. The system will verify the application signature to check whether the application is tampered with before installing the application.

The system will also verify the application signatures before updating the pre-installed applications or user-installed applications (UIAs). The applications can be updated only when their signatures are the same as those of the applications to be updated, which prevents malicious applications from taking place of the existing applications.

Android application programs use self-signed certificates, and such certificates do not need to be signed by certification authorities. Code signatures are used to:

- Check application integrity (whether the application has been tampered with) and legitimacy (whether the application is from an authorized developer).
- Check whether self-signed certificates are consistent before updating the applications. The applications can be updated only when their new and old signature certificates are consistent.
- Establish trust between applications. Based on mutual trust, applications of the same user ID can securely share code and data.

Application Sandbox

EMUI uses the application sandbox mechanism originally provided by Android, making sure that all applications run in the sandbox and are isolated from each other to ensure security when running.

Runtime Memory Protection

Malicious programs will probably obtain memory addresses by viewing the memory if the allocated memory addresses are relatively fixed during program running. To solve this problem, EMUI supports Android's original address space layout randomization (ASLR).

ASLR and data execution protection (DEP): ASLR is a security technique involved in protection from buffer overflow attacks. To prevent an attacker from locating attach code positions, ASLR randomly arranges the address space positions of the stack, heap, and libraries. ASLR makes it harder for attackers to exploit memory vulnerabilities. DEP marks specific memory locations in a process as non-executable, helping prevent attacks on memory vulnerabilities.

Secure Input

EMUI provides secure input function when users are entering passcodes. If the secure input function is enabled, the system will automatically switch to secure input when the user enters the passcode. Secure input and common input are managed separately. To safeguard user passcodes, the secure input method does not support memorizing or association functions. It cannot connect to the Internet or collect user passcodes. After the secure input is launched, screen recording cannot be performed in the background, and no third-party apps can capture screenshots. (Note: Self-developed input methods will be used in some bank APKs, and the secure input method does not take effect.)

Application Threat Detection

Security risks may exist in third-party applications, so downloading applications from third-party channels may introduce malicious threats.

The Android OS can check whether application sources are legitimate during application installation. By default, only applications in Huawei HiApp can be installed. It is recommended that the mobile phones be set as disabling installations of applications from unknown sources to avoid risks.

Huawei HiApp provides more than 10 top anti-threat engines and the manual approval mode to ensure the security of official application sources. Users are recommended to download applications from Huawei HiApp to ensure application security.

Phone Manager with built-in industry-leading antivirus engines is provided on the devices. It has powerful anti-threat engines at both the local end and the cloud that support local and online scanning, enabling users to find out and check whether viruses exist in downloaded installation packages and running applications. Once a virus is detected, the system will give a warning immediately, and notifies the user of deleting viruses.

Malicious Website Detection

EMUI can detect whether the websites are phishing websites or websites with malicious threats in scenarios like web browsing and sending text messages. When a user browses a malicious web page, EMUI checks the website so that the browser can intercept the access to the website, and remind the user of the security risks. And it can identify malicious websites and notify users of the security risks when they receive text messages.

Traffic Management

EMUI provides the traffic management function, including the management of mobile data and data generated over Wi-Fi. They are measured separately to implement fine-grained management. The traffic management function enables real-time monitoring of the traffic usage of each application and displays the results to the user. In addition, it can control application networking types to prevent mobile data consumption in the background.

6 Cyber Security

When devices connect to the network, the secure connection mechanism is needed. Otherwise, they may connect to malicious sites, resulting in data leakage. This section focuses on EMUI security mechanism of the network connection and transmission. EMUI uses standard network security protocols, such as VPN, SSL/TLS, and Wi-Fi, to ensure the security in device connection and data transmission.

VPN

With a VPN, a user can establish a secure private network using public network links. The VPN user can perform secure data transmission and has overall control rights on the network.

Devices support VPN settings. Users can set VPN parameters and therefore access sensitive information on the devices securely.

The supported VPN modes are PPTP, L2TP, L2TP/IPSec PSK, and L2TP/IPSec CRT.

SSL/TLS

Devices support SSLv3 and TLSv1.0, 1.1, and 1.2. They also support SSL/TLS through a third-party OpenSSL protocol stack.

WPA/WPA2

Wi-Fi connection can be authenticated using various methods, such as WEP, WPA/WPA2 PSK, 802.1x EAP, WPS, and WAPI.

In addition, devices also support the WLAN hotspot function, which is disabled by default.

WLAN hotspot supports WPA2 PSK authentication to ensure the security of the connection.

Secure Wi-Fi Detection

Wi-Fi in public areas provides convenience to people, but at the same time, it may be illegally used to steal users' privacy and perform phishing, which will bring security issues like privacy disclosure and economic losses to users. EMUI provides Wi-Fi threat detection engine on access points. It detects the Wi-Fi to be connected. Once security risks are detected, it will notify users so that they can take measures to ensure connection security.

7 Communication Security

This chapter describes security protection that EMUI provides for device communication. Users many often receive crank calls and fraud messages. To minimize the possibility that users are deceived, EMUI provides functions such as anti-rogue base stations, block and filter, and text message encryption to safeguard communication.

Defense Against Rogue Base Stations*

Text messages sent from rogue base stations disturb users and may contain malicious website hyperlinks. Once users click such hyperlinks, they may have economic losses or other adverse impacts. EMUI provides chip-level rogue base station prevention function based on HiSilicon chips (other chip platforms are not supported). It analyzes parameter characteristics of rogue GSM base stations and normal base stations and selects the most secure base station. When a modem decodes system messages, it can identify rogue base stations and will not select the cells where rogue base stations reside.

Additionally, EMUI cooperates with banks for official website verification. After receiving a text message carrying information like "xx bank", EMUI checks whether the website in the text message is the official one, preventing phishing from rogue base stations.

Block and Filter

In daily life, many users have ever received harassing text messages or calls about product sales, real estate investment, or bank load. EMUI is able to block such harassing text messages and calls. Users can enable or disable the blocking and recording functions, and clear or restore the blocking records. In addition, users can mark harassing calls with different types and blacklist them to prevent further harassment. Users can blacklist a phone number from contacts, call records, or text messages, or manually create a blacklist item.

EMUI provides the following block and filter rules. Users can set one or multiple of the rules as required:

- Intelligent interception: EMUI intercepts calls and messages based on the phone number blacklist updated in cloud.
- Blacklist-based interception: EMUI intercepts calls and messages based on the user-specified blacklist.
- Contacts-based interception: EMUI intercepts calls and messages from phone numbers not included in the contacts.
- Unknown number-based interception: EMUI intercepts calls and messages unknown phone numbers.

Text Message Encryption

Text messages are personal privacy data, which may be stolen during network transmission or storage on devices. To protect text message-based communication, EMUI provides the text message encryption function in the default SMS client. Users can enable this function in their SMS client. As text message encryption is based on key management servers, Huawei IDs are required to activate this function. In addition, this function takes effect only when both the sender and receiver of text messages use devices that support the EMUI text message encryption function. Text messages are encrypted with the receiver's phone number as the public key. A user can send encrypted text messages without prior notification. Only the correct receiver can decrypt the ciphertext message.

Device Interconnection Security

EMUI provides authentication services for devices that log in with the same Huawei ID and can perform security authentication for two connected devices under the same LAN to check whether the devices are trustworthy devices using the same Huawei ID. Devices not registered under this Huawei ID will not pass the authentication.

Devices that support one-key hotspot and Huawei Share can connect to devices that log in with the same Huawei ID through Bluetooth or Wi-Fi P2P to share Wi-Fi hotspots and files.

If a device has enabled one-key hotspot, the device will send signals through Bluetooth Low Energy to connect with devices that log in with the same Huawei ID. The EMUI device authentication services perform the authentication process based on the trusted public keys of both devices to verify whether the other device is using the same Huawei ID. If the authentication passes, the EMUI device authentication services will provide the key used for creating this hotspot session, so that the one-key hotspot can use the session key to encrypt private hotspot connection information.

Users can enable Huawei Share to share files with nearby devices that log in with the same Huawei ID. The device will use EMUI device authentication services to verify whether the devices are under the same Huawei ID and send files through with encryption.

8 Payment Security

This chapter describes security protection for Huawei Pay and other mobile payment apps. For third-party payment apps, EMUI can identify malicious apps, isolate the payment environment for protection, and support verification code encryption to ensure payment security.

Huawei Pay

Using Huawei Pay, users can conduct payment on supported Huawei devices in a convenient, secure, and confidential way. Huawei Pay has enhanced security in both hardware and software design.

Huawei Pay is also designed to protect users' personal information. It will not collect any transaction information that can be used to identify a user. Payment happens only among payers, payees, and card issuers.

Huawei Pay Components

Secure element: is a chip that has been certified and recognized in the industry. It complies with the requirement for digital payment in the finance industry.

NFC controller: processes near field communication protocols and supports communication between the app processor and secure element and between the secure element and POS machine.

Huawei Pay app: refers to "Wallet" on devices that support Huawei Pay. In this app, users can add and manage credit and debit cards and conduct payment through Huawei Pay. Users can also query their payment cards and other information about the card issuers.

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

How Does Huawei Pay Use the Secure Element?

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

How Does Huawei Pay Use the NFC Controller?

As the entry of the secure element, the NFC controller ensures that all contactless payments are conducted through POS terminals near the payment devices. The NFC controller only marks the payment requests from devices in the field as contactless transaction.

Once a cardholder uses the fingerprint or passcode for the payment, the controller sends the contactless response prepared by the secure element to the NFC field. In this manner, detailed payment authorization information for contactless transaction is saved only in the local NFC field and will not be disclosed to the app processor.

Bank Card Binding

When a user adds a bank card to Huawei Pay, Huawei securely sends the payment card information and other information about the user account and device to the card issuer. According to the preceding information, the card issuer determines whether to allow the user to add the card to Huawei Pay.

Huawei Pay uses the server invoking commands to send and receive packets exchanged with the card issuer or network. The card issuer or network uses these commands to verify, approve, and add payment cards to Huawei Pay. The sessions between clients and servers are encrypted using SSL.

Complete payment card numbers will not be stored on Huawei servers. Instead, unique device card numbers are created, encrypted, and stored in the secure element. Huawei is inaccessible to the encryption method. Each device card number is unique, different from a common bank card number. Card issuers can prevent the use of device card numbers on magnetic stripe cards, phones, or websites. Device card numbers will only be stored in secure elements and will never be stored on Huawei Pay servers or backed up to HiCloud.

Adding Bank Cards to Huawei Pay

The process of manually adding a payment card requires the name, card number, expire date, and CVV code. Users can enter such information in the Wallet app or use the camera function to input the information. After the camera captures payment card information, the Wallet will try to fill in the card number. After all information is entered, the process verifies the information except the CVV code. Such information is encrypted and sent to the Huawei Pay server.

If any clauses and conditions are returned by the card issuer for the payment card confirmation process, Huawei downloads the clauses and conditions and displays them on the user's device.

If the user accepts the clauses 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 payment 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 payment card (if GPS is enabled).

The following operations are performed in the binding process:

- The device downloads the credential file representing the bank card.
- The mobile phone binds the payment card to the secure element.
- Extra Verification

Card issuers determine whether to perform extra verification on bank cards. According to the functions supported by card issuers, users can select text message verification as the extra verification means.

Users can select the contact information archived by their card issuers to obtain text message notification and enter the received verification code in the Wallet app.

Payment Authorization

The secure element permits the payment only after receiving authorization from the mobile phone to determine that the user has passed authentication through fingerprint or device passcode. If possible, fingerprint is the default payment mode. Users can use the passcode to replace fingerprint at any time. If fingerprint does not match for one time, the system automatically prompts you to enter the passcode.

Using Huawei Pay for Contactless Payment

If a Huawei mobile phone is powered on and detects an NFC field, it displays related bank cards. The user can access the Huawei Pay app and select a bank card, or a specific fingerprint sensor to evoke the payment page.

If the user is not authenticated, no payment information will be sent. After the user is authenticated, the device card number and dynamic security code dedicated for transaction are used during payment.

Suspending, Removing, and Erasing Payment Cards

Card issuers or payment networks can suspend the payment function of Huawei Pay payment cards or remove the cards from devices even if the devices do not access cellular networks or WLANs.

Secure Keys*

Second-generation U key (such as USB key and audio key) is the main network transaction security solution for banks. As extra security hardware, the second-generation U key is easy to damage and lose, has a low use rate and poor user experience, and is not convenient to carry. While the main security strategy of apps with a mobile payment function is binding with mobile phones during transactions through bank payment channels. The transactions are confirmed through SMS messages and pose high security risks. Users are worried that their money may be stolen during payment.

Huawei Secure keys are combined with an independent internal secure element. The secure element is an authenticated chip widely accepted in the industry and 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 finance-level hardware protection for electronic payment.

When providing banks' mobile phone certificate services, Huawei's remote Trusted Service Manager remotely manages the secure element and establishes a Secure Copy Protocol (SCP) channel with the secure element to create a trusted, independent, and secure running space within the secure element. After an applet is installed in the space, independent public and private key pairs are generated within the secure element. The applet uses the TEE's Trusted User Interface (TUI) to set a bank certificate PIN. After the TUI is launched, screen interaction events are managed by the TEE to prevent Android background apps from capturing screenshots, recording the screen, and tracking on-screen touch points and ensure secure certificate PIN entering. Users verify their PINs on the TUI to complete transactions during certificate use.

During the entire lifecycle from public and private certificate key generation to certificate destruction, the private key is always in the secure element and therefore is secure.

Viewing Secure Keys Apps

Secure keys can check app package names and signatures. Only official apps will appear on the management screen to avoid malicious fake apps. Secure keys has a uniform viewing interface and management access to Secure keys apps, to query and manage Secure keys apps on users' phones.

Secure Keys Switch

The phone system has a switch to avoid background programs and apps from maliciously invoking the bank certificate. Turning on and off the switch is equal to inserting and removing a traditional USB key. When the switch is turned off, all certificate-related business cannot be conducted. Therefore, Secure keys can offer users the experience of being able to control hardware security.

Dedicated Secure Keys Section

To ensure that all Secure keys finance apps are from reliable sources, HiApp has a dedicated section for finance apps supporting Secure keys, and all these finance apps have official signatures. Users can find all kinds of finance apps supporting Secure keys business in this section.

Payment Protection Center

The payment protection center protects payment apps. It can isolate payment apps and detect app sources and threats in payment environments to secure financial, property, insurance payment apps.

App source detection: Apps in the payment protection center must be from the payment area in the Huawei HiApp or payment apps that have passed Huawei payment market's payment area authentication.

Smart reminder: When a user launches a qualified app, the phone will automatically remind the user to add the app to the payment protection center.

Access control and isolation: The payment protection center can access apps inside the space and control or isolate apps outside the space. As some payment apps (such as Alipay and WeChat) frequently interact with the external network, the center prevents access only from untrusted apps (malicious apps that have been detected) outside the area.

Financial property apps (such as China Merchants Bank) do not need to interact with the external network. The payment protection center deeply isolates such apps and prevents access from third-party apps outside the area.

Threat detection and system protection: The payment protection center can detect threats, Wi-Fi threats, text verification codes, ROOT and protect input methods.

Protecting SMS Verification Codes

Currently, SMS verification codes have become one of important authentication factors for mobile apps. Once an SMS verification code is intercepted, the user is faced with information leak or economic loss risks. To minimize such risks, EMUI provides protection for SMS verification codes to prevent malicious apps from intercepting users' text messages.

The SMS verification code intelligent identification engine is added to the EMUI system layer. After identifying an SMS verification code, the engine sends the text message only to the default SMS client set in the EMUI system. If the default SMS client is EMUI's built-in SMS client, the SMS client encrypts the text message with the verification code and filters access to the message, preventing third-party SMS clients or apps from accessing the message. Even if a third-party SMS client or app directly accesses the SMS database, text messages with verification codes are encrypted, and the client or app cannot decrypt them.

📖 NOTE

This function takes effect only when EMUI's built-in SMS client is set as the default SMS client.

9 Internet Cloud Service Security

Huawei has established a series of powerful cloud services to help users use devices more effectively. In design, these Internet services inherit the security objectives promoted by EMUI on the whole platform. Cloud services protect users' personal data stored on the Internet or transferred over the network, defend against threats and network attacks, and prevent malicious or unauthorized access to such information and services. Huawei cloud services use a unified security architecture which does not affect the overall usability of EMUI while ensuring user data security.

Huawei ID

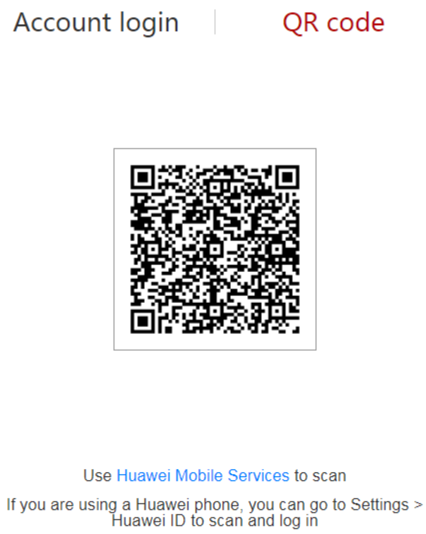
A Huawei ID can be used to access all Huawei services, such as HiCloud, HiApp, HiGame, Huawei Video, and Huawei Music. Ensuring Huawei ID security and preventing unauthorized access to user accounts are very important to users. To achieve this goal, Huawei requires that users use a strong password which is not commonly used and contains at least 8 characters in forms of lowercase and uppercase letters and digits. On this basis, users can add more characters and punctuation marks (the maximum password length is 32 characters) to make the password more secure.

In case of major changes to a Huawei ID, Huawei will send a text message, email, and notification to the concerned user, such as when the password is changed or used on a new device. If any error occurs, Huawei will prompt users to immediately change the passwords. Huawei has also adopted various policies and procedures to protect users' Huawei IDs, including limiting the numbers of login and password resetting attempts, continuously monitoring fraudulent activities for attack identification, and regularly reviewing existing policies for timely update according to new information that may affect user security.

Code Scanning Login

To facilitate user's access to Huawei Internet cloud services through browsers using their Huawei ID, the Huawei ID provides the code scanning login function. After logging in with their Huawei ID on a Huawei phone, users can use the code scanning function and scan the QR code on the login page to access corresponding Huawei Internet cloud services.

Figure 9-1 Code scanning login



Account login | QR code

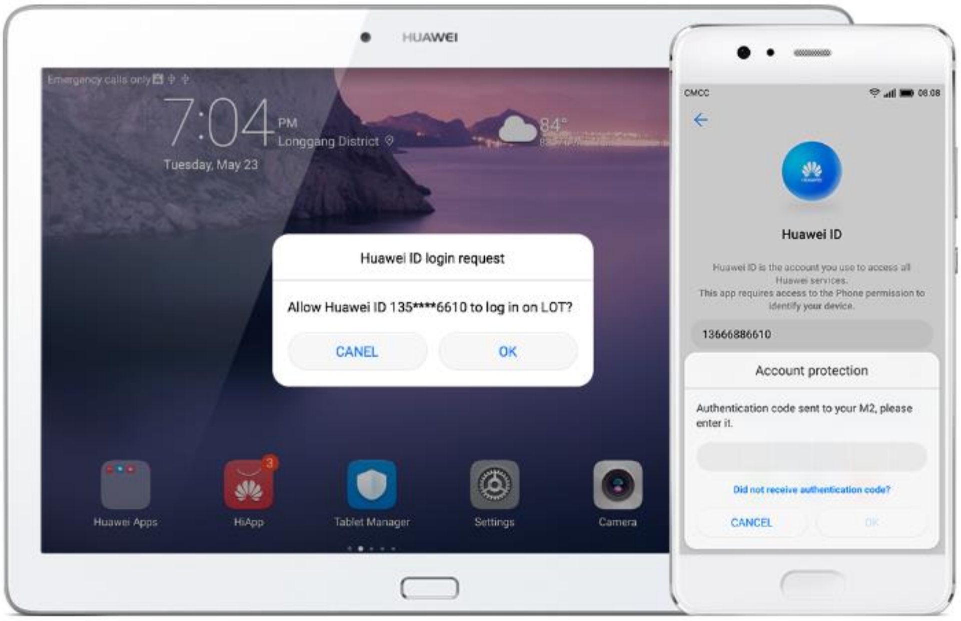
Use Huawei Mobile Services to scan

If you are using a Huawei phone, you can go to Settings > Huawei ID to scan and log in

Account Protection

Two-factor authentication is the optimal account protection solution and ensures more secure Huawei ID use.

Figure 9-2 Account protection



Huawei ID login request

Allow Huawei ID 135****6610 to log in on LOT?

CANEL OK

Huawei ID

Huawei ID is the account you use to access all Huawei services.
This app requires access to the Phone permission to identify your device.

13666886610

Account protection

Authentication code sent to your M2, please enter it.

Did not receive authentication code?

CANCEL OK

Account protection only allows users to log in with their Huawei ID using their trusted devices. If a user logs in with their Huawei ID using a new device, the user will need to enter their Huawei ID password and security verification code. The verification code is automatically displayed on their trusted devices or sent to their trusted phone number. If the new device passes the verification, it will become the user's trusted device. For example, a user has a HUAWEI M2, and the user wants to log in with their Huawei ID on their new HUAWEI P10. Their HUAWEI P10 will require them to enter their Huawei ID password and the verification code displayed on their HUAWEI M2. Therefore, Huawei IDs and using Huawei ID business (such as HiCloud, HiApp, Wallet, and HiGame) become more secure.

Huawei ID Message

The Huawei ID message function is a message sending and receiving service applicable to Huawei devices. This function supports texts and attachments, such as photos, contact information, and location information. The information is displayed on all registered devices of the user so that the user can continue the dialog on any of the devices. Huawei does not record users' information or attachments. In addition, the content is under end-to-end encryption protection.

HiCloud

HiCloud provides the function of storing user's contacts, messages, photo albums, call records, reminders, calendars, browser bookmarks, and other contents, and synchronizes information between the users' devices. The user can log in with their Huawei ID to set HiCloud and choose services as they want.

When the user logs out of their Huawei ID, related authentication information will be deleted, and after obtaining user confirmation, HiCloud will delete all related data to ensure that the user's personal data is not stored on the unused device. The user can log in with their Huawei ID on a new authenticated device to restore the HiCloud data.

Huawei ID–based Key

HiCloud files divide into different blocks. Each block is encrypted or decrypted using AES-128. HiCloud encryption and decryption require Huawei ID login. After a user successfully logs in with a Huawei ID, HiCloud derives an encryption factor for the Huawei ID and sends the factor and block metadata to the hardware encryption and decryption system. HiCloud files are encrypted and decrypted in this system and are then sent to the user's device through secure transmission channels. When the user data is stored on HiCloud, it is protected through the key bound to the Huawei ID. This means that no one else can read or write the data.

HiCloud Backup

HiCloud backup backs up data (including device settings, app data, and photos and videos on the device) to the cloud only when user devices can access the Internet through a WLAN. HiCloud will encrypt and protect backed-up data.

10 Device Management

This chapter describes the device management function of EMUI. For enterprise users, EMUI supports the native "Android for Work" framework of Android and provides capabilities such as enabling enterprise users of third-party MDM platforms to apply for enterprise certificates and granting the authorization signature. For scenarios where the user loses his/her mobile phone, EMUI provides Find My Phone, remote lock, remote data erasure, and other functions.

Find My Phone & Activation Lock

EMUI provides the function Find My Phone, which needs to be manually enabled by the user. After enabling the function, the user can locate (including active positioning and automatic location reporting at a lower battery level) a lost device, cause the phone to ring, lock, and erase device data by logging in to the cloud service website (cloud.huawei.com) or using the Fine My Phone function on a Huawei phone to ensure device data security. (Note: The function is available only in China.)

In addition, EMUI provides the function of activation lock. Enabling Find My Phone will enable the activation lock function on the mobile phone at the same time. If an unauthorized user attempts to forcibly erase data from the lost phone, after reboot of the phone, the user needs to log in to the Huawei ID to re-activate the phone. This function ensures that unauthorized users cannot activate or use the phone, protecting security of the phone.

MDM

The MDM function is completely inherited from the native "Android for Work" framework of Android. By creating work profiles, the enterprise IT system can easily control and manage Android devices. Android for Work supports mainstream third-party EMM vendors. It communicates with the EMM server and distributes device configuration and management policies through the device policy controller application on the device.

In addition, EMUI opens authorization APIs for device management to EMM vendors. In regions where Google Mobile Services are unavailable or scenarios where the vendors do not want to rely on Google Mobile Services, the vendors can install a third-party MDM client on the device and call the open APIs to control and manage the device. Authorization is required for API call to implement permission control and to ensure security.

MDM API

Currently, EMUI provides (including but not limited to) the following access restriction interfaces for EMM vendors and application developers who need to implement device configuration and access control. For details about the different interfaces, see the Huawei open platform website.

- Allow/Forbid the app to turn on the Bluetooth
- Allow/Forbid the app to make calls
- Allow/Forbid the app to send text messages
- Allow/Forbid the app to send multimedia messages
- Allow/Forbid the app to access contacts
- Allow/Forbid the app to modify contacts
- Allow/Forbid the app to delete contacts
- Allow/Forbid the app to read text messages and multimedia messages
- Allow/Forbid the app to read the call record
- Allow/Forbid the app to modify the call record
- Allow/Forbid the app to delete the call record
- Allow/Forbid the app to read the schedule information
- Allow/Forbid the app to read the personal location information
- Allow/Forbid the app to read information on the phone
- Allow/Forbid the app to turn on the camera
- Allow/Forbid the app to turn on the microphone
- Allow/Forbid the app to connect to the network
- Enable/Disable the WLAN
- Enable/Disable the WLAN hotspot
- Enable/Disable USB debugging mode and data transmission
- Enable/disable storage access (MicroSD card)
- Enable/disable NFC
- Enable/Disable data connection
- Enable/disable call making
- Enable/Disable SMS
- Enable/Disable status bar drop-down list
- Allow/Forbid ending current call
- Allow/Forbid power-off
- Allow/Forbid restart
- Allow/Forbid obtaining ROOT status
- Allow/Forbid all-time app running
- Allow/Forbid preventing app starting and running
- Allow/Forbid stopping app process
- Allow/Forbid silent installation of an app
- Allow/Forbid silent uninstallation of apps
- Allow/Forbid deleting app data
- Enable/Disable installation of apps in the specified app store and block installation of apps in other app stores and other installation methods such as Android Debug Bridge (ADB) and SD card installation
- Allow/Forbid installing apps

- Allow/Forbid uninstalling apps
- Allow/Forbid configuring Exchange parameters of Huawei mailbox

Device Management Certificate Authorization

For device management APIs required by enterprise mobile working customers, EMUI grants the customers corresponding use permissions by proving a signed enterprise certificate. The enterprises can apply for the device management API use permission from Huawei's open platform.

Huawei issues device management certificates through Huawei open platforms to app developers qualified by Huawei. After the developer integrates the certificate into the developed Android package (APK), the APK can normally call the authorized APIs on Huawei devices.

When a user installs an APK having the device management certificate, EMUI analyzes and verifies each item of the certificate. After confirming that the signature is correct (the certificate is issued by Huawei and authentic), the APK passes the authorization and is normally installed. If the certificate fails the verification, the APK installation fails in order to ensure security of Huawei devices.

11 Privacy Protection

This chapter describes user privacy protection. Huawei devices may contain user privacy data, such as contacts, short messages, and photos. To protect user privacy, EMUI ensures that preset applications fully meet privacy compliance requirements, and provides application permission management, notification management, location-based service (LBS), and other privacy management functions. in addition, to further protect users' privacy, EMUI provides extended functions such as file safe, Private Space, and positioning disabling.

Permission Management

The Android operating system provides a permission management mechanism, which is designed to allow or restrict apps' access to APIs and resources. By default, no permissions are granted to Android apps, and access to protected APIs or resources is restricted to ensure security of such APIs and resources. During installation, apps request permissions, and users determine whether to grant the permissions.

The user is only notified of requested permissions at the initial installation of the app, and can only complete the app installation after granting the permission. As a result, users fail to deal with unwanted permissions requested by the app.

To supplement Android's permission management, Huawei extends the existing app permission management by enabling users to allow/deny permissions to an installed app for fine-grained control. The permission management function can manage the following permissions:

Android-defined permissions:

- Phone call
- Network
- SMS
- Contacts
- Call record
- Camera
- Location data
- Recording
- Wi-Fi
- Bluetooth
- Calendar

EMUI-defined permissions:

- Sending multimedia messages
- Obtaining motion data
- Using call transfer (CT)
- Obtaining browser Internet access records
- Obtaining the installed app list
- Suspended window
- Creating desktop shortcut

LBS

To protect users' location information, EMUI enhances Android's native LBS. In addition to disabling the Global Positioning System (GPS) service, it will also disable Wi-Fi and mobile base station positioning by turning off the LBS. In this way, users' location positioning can be completely disabled, ensuring user privacy.

Notification Management

For troubles caused by frequent notifications from the app, EMUI provides a notification management function. Users can allow or forbid an app to send notifications. In scenarios where users allow the notification sending, EMUI provides fine-grained management for users to configure as required whether to allow the app to display notifications in the status bar, lock screen, or on the top of the screen as a banner, and to allow ringing or vibrating upon notification receiving, to avoid unnecessary troubles caused by frequent notifications.

App Lock

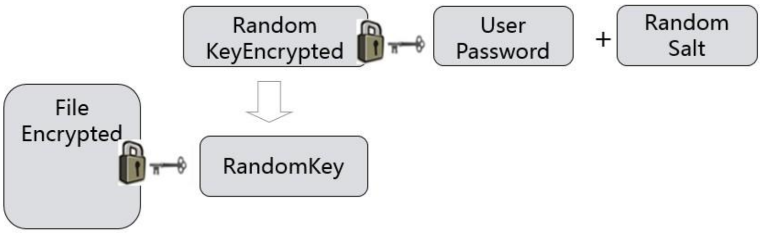
To protect the user's privacy when he/she lends the mobile phone to others, EMUI provides a function of app access passcode setting. The phone user must enter the correct passcode for authentication to use the app. Therefore, this function prevents unauthorized access to the locked apps and protects users' privacy.

File Safe

EMUI provides the file safe function, which allows users to add sensitive or important personal data to the file safe for protection. The file safe is a space encrypted using the user's password. All content in the safe is encrypted and protected using the user's key and be accessed only by the user.

Files in the file safe are encrypted using a randomly generated encryption key and AES256, and the encryption key is encrypted and protected using the safe password entered by the user. The user's password is not stored and cannot be restored.

Figure 11-1 File safe



Private Space*

To protect users' important privacy, EMUI provides Private Space. Designed based on the native multi-user mechanism of Android, Private Space is the encrypted and independent user space isolated from the master user space. Its app data is separately stored and isolated from app data of other users. Private Space provides isolation protection for data and apps stored in the internal memory. Users accessing Private Space cannot access the external SD card at the same time.

Also, Private Space provides an entrance hiding function. After the entrance to Private Space is hidden, other user space is completely unaware of Private Space unless switched to it on the user's lock screen homepage through the passcode or fingerprint. The quick user space switch function allows the user to use different fingerprints or passcodes to enter different user space: Use the device owner's fingerprint or passcode to enter Main Space; use the fingerprint or passcode to Private Space to enter Private Space. (Note: Main Space and Private Space must not have the same fingerprint or passcode.)

To facilitate users' private data protection, the privacy space supports file (including three types of files, such as videos, audio, and images) transfer with Main Space.

Privacy Policy

EMUI provides an explicit privacy policy statement in the system and explicitly notifies the user to check and confirm the statement in the startup wizard. In addition, the user can check the privacy policy statement in **Settings**. Due to different privacy policies in different countries, users in different countries should use specific privacy policy statements on EMUI released in the local countries.

Read Huawei Privacy Policy at:

http://consumer.huawei.com/minisite/worldwide/privacy-policy/cn/index.htm.

12 Conclusion

EMUI attaches great importance to users' device security and privacy and provides an end-to-end (from bottom-layer chips and systems to apps) security protection capability based on chip hardware. EMUI constructs a trusted basic architecture for the device based on the chip hardware, and constructs security experience considering both security and user experience based on higher security and good computing performance of the device hardware.

EMUI is developed based on the Android operating system. At the system layer, EMUI enhances system security through enhancing kernel security. Based on underlying trusted platform and system security enhancement, it provides a more secure system control capability for the upper layer. On the app layer, EMUI provides virus scanning, Block and Filter, data management, notification management, and other functions, and works together with the cloud to ensure security.

While providing security solutions, Huawei also attaches great importance to the establishment of the security process and security capabilities to implement security management through the product life cycle.

Huawei has set up a dedicated computer emergency response team (CERT) which is dedicated to improving product security. Any organization or individual that finds security vulnerabilities in Huawei products can contact Huawei at PSIRT@huawei.com. Huawei PSIRT will contact you in the shortest time while organizing internal vulnerability fixing, releasing vulnerability warning, and pushing patches for update. Huawei is sincerely willing to jointly construct Huawei device security with you.

13 Acronyms and Abbreviations

Table 13-1 Acronyms and abbreviations

Acronym/Abbreviation	Full Name
3DES	Triple Data Encryption Algorithm
ADB	Android Debug Bridge
AES	Advanced Encryption Standard
API	Application Programming Interface
ARM	Advanced RISC Machines
ASLR	Address Space Layout Randomization
CERT	Computer Emergency Response Team
HiCloud	HiCloud
DEP	Data Execution Prevention
ECB	Electronic Code Book
ECC	Elliptic Curves Cryptography
ECDSA	Elliptic Curve Digital Signature Algorithm
EAP	Extensible Authentication Protocol
EMM	Enterprise Mobility Management
EMUI	Emotion UI
GP	GlobalPlatform
HKIP	Huawei Kernel Integrity Protection
HMAC	Hashed message Authentication Code
HOTA	Huawei Over The Air
HUK	Hardware Unique Key
IPsec	IP Security

Acronym/Abbreviation	Full Name
L2TP	Layer Two Tunneling Protocol
LKM	Loadable Kernel Module
LSM	Linux Security Modules
MDM	Mobile Device Management
NFC	Near Field Communication
NIST	National Institute of Standards and Technology
OTA	Over The Air
PPTP	Point-to-Point Tunneling Protocol
PRNG	Pseudo Random Number Generator
PSK	Pre-Shared Key
ROM	Read-Only Memory
RSA	Rivest Shamir Adleman
RPMB	Replay Protected Memory Block
SD	Secure Digital Memory Card
SELinux	Secure Enhanced Linux
SHA	Secure Hash Algorithm
SSL	Security Socket Layer
TEE	Trusted Execution Environment
TLS	Transport Layer Security
TUI	Trusted User Interface
VPN	Virtual Private Network
WAPI	WLAN Authentication and Privacy Infrastructure
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access
WPS	Wi-Fi Protected Setup

Change History

Date	Description
2017-10-31	Added the following new contents for EMUI 8.0: • HKIP • File system encryption key protection • SD card encryption • Secure input function improvement • Device interconnection security • Secure keys • Payment protection center optimization • Code scanning login • Find My Phone function improvement • Privacy space function improvement
2017-05-31	Released the first version.