GB

NATIONAL STANDARD OF THE

PEOPLE'S REPUBLIC OF CHINA

ICS 35.030
CCS L 80

GB/T 41388-2022

Information security technology - Trusted execution environment - Basic security specification

信息安全技术 可信执行环境

基本安全规范

Issued on: April 15, 2022 **Implemented on: November 01, 2022**

Issued by: State Administration for Market Regulation;

Standardization Administration of the PRC.

Table of Contents

Information security technology - Trusted execution environment - Basic security specification

1 Scope

This document establishes the overall technology architecture of trusted execution environment system; describes the basic requirements of trusted execution environment, trusted virtualization system, trusted operating system, trusted application and service management, cross-platform application middleware, etc. and their test evaluation methods.

This document applies to guide the design, production, and testing of trusted execution environment system.

2 Normative references

The contents of the following documents, through normative references in this text, constitute indispensable provisions of this document. Among them, for dated references, only the edition corresponding to that date applies to this document. For undated references, the latest edition (including all amendments) applies to this document.

GB/T 20271-2006 Information security technology - Common security techniques requirement for information system

GB/T 25069-2010 Information security technology glossary

3 Terms and definitions

The terms and definitions defined in GB/T 25069-2010 and the following ones apply to this document.

3.1

Virtualization

A method of virtualizing one or more forms of resources into another or more forms of resources.

3.2

Trusted virtualization

a hardware-level isolation mechanism. It shall be ensured that the isolated resources of the trusted execution environment are not accessed by the rich execution environment. The specific resources that need to be isolated include but are not limited to: CPU, memory, clock source, cryptographic unit, debug unit, etc. See Appendix A for the trusted execution environment hardware reference architecture.

6.1.2 Trusted clock source

The trusted clock source is the hardware basis of the watchdog timer and the trusted execution environment scheduling timer, which are used in the trusted execution environment system. The trusted clock source shall run immediately after the device is powered on; and cannot be disabled, turned off, tampered, etc.; to avoid the destruction of the programming state in the trusted execution environment system due to external interference and other reasons.

6.1.3 Trusted random source

The trusted random source provides random sources for various encryption-decryption algorithms in the trusted execution environment. The random number generator involved in the random source shall be a true random number hardware generator that meets the relevant cryptographic requirements. The random number generator shall be set to be accessible only through the trusted execution environment.

6.1.4 Trusted debug unit

The trusted debug unit is responsible for the debug function of the entire device. The hardware foundation shall meet the following requirements:

a) The trusted debug unit hardware subsystem shall ensure that all intrusive debug mechanisms can be disabled;

b) The trusted debug unit hardware subsystem shall ensure that all debug mechanisms (including non-intrusive and intrusive) can be set so that only the trusted execution environment can use them;

c) The trusted debug hardware unit subsystem shall ensure that all debug mechanisms (including non-intrusive and intrusive) can be used by the rich execution environment. When the trusted debug hardware unit subsystem is set to be accessible to both the rich execution environment and the trusted execution environment, the debug subsystem is not allowed to access or modify any registers and memory in the trusted execution environment;

d) For the registers of the trusted debug hardware unit subsystem, it shall ensure continuous power supply during use; or ensure that the registers are completely saved before the system is powered off AND completely restored when the system resumes power supply.

6.1.5 Trusted peripheral

Trusted peripherals refer to a class of peripherals that have certain security or privacy requirements for drive control and data acquisition. Before using trusted peripherals, the system shall be switched to the trusted execution environment, to prevent any malicious code staying in the rich execution environment from monitoring and logging data input.

For peripherals whose acquisition and processing process cannot be completely controlled by the trusted execution environment, the following two methods should be used for processing:

a) Encrypted transmission mode, that is, establish an encrypted channel between trusted peripherals and trusted execution environment, to ensure the confidentiality, integrity and authenticity of data during transmission;

b) Monitoring mode, through the trusted execution environment, strictly detect and monitor the security status of the rich execution environment and the entire device; control the risks in time.

6.2 Trusted root

The trusted root provides support for the establishment and operation of the trusted execution environment, which can be hardware, code, and data. The trusted root shall have the following security requirements:

a) It shall have three basic security characteristics of confidentiality, integrity, and authenticity; can provide support for the security authentication, security measurement, and security storage of the trusted execution environment system;

b) It shall provide an access control mechanism, to ensure that unauthorized users cannot access and tamper with the data and code of the trusted root.

6.3 Secure boot requirements

Secure boot is to use a security mechanism to verify the integrity and authenticity of the software code at each stage of the trusted execution environment system startup process, to prevent unauthorized or maliciously tampered code from being executed. The secure boot process builds a chain of trust. The whole process starts with a trusted root. Other components or codes need to pass the integrity and authenticity verification before they can be executed. The secure boot process shall ensure the integrity and authenticity of the trusted execution environment system.

Secure boot shall meet the following requirements:

a) It shall ensure the robustness of the cryptographic algorithms themselves used to verify integrity and authenticity;

8 Trusted operating system

A trusted operating system shall have basic system functions in conventional operating systems such as process management, memory management, device management, and file management. Trusted operating systems are required to meet corresponding security technical requirements in terms of access control, identity authentication, data integrity, and trusted path, etc., including:

a) It shall be ensured that trusted applications and trusted services can only access the corresponding resources according to their assigned permissions; and cannot access beyond their authority;

b) It shall ensure the correctness and integrity of the system itself, trusted services, and application startup;

c) It shall ensure the authenticity and integrity of the system itself, trusted services, and application data and codes;

d) It shall have access control capabilities between trusted applications and between trusted applications and trusted services;

e) For the management of system permissions, it shall avoid granting the highest authority to trusted services and applications; to prevent the normal operation of the system kernel and other trusted applications and services from being affected when a single trusted application and service are abnormal.

9 Trusted application and service management

9.1 Basic description

Trusted application and service management is responsible for the installation, update, deletion, and security attribute configuration management of trusted applications and trusted services in a trusted execution environment. Trusted applications and services in the trusted execution environment can be managed locally or remotely through the TAM background. The security requirements they follow shall be consistent.

9.2 Technology architecture

9.2.1 Architecture description

The release process of trusted applications and services is shown in Figure 2. The device provider (or authorized service provider) is the owner of the trusted execution environment system; is responsible for the management of the device provider (or authorized service provider) root certificate AND the signing-issuance of the application release certificate. The trusted application provider is responsible for the

consumption state. The starting point of persistence time for trusted applications is different for each trusted application; but it shall remain persistent across restarts.

10.2 Trusted encryption-decryption service

The trusted execution environment system shall integrate trusted encryption-decryption services, to provide encryption-decryption functions for trusted applications and other trusted services. Trusted encryption-decryption services shall ensure that, only trusted applications or trusted services that have obtained corresponding authorization can access the keys.

10.3 Trusted storage service

The trusted execution environment system shall integrate trusted storage services, to provide trusted storage functions for trusted applications and other trusted services. Trusted storage services include but are not limited to the following functions:

a) Read-write operations on storage objects are required to ensure atomicity of operations, confidentiality of data, and integrity of data;

b) Trusted storage shall have an access control mechanism, to ensure that only authorized applications can access the corresponding storage space;

c) Trusted storage should provide defenses against data rollback attacks.

10.4 Trusted identity authentication service

The trusted execution environment system may integrate trusted identity authentication services, to provide identity authentication functions for trusted applications or other trusted services in the trusted execution environment. By identifying the user's personal identity digital feature information, the trusted identity authentication service identifies the legitimacy of the user's identity and whether it has the right to operate related functions. The trusted identity authentication service can use (but not limited to) the following identity authentication methods to complete the judgment of the user's legitimacy: Password, fingerprint, face. The trusted identity authentication service should be completed based on the collaborative operation of other trusted services, such as trusted storage services, trusted human-computer interaction, and trusted encryption-decryption services.

10.5 Trusted device authentication service

The trusted execution environment system may integrate the trusted device authentication service, to prove the authenticity of the device. The trusted device authentication service can provide (but not limited to) the following types of functions:

a) Prove the authenticity of the device identification and the device source;

If the above expected results are satisfied, it is determined as conformity; other cases are determined as nonconformity.

13.1.1.4 Trusted debug unit

The test evaluation method for trusted debug unit is as follows.

a) Test method:

1) Review the documents submitted by the manufacturer; check the trusted debug unit design of trusted execution environment;

2) Attempt to use trusted debug unit via intrusive debug mechanism;

3) Set all debug mechanisms (including non-intrusive and intrusive) of the trusted debug unit to be used only by the trusted execution environment; attempt to use the debug mechanism outside the trusted execution environment;

4) Set all debug mechanisms (including non-intrusive and intrusive) of the trusted debug unit so that they can be used by the rich execution environment and the trusted execution environment. Attempt to access or modify registers and memory in the trusted execution environment through debug mechanisms;

5) After setting the registers of the hardware subsystem of the trusted debug unit, try to power off the system. After powering on, read the corresponding registers again; compare them with the register values before powering off.

b) Expected results:

1) The trusted debug unit hardware subsystem ensures that all intrusive debug mechanisms can be disabled;

2) The trusted debug unit hardware subsystem ensures that all debug mechanisms (both non-intrusive and intrusive) can be set to be used only by the trusted execution environment;

3) The trusted debug unit hardware subsystem ensures that all debug mechanisms (both non-intrusive and intrusive) can be used by the rich execution environment. When the trusted debug hardware subsystem is set to be accessible to both the rich execution environment and the trusted execution environment, the debug subsystem is not allowed to access or modify any registers and memory in the trusted execution environment;

4) During the use of the registers of the hardware subsystem of the trusted debug unit, the continuous power supply is guaranteed. Or it can be ensured that the registers are completely saved before the system is powered off AND completely restored when the system resumes power supply.

c) Result determination:

If the above expected results are satisfied, it is determined as conformity; other cases are determined as nonconformity.

13.1.1.5 Trusted peripheral

The test evaluation method for trusted peripheral is as follows.

a) Test method:

1) Review the documents submitted by the manufacturer; check the trusted peripheral design of trusted execution environment;

2) When using trusted peripherals in the trusted execution environment, try to monitor and record the interaction data of trusted peripherals through the rich execution environment;

3) For peripherals whose acquisition and processing process cannot be completely controlled by the trusted execution environment, check the data transmission protection mechanism between the trusted peripheral and the trusted execution environment. Verify whether the trusted execution environment has a monitoring and response mechanism for the security status of the rich execution environment and the entire device.

b) Expected results:

1) Before using the trusted peripherals, the system is switched to the trusted execution environment, to prevent any malicious code staying in the rich execution environment from monitoring and logging data input.

2) For peripherals whose acquisition and processing process cannot be completely controlled by the trusted execution environment, encrypt and transmit data between the trusted peripheral and the trusted execution environment. And the trusted execution environment has a monitoring and response mechanism for the security status of rich execution environment and the entire device.

c) Result determination:

If the above expected results are satisfied, it is determined as conformity; other cases are determined as nonconformity.

13.1.2 Trusted root

The test evaluation method for trusted root is as follows.

a) Test method:

6) Check the level permission settings of the virtual machine inside the trusted virtualization system, to verify whether the highest-level permission is granted to the virtual machine.

b) Expected results:

1) The trusted virtualization system has the ability to dynamically manage virtual machines such as creation and deletion; as well as the ability to manage hardware resources such as CPU, memory, interrupts, and peripherals of virtual machines;

2) The virtual machines in the trusted execution environment have the ability to communicate with each other and exchange data. The communication between the virtual machines has the access control ability;

3) The virtual machine in the trusted execution environment only accesses the corresponding resources according to its assigned permissions; cannot access beyond its authority;

4) The trusted virtualization system ensures the correctness and integrity of the virtual machine loading and running process; as well as the authenticity and integrity of the virtual machine data and code;

5) The trusted virtualization system does not give internal virtual machines the highest level of authority. When a single virtual machine crashes or has security risks, it will not affect the normal work of the trusted execution environment system itself and other internal virtual machines.

c) Result determination:

If the above expected results are satisfied, it is determined as conformity; other cases are determined as nonconformity.

13.3 Trusted operating system

The test evaluation method for trusted operating system is as follows.

a) Test method:

1) Review the documents submitted by the manufacturer; check the design of trusted operating system;

2) Check the access control policy of trusted operating system; try to use trusted application and trusted service to respectively access the resources allowed and not allowed by the policy; verify whether the policy is valid;

3) Check the trusted operating system itself, trusted service and application startup process; try to tamper with the startup code and bypass the integrity

verification process;

4) Check the authenticity and integrity protection mechanism of trusted operating system itself, trusted service and application data and codes; try to tamper with the corresponding data and codes;

5) Check access control policies between trusted applications, between trusted applications and trusted services. Attempt to use trusted application and trusted service to respectively access trusted applications and trusted services that are permitted and not permitted by the policy, to verify whether the policy is valid;

6) Check the permission settings of trusted services and trusted applications inside the trusted operating system; verify whether trusted services and trusted applications are given the highest permissions.

b) Expected results:

1) The trusted operating system has basic system functions such as process management, memory management, device management, and file management in conventional operating systems;

2) The trusted operating system ensures that trusted applications and trusted services can only access corresponding resources according to their assigned permissions; and cannot access beyond their authority;

3) The trusted operating system ensures the correctness and integrity of the system itself, trusted service and application startup;

4) The trusted operating system ensures the authenticity and integrity of the system itself, trusted service and application data and codes;

5) Have access control capability between trusted applications and between trusted applications and trusted services;

6) The system authority management of trusted operating system will not grant the highest authority to the trusted services and applications. The crash or security issues of a single trusted application and service will not affect the normal operation of the system kernel and other trusted applications and services.

c) Result determination:

If the above expected results are satisfied, it is determined as conformity; other cases are determined as nonconformity.

This is an excerpt of the PDF (Some pages are marked off intentionally)

------ The End ------