GB

NATIONAL STANDARD OF THE

PEOPLE'S REPUBLIC OF CHINA

ICS 35.040
L 80

GB/T 20272-2019

Replacing GB/T 20272-2006

Information security technology - Security technical requirements for operating system

信息安全技术 操作系统安全技术要求

Issued on: August 30, 2019 **Implemented on: March 01, 2020**

Issued by: State Administration for Market Regulation;

Standardization Administration of the PRC.

Table of Contents

Information security technology - Security technical requirements for operating system

1 Scope

This Standard specifies the security technical requirements for operating system of five security classes.

This Standard applies to the research and development, testing, maintenance, and evaluation of security of operating system.

2 Normative references

The following documents are indispensable for the application of this document. For the dated references, only the editions with the dates indicated are applicable to this document. For the undated references, the latest edition (including all the amendments) are applicable to this document.

GB 17859-1999 Classified criteria for security protection of computer information system

GB/T 18336.3-2015 Information technology - Security techniques - Evaluation criteria for IT security - Part 3: Security assurance components

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

GB/T 29240-2012 Information security technology - General security technique requirements and testing and evaluation method for terminal computer

3 Terms and definitions

The terms and definitions defined in GB 17859-1999, GB/T 18336.3-2015, GB/T 20271-2006, and GB/T 29240-2012 and the following ones apply to this document.

3.1 Security of operating system

The confidentiality, integrity, and availability of operating system itself and of the information it stores, transmits, and processes.

a) User identification function:

1) Before users enter the operating system, they shall be identified first.

2) The user identification of operating system should use the username and UID.

b) User authentication function:

1) The password is used for authentication and is authenticated each time the user logs into the system and when the system is reconnected;

2) The password shall be invisible; during storage and transmission, shall be securely protected, to ensure that it is not accessed, modified, and deleted without authorization;

3) By predefining the value of unsuccessful authentication attempt (including the threshold of the number of attempts and time) and clearly specifying the measures taken when the value is reached, achieve the processing of authentication failure.

c) For users registered to the operating system, the user process shall be associated with its owner user, so that the behavior of user process can be traced back to the owner user of the process.

6.1.1.2 Discretionary access control

The discretionary access control functions of SSOOS are as follows:

a) The owner of the object shall have the right to modify its access rights for all objects owned by it;

b) The owner of the object shall be able to set the access control attributes of other users for the object it owns. The access control attributes include at least: read, write, execute, etc.;

c) The subject's access to the object shall follow the object's discretionary access control right attribute;

d) The granularity of the access control object is controlled in files and directories.

6.1.1.3 Data integrity

For user data transmitted internally the operating system (such as inter-process communication), there shall be a function of ensuring the integrity of user data.

e) Identify all possible states of SSOOS operation (including operational failures or operational errors) and their causal relationship and connection with maintaining secure operation;

f) Describe the security policy which shall be enforced to ensure secure operation of SSOOS.

6.1.3.2.2 Preparation procedure

The developer shall provide the operating system and its preparation procedure. The description of preparation procedure shall meet the following requirements:

a) Describe all the steps necessary to securely receive an operating system consistent with the developer delivery procedure;

b) Describe all the steps necessary to securely install the operating system and its operation environment.

6.1.3.3 Life cycle support

6.1.3.3.1 Configuration management capability

The developer's configuration management capabilities shall meet the following requirements:

a) Provide a unique identification for different versions of operating system;

b) Provide a configuration management document describing a method for uniquely identifying a configuration item;

c) The configuration management system uniquely identifies all configuration items.

6.1.3.3.2 Configuration management scope

The developer shall provide a list of SSOOS configuration items and a brief description of the developer of the configuration item. The list of configuration items shall contain the following:

a) SSOOS, evaluation evidence for security assurance requirements, and components of SSOOS;

b) Unique identification of configuration item;

c) For each configuration item related to security function, the list of configuration items briefly describes the developer of the configuration item.

algorithms which comply with national cryptography-related regulations are used.

6.1.3.5 Vulnerability evaluation

Based on the identified potential vulnerabilities, the operating system shall resist attacks by attackers with a basic attack potential.

Note: The resisting to attacks by attackers with basic attack potential needs to be comprehensively considered based on the following 5 specific factors: attack time, attacker ability, level of understanding of the operating system, time of access to operating system or number of attack samples, attack equipment used. See GB/T 30270-2013 Appendix A, A.8.

6.2 Class 2: system audit protection class

6.2.1 Security function requirements

6.2.1.1 Identity authentication

The identity authentication functions of SSOOS are as follows:

a) User identification function:

1) Before users enter the operating system, they shall be identified first;

2) The user identification of operating system shall use the username and UID. **Throughout the life cycle of the operating system, achieve the unique identification of the user, as well as the consistency between the username or alias, UID, and the like.**

b) User authentication function:

1) **Adopt mechanisms such as management-enhanced password authentication/token-based dynamic password authentication.** Authenticate each time the user logs in to the system and the system is reconnected;

2) **The authentication information shall be invisible;** during storage and transmission, shall be securely protected, to ensure that it is not accessed, modified, and deleted without authorization;

3) By predefining the value of unsuccessful authentication attempt (including the threshold of the number of attempts and time) and clearly specifying the measures taken when the value is reached, achieve the processing of authentication failure.

configuration parameters. Before initializing and protecting security-related data structures, it shall define the security policy attributes of user and administrator.

b) It shall distinguish normal operation mode and system maintenance mode.

c) **Before a normal user accesses the system, the system shall be installed and configured in a secure manner.**

d) **For regular system maintenance such as backups which does not affect SSOOS, it may be performed in normal operation mode.**

e) **After the operating system is installed, before normal user accesses, the system shall be configured with initial user and administrator responsibilities, root directories, audit parameters, system audit trail settings, and appropriate access controls for files and directories.**

f) **Only system administrators are allowed to modify or replace the executable program provided by the system.**

g) After the fault or interruption of SSOOS, it shall be recovered with minimal damage. According to the description of 5.1.2.2 failure protection in GB/T 20271-2006, it shall handle the SSF fault.

h) **It shall control and audit the use of system console.**

i) Developers of operating systems shall timely release patches for discovered vulnerabilities. The administrator of operating system shall promptly obtain, uniformly manage, and timely apply patches to repair the vulnerabilities of the operating system.

6.2.2.2 Resource utilization

6.2.2.2.1 Fault tolerance

The fault tolerance functions of SSOOS are as follows:

a) Certain measures shall be taken to ensure that the SSF can maintain normal operation when certain deterministic fault conditions occur in the system, **for example, the service level of system detection and reporting system has been reduced to a predetermined minimum;**

b) **When the service level of system resources is reduced to a predetermined minimum, it shall be able to detect and issue reports;**

c) **It shall provide the ability to run the system in maintenance mode, in which all security functions are disabled. The system only allows the**

1) **The date, time, source of this login and last successful login to the system;**

2) **The identity authentication failure since the last successful access to the system;**

3) **The number of days the password expires;**

4) **The number of successful or unsuccessful events can be expressed by an integer count, a timestamp list, and the like.**

6.2.2.4 Trusted measurement

The trusted measurement functions of SSOOS are as follows:

a) **When the operating system is started, it shall measure the integrity of the operating system kernel;**

b) **When the executable program is started, it shall measure the integrity;**

c) **The reference values of integrity measurement shall be stored securely, to prevent them from being tampered with.**

6.2.2.5 Security policy configuration

Security policy configuration functions shall be provided for identity authentication, **security audit**, network security protection, resource utilization, and user login access control.

6.2.3 Security assurance requirements

6.2.3.1 Development

6.2.3.1.1 Security architecture

The developer shall provide the security architecture description document of SSOOS. The security architecture description document shall meet the following requirements:

a) Consistent with the description of security function requirements and self-security protection requirements in the SSOOS design document;

b) Describe the security domain of SSOOS;

c) Describe why the SSOOS initialization process is secure;

d) Confirm that SSOOS can prevent damage;

6.2.3.4.1 Coverage

The developer shall provide a testing coverage document. The testing coverage document shall meet the following requirements:

a) Confirm the correspondence between the testing in the testing document and the SSOOS interface in the functional specification description;

b) **Confirm that all SSOOS interfaces in the functional specification description have been tested.**

6.2.3.4.2 Depth

Developers shall provide the analysis document of testing depth. The analysis document of testing depth shall meet the following requirements:

a) **Confirm the correspondence between the testing and security functions in the testing document and the self-security protection;**

b) **Confirm that all security functions and self-security protection functions in the SSOOS design have been tested.**

6.2.3.4.3 Functional testing

Developers shall test SSF and self-security protection functions. The testing document shall include the following:

a) Testing plan: Identify the tests to be performed and describe the scenario for performing each testing. These scenarios include any sequential dependencies on other testing results;

b) Expected testing results: Indicate the expected output after the testing is completed;

c) Actual testing results: Consistent with the expected testing results;

d) Confirm that known vulnerabilities have been corrected, eliminated, or invalidated, and retesting is performed after the vulnerabilities have been removed, to verify that they have been eliminated and that no new vulnerabilities have been introduced.

6.2.3.4.4 Independent testing

Developers shall provide a set of equivalent resources as they used in self-tests, for testing of SSOOS.

6.2.3.4.5 Cryptographic testing

accessed, modified, and deleted without authorization;

3) By predefining the value of unsuccessful authentication attempt (including the threshold of the number of attempts and time) and clearly specifying the measures taken when the value is reached, achieve the processing of authentication failure.

c) For users registered to the operating system, the user process shall be associated with its owner user, so that the behavior of user process can be traced back to the owner user of the process.

6.3.1.2 Discretionary access control

The discretionary access control functions of SSOOS are as follows:

a) The owner of the object shall have the right to modify its access rights for all objects owned by it.

b) The owner of the object shall be able to set the access control attributes of other users for the object it owns. The access control attributes include at least: read, write, execute, etc.

c) The subject's access to the object shall follow the object's discretionary access control right attribute.

d) **For discretionary access control with finer granularity, the granularity of the access control subject is controlled to a single user.** The granularity of the access control object is controlled in files and directories.

e) When the subject generates an object, the object shall have a default value of discretionary access control right attribute set by the subject.

f) Discretionary access control shall be able to combine identity authentication and audit to make users bear clear responsibility for their own behavior by confirming the authenticity of the user's identity and recording the user's various accesses.

g) **The object owner shall be able to set the object it owns: The owner is the only subject who has the right to modify its access rights.**

h) **The object owner is not allowed to assign control right of the object to other subjects.**

6.3.1.3 Label and mandatory access control

The label and mandatory access control functions of SSOOS are as

The object reuse functions of SSOOS are as follows:

a) **Ensure that unauthorized users cannot find information which, after use, is returned to the system's storage media (including at least: disk and memory, etc.);**

b) **Ensure that unauthorized users cannot find previous information which the system has assigned to its storage media (including at least: disk and memory, etc.).**

6.3.1.6.2 Data confidentiality

The data encryption functions of SSOOS are as follows:

a) It shall provide file encryption function. Users may encrypt-protect specified files and directories;

b) SUPPORT for protecting keys in hardware;

c) **It shall provide file system encryption function, to transparently encrypt and decrypt files and directories stored in the encrypted file system.**

6.3.1.7 Network security protection

The network security protection functions of SSOOS are as follows:

a) SUPPORT two-way network access control based on IP address, port, physical interface, **and applications**. DISCARD packets which do not meet the pre-defined policy;

b) The network transmission data shall be able to be encrypted and integrity-protected;

c) **SUPPORT two-way network trusted access authentication based on system identity and system running state;**

d) **Network access shall be controlled so that only authorized processes can access the network.**

6.3.2 Self-security requirements

6.3.2.1 Operation security protection

The operation security protection functions of SSF are as follows:

a) An installation mechanism shall be provided to set and upgrade configuration parameters. Before initializing and protecting security-

b) When the service level of system resources is reduced to a predetermined minimum, it shall be able to detect and issue reports.

c) It shall provide the ability to run the system in maintenance mode, in which all security functions are disabled. The system only allows the system administrator to enter maintenance mode.

d) **The system shall provide the process of software and data backup and recovery and add a restart synchronization point to the system, to facilitate system recovery.**

e) **The system shall provide mechanisms and processes which can be used to periodically confirm the correct operation of the system. These mechanisms or processes involve monitoring of system resources, proper operation of hardware and firmware units, detection of error conditions which may propagate throughout the system, and detection of communication errors which exceed user-defined thresholds, etc.**

6.3.2.2.2 Service priority

The service priority functions of SSOOS are as follows:

a) The service priority policy shall be adopted to set the subject to use the priority of a subset of resources within the SSF control scope to manage and allocate operating system resources;

b) It shall be ensured that access to all operating system resources is based on the priority set by the subject.

6.3.2.2.3 Resource allocation

The resource allocation functions of SSOOS are as follows:

a) In accordance with the requirements of 5.1.4.2 a) maximum quota resource allocation in GB/T 20271-2006, it shall carry out the management and allocation of operating system resources. The quota mechanism ensures that users and subjects will not monopolize certain controlled resources.

b) It shall be ensured that, when an authorized subject makes a request, resources can be accessed and utilized.

c) It shall, based on each user or each user group, provide a mechanism, to control its consumption of disks and the use of resources such as CPU.

d) **It shall provide the user with the permission to view the modification**

described in the design can be mapped to the SSOOS interface which invokes it;

g) According to the module, describe security functions and self-security protection.

6.3.3.2 Guiding document

6.3.3.2.1 Operation user guide

Developers shall provide clear and reasonable operation user guide. The operation user guide is consistent with all other documents provided for the evaluation. The description of each user role shall meet the following requirements:

a) Describe the functions and privileges which users can access in a secure processing environment and include warning information which may cause harm;

b) Describe how to use the security functions and self-security protection provided by SSOOS in a secure manner;

c) Describe the security function and self-security protection and interface, especially all security parameters controlled by the user; specify the security value when appropriate;

d) Define each security-related event related to the security function and self-security protection, including changing the security features of the entity controlled by SSOOS;

e) Identify all possible states of SSOOS operation (including operational failures or operational errors) and their causal relationship and connection with maintaining secure operation;

f) Describe the security policy which shall be enforced to ensure secure operation of SSOOS.

6.3.3.2.2 Preparation procedure

The developer shall provide the operating system and its preparation procedure. The description of preparation procedure shall meet the following requirements:

a) Describe all the steps necessary to securely receive an operating system consistent with the developer delivery procedure;

b) Describe all the steps necessary to securely install the operating system and its operation environment.

6.3.3.3.3 Delivery procedure

Developers shall use a certain delivery procedure to deliver the operating system and document the delivery process. When delivering the specified version of operating system to the user, the delivery document shall describe all the procedures necessary to maintain security.

6.3.3.3.4 Development security

Developers shall provide development security document. The development security document shall describe all physical, procedural, personnel, and other security measures necessary to protect the confidentiality and integrity of SSOOS design and implementation in the SSOOS development environment.

6.3.3.3.5 Life cycle definition

Developers shall establish a life cycle model to perform necessary control of the development and maintenance of SSOOS and provide a life cycle definition document describing the model used to develop and maintain SSOOS.

6.3.3.3.6 Tools and technology

Developers shall clearly define the tools for developing SSOOS and provide documentation for the development tools. The development tool documentation shall unambiguously define the meaning of all statements and all agreements and commands used by implementations; shall unambiguously define the meaning of all implementation dependencies.

6.3.3.4 Testing

6.3.3.4.1 Coverage

The developer shall provide a testing coverage document. The testing coverage document shall meet the following requirements:

a) Confirm the correspondence between the testing in the testing document and the SSOOS interface in the functional specification description;

b) Confirm that all SSOOS interfaces in the functional specification description have been tested.

6.3.3.4.2 Depth

Developers shall provide the analysis document of testing depth. The analysis document of testing depth shall meet the following requirements:

a) Confirm the correspondence between the testing and security functions in

c) The security attribute labels of the subject and object constitute a confidentiality and integrity security policy model. **These security policy models have corresponding semi-formal proofs.** Mandatory access control shall be based on the label and security policy model, to achieve access control of the subject's operations such as reading, writing, and executing objects.

d) The mandatory access control shall be closely integrated with the security functions such as user identity authentication and labeling, so that the security control of the system to the user includes the whole process from system startup to exiting the system, and the control scope of the mandatory access control to the object involves the storage, processing, and transmission process **and information input and output operations** inside the operating system.

e) The system's general management, security-related management, and audit management are assumed by the system administrator, system security officer, and system auditor, respectively. According to the division of functions and **the principle of minimum authorization**, each of them is granted the **least** permissions they need to complete the tasks they undertake. They shall form a mutual restraint relationship. The system security officer uniformly manages events and information related to security mechanisms such as mandatory access control in the operating system.

f) For the network operating system running on multiple computers in the network environment, when unified management is required, it shall consider the consistency of the security attribute settings of the subject and the object in each computer operating system; shall achieve the user data confidentiality and integrity protection between across-network operating systems.

6.4.1.4 Security audit

The security audit functions of SSOOS are as follows:

a) It shall be able to generate an audit log for the following events:

1) The use of security functions such as identity authentication, discretionary access control, and label and mandatory access control;

2) The operation of creating and deleting objects;

3) Network session;

4) All administrator operations.

2) Date and/or time;

3) User identity;

4) Object name;

5) Success or failure.

f) It shall provide the protection function of audit log:

1) Ensure that the audit mechanism is enabled by default and protect the opening and closing of audit log;

2) Protect audit logs from unauthorized access;

3) Ensure that the audit log is not tampered with and deleted; RECORD the behavior of attempting to tamper with and delete the audit log.

g) It shall provide the audit log lookup function in a way which is easy for the user to understand.

h) The audit log shall be stored in a power-down, non-lost storage medium. The system administrator shall be able to define a threshold which exceeds the audit trail storage limit. When the threshold is exceeded, the administrator will be alerted. When the audit storage space is exhausted, the oldest audit records stored are overwritten.

6.4.1.5 Data integrity

The data integrity protection functions of SSOOS are as follows:

a) For **data transmission** within the operating system (such as inter-process communication), there shall be a function of ensuring the data integrity.

b) When accessing data, it shall check whether the user data stored on the storage media is complete **and recover when it is found to be incomplete.**

c) It shall provide a utility to verify the integrity of the file system and disk. This utility shall be able to be executed automatically by the operating system.

6.4.1.6 Data confidentiality

6.4.1.6.1 Object reuse

The object reuse functions of SSOOS are as follows:

resource allocation in GB/T 20271-2006, it shall carry out the management and allocation of operating system resources. The quota mechanism ensures that users and subjects will not monopolize certain controlled resources.

b) It shall be ensured that, when an authorized subject makes a request, resources can be accessed and utilized.

c) It shall, based on each user or each user group, provide a mechanism, to control its consumption of disks and the use of resources such as CPU.

d) It shall provide the user with the permission to view the modification history of accessible system resources.

6.4.2.3 User login access control

The user login access control functions of SSOOS are as follows:

a) It shall, according to the requirements of 5.1.5 a) session establishment mechanism in GB/T 20271-2006, based on the access address or port, allow or deny the user's login. The authentication mechanism is not allowed to be bypassed.

b) It shall, according to the requirements of 5.1.5 c) multiple concurrent session limit in GB/T 20271-2006, limit the maximum number of concurrent sessions of the system and use the default value as the limit number of sessions.

c) After successfully logging into the system, the operating system shall record and display the following data to the user:

1) The date, time, source of this login and last successful login to the system;

2) The identity authentication failure since the last successful access to the system;

3) The number of days the password expires;

4) The number of successful or unsuccessful events can be expressed by an integer count, a timestamp list, and the like.

d) After the specified unused time limit, the system shall disconnect the session or re-authenticate the user. The system provides default values for the time limit.

e) The system shall provide a mechanism to lock the user's keyboard. The

security protection requirements in the SSOOS design document;

b) Describe the security domain of SSOOS;

c) Describe why the SSOOS initialization process is secure;

d) Confirm that SSOOS can prevent damage;

e) Confirm that SSOOS can prevent bypass.

6.4.3.1.2 Functional specification description

The developer shall provide a description of functional specification. The functional specification description shall meet the following requirements:

a) Fully describe SSF and self-security protection;

b) Describe the purpose and use method of all SSOOS interfaces;

c) Identify and describe all parameters related to each SSOOS interface;

d) Describe all behaviors associated with the SSOOS interface during implementation;

e) Confirm the traceability of security function requirements and self-security protection requirements to the SSOOS interface;

f) Describe all direct error messages which may be caused by calls to each SSOOS interface;

g) **USE a semi-formal mode to describe the SSOOS interface.**

6.4.3.1.3 Implementation representation

Developers shall provide description of implementation representation and provide a full implementation representation of SSOOS in their chosen locations. Implementation representation and description shall meet the following requirements:

a) It shall define SSF and self-security protection in detail, so that SSOOS can be generated without further design;

b) The implementation representation is provided in the form used by the developer;

c) In the description of implementation representation, it shall provide the mapping between the SSOOS design description and the implementation representation and demonstrate the consistency.

6.4.3.3.5 Life cycle definition

Developers shall establish a life cycle model to perform necessary control of the development and maintenance of SSOOS and provide a life cycle definition document describing the model used to develop and maintain SSOOS.

6.4.3.3.6 Tools and technology

Developers shall describe the implementation standards used. Developers shall clearly define the tools for developing SSOOS and provide documentation for the development tools. The development tool documentation shall unambiguously define the meaning of all statements and all agreements and commands used by implementations; shall unambiguously define the meaning of all implementation dependencies.

6.4.3.4 Testing

6.4.3.4.1 Coverage

The developer shall provide a testing coverage document. The testing coverage document shall meet the following requirements:

a) Confirm the correspondence between the testing in the testing document and the SSOOS interface in the functional specification description;

b) Confirm that all SSOOS interfaces in the functional specification description have been tested.

6.4.3.4.2 Depth

Developers shall provide the analysis document of testing depth. The analysis document of testing depth shall meet the following requirements:

a) Confirm the correspondence between the testing in the testing document and the security functions and self-security protection module in SSOOS design;

b) Confirm that all security function modules and self-security protection function modules in the SSOOS design have been tested.

6.4.3.4.3 Functional testing

Developers shall test SSF and self-security protection functions. The testing document shall include the following:

a) Testing plan: Identify the tests to be performed and describe the scenario for performing each testing. These scenarios include any sequential

2) **The process used to determine the existence of a covert channel and the information required for performing covert channel analysis;**

3) **All assumptions made during covert channel analysis;**

4) **The way to estimate the bandwidth of covert channel in the worst case;**

5) **The maximum availability of each identifiable covert channel;**

6) **USE measures such as blocking, bandwidth limiting, or auditing to process the identified covert channels; demonstrate the effectiveness of the processing measures.**

Note: The resisting to attacks by attackers with **moderate** attack potential needs to be comprehensively considered based on the following 5 specific factors: attack time, attacker ability, level of understanding of the operating system, time of access to operating system or number of attack samples, attack equipment used. See GB/T 30270-2013 Appendix A, A.8.

6.5 Class 5: access verification protection class

6.5.1 Security function requirements

6.5.1.1 Identity authentication

The identity authentication functions of SSOOS are as follows:

a) User identification function:

1) Before users enter the operating system, they shall be identified first;

2) The user identification of operating system shall use the username and UID. Throughout the life cycle of the operating system, achieve the unique identification of the user, as well as the consistency between the username or alias, UID, and the like.

b) User authentication function:

1) Adopt a combination of management-enhanced password authentication and/or biometric authentication and/or digital certificate authentication **and/or authentication based on protocol formal analysis**, etc.; use multiple authentication mechanisms to authenticate the authenticity of the user's identity. Authenticate each time the user logs in to the system and the system is reconnected;

3) Simple attack detection: When a system event is found to match a feature event which potentially violates a system security policy, it shall be able to point out that an event which potentially violates the system security policy is about to happen;

4) **Complex attack detection: Maintain a sequence of events of a known intrusion scenario and a list of feature events which potentially violate system security policies; compare system activity records against feature events and sequence of events. When a system activity is found to match a feature event or sequence of events, it shall be able to point out that an event which potentially violates the system security policy is about to happen.**

d) When a potential security infringement is detected, it shall generate a real-time alarm, terminate the violation process, **cancel the service, disconnect and lock the user account.**

e) It shall provide a selectable query function for the audit log, to support selection and sorting access according to one of the following conditions or logical combination and to be able to export query results:

1) Event type;

2) Date and/or time;

3) User identity;

4) Object name;

5) Success or failure.

f) It shall provide the protection function of audit log:

1) Ensure that the audit mechanism is enabled by default and protect the opening and closing of audit log;

2) Protect audit logs from unauthorized access;

3) Ensure that the audit log is not tampered with and deleted; RECORD the behavior of attempting to tamper with and delete the audit log. **It shall be able to recover audit logs which have been tampered with and deleted.**

g) It shall provide the audit log lookup function in a way which is easy for the user to understand.

6.5.1.7 Trusted path

When the local user and the remote user perform initial login and/or authentication, the operating system shall establish a secure communication path between it and the user. This path carries on the trusted identification to its end point and can protect the authentication communication data from modification and leakage.

6.5.1.8 Trusted channel

A communication channel shall be provided between the operating system and another trusted operating system. This channel is logically isolated from other communication channels. This channel carries on the trusted identification to its end point and can protect the communication data from modification and leakage.

6.5.1.9 Network security protection

The network security protection functions of SSOOS are as follows:

a) SUPPORT two-way network access control based on IP address, port, physical interface, and applications. DISCARD packets which do not meet the pre-defined policy;

b) The network transmission data shall be able to be encrypted and integrity-protected;

c) SUPPORT two-way network trusted access authentication based on system identity and system running state;

d) Network access shall be controlled so that only authorized processes which pass trusted measurement can access the network.

6.5.2 Self-security requirements

6.5.2.1 Operation security protection

The operation security protection functions of SSF are as follows:

a) An installation mechanism shall be provided to set and upgrade configuration parameters. Before initializing and protecting security-related data structures, it shall define the security policy attributes of user and administrator.

b) It shall distinguish normal operation mode and system maintenance mode.

c) Before a normal user accesses the system, the system shall be installed

c) The reference values of integrity measurement shall be stored in a trusted manner, to prevent them from being tampered with;

d) SUPPORT hardware trusted chips as a root of trust.

6.5.2.5 Trusted recovery

When the system fails or the service is interrupted, it shall be ensured that the operating system can automatically restore to a secure operation state **without losing user data**.

6.5.2.6 Security policy configuration

Security policy configuration functions shall be provided for identity authentication, label and mandatory access control, security audit, trusted path, trusted channel, network security protection, resource utilization, and user login access control.

6.5.3 Security assurance requirements

6.5.3.1 Development

6.5.3.1.1 Security architecture

The developer shall provide the security architecture description document of SSOOS. The security architecture description document shall meet the following requirements:

a) Consistent with the description of security function requirements and self-security protection requirements in the SSOOS design document;

b) Describe the security domain of SSOOS;

c) Describe why the SSOOS initialization process is secure;

d) Confirm that SSOOS can prevent damage;

e) Confirm that SSOOS can prevent bypass.

6.5.3.1.2 Functional specification description

The developer shall provide a description of functional specification. The functional specification description shall meet the following requirements:

a) Fully describe SSF and self-security protection;

b) USE a semi-formal mode to describe the SSOOS interface;

e) According to the module, describe security functions and self-security protection;

f) The mapping relationship provided can confirm that all of the behavior described in the design can be mapped to the SSOOS interface which invokes it;

g) **Provide a semi-formal description for each module, including its purpose, interactions, interfaces, return values of other interfaces, interfaces invoked by other modules, and, where appropriate, with informal, explanatory descriptions.**

6.5.3.1.5 SSOOS internal structure

The developer shall provide a description of internal structure of SSOOS and the documentation of the process of argumentation. The internal structure documentation of SSOOS shall meet the following requirements:

a) The argumentation process document is used to determine the characteristics of “reasonable structure” **and complexity**;

b) The internal structure description of SSOOS confirms that the specified entire SSOOS internal structure is reasonable **and not too complicated**.

6.5.3.1.6 Security policy model

Developers shall provide a formal security policy model, including at least: mandatory access control policy, integrity policy. The security policy model shall meet the following requirements:

a) **The model is formal, supplemented with explanatory text as necessary, and identifies modeled security policy;**

b) **For all modeled policies, the model defines the security of the operating system and provides formal proof that the operating system cannot reach an unsecure state;**

c) **The consistency of the model with the functional specification description is discussed in the correct formal level;**

d) **This correspondence indicates that the functional specification description is consistent and complete with respect to the model;**

e) **The correspondence argument indicates that the interface described in the functional specification description is consistent and complete with respect to the mandatory access control policy and the integrity policy.**

6.5.3.4 Testing

6.5.3.4.1 Coverage

The developer shall provide a testing coverage document. The testing coverage document shall meet the following requirements:

a) Confirm the correspondence between the testing in the testing document and the SSOOS interface in the functional specification description;

b) Confirm that all SSOOS interfaces in the functional specification description have been **fully** tested.

6.5.3.4.2 Depth

Developers shall provide the analysis document of testing depth. The analysis document of testing depth shall meet the following requirements:

a) Confirm the correspondence between the testing in the testing document and the security functions and self-security protection module in SSOOS design;

b) Confirm that all security function modules and self-security protection function modules in the SSOOS design have been tested.

6.5.3.4.3 Functional testing

Developers shall test SSF and self-security protection functions. The testing document shall include the following:

a) Testing plan: Identify the tests to be performed and describe the scenario for performing each testing. These scenarios include any sequential dependencies on other testing results;

b) Expected testing results: Indicate the expected output after the testing is completed;

c) Actual testing results: Consistent with the expected testing results;

d) Confirm that known vulnerabilities have been corrected, eliminated, or invalidated, and retesting is performed after the vulnerabilities have been removed, to verify that they have been eliminated and that no new vulnerabilities have been introduced;

e) **An analysis of the sequential dependence of test steps.**

6.5.3.4.4 Independent testing

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

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