



**openEuler**  
**20.03 LTS**

# **Security Hardening Guide**

**Date**      **2020-04-01**

---

# Contents

---

<b>Terms of Use .....</b>	<b>iv</b>
<b>Preface .....</b>	<b>v</b>
<b>1 OS Hardening Overview .....</b>	<b>6</b>
1.1 Security Hardening Purpose.....	6
1.2 Security Hardening Solution .....	6
1.3 Security Hardening Impacts .....	7
<b>2 Security Hardening Guide.....</b>	<b>9</b>
2.1 System Services .....	9
2.1.1 Hardening the SSH Service.....	9
2.2 File Permissions .....	16
2.2.1 Setting the Permissions on and Ownership of Files .....	16
2.2.2 Deleting Unowned Files .....	16
2.2.3 Removing a Symbolic Link to /dev/null .....	17
2.2.4 Setting the umask Value for a Daemon .....	17
2.2.5 Adding a Sticky Bit Attribute to Globally Writable Directories .....	18
2.2.6 Disabling the Globally Writable Permission on Unauthorized Files.....	18
2.2.7 Restricting Permissions on the at Command .....	19
2.2.8 Restricting Permissions on the cron Command.....	19
2.2.9 Restricting Permissions on the sudo Command .....	20
2.3 Kernel Parameters .....	20
2.3.1 Hardening the Security of Kernel Parameters .....	20
2.4 Authentication and Authorization .....	23
2.4.1 Setting a Warning for Remote Network Access.....	23
2.4.2 Forestalling Unauthorized System Restart by Holding Down Ctrl, Alt, and Delete .....	23
2.4.3 Setting an Automatic Exit Interval for Shell.....	24
2.4.4 Setting the Default umask Value for Users to 0077 .....	24
2.4.5 Setting the GRUB2 Encryption Password .....	25
2.4.6 Setting the Secure Single-user Mode .....	26
2.4.7 Disabling Interactive Startup .....	26
2.5 Account Passwords .....	26
2.5.1 Shielding System Accounts.....	26
2.5.2 Restricting Permissions on the su Command.....	27

---

2.5.3 Setting Password Complexity .....	27
2.5.4 Setting the Password Validity Period.....	28
2.5.5 Setting Password Encryption Algorithms .....	29
2.5.6 Locking an Account After Three Login Failures .....	30
2.5.7 Hardening the su Command.....	31
<b>3 Security Hardening Tools.....</b>	<b>32</b>
3.1 Security Hardening Procedure .....	32
3.2 Hardening Items Taking Effect .....	34
<b>4 SELinux Configuration .....</b>	<b>35</b>
<b>5 Appendix .....</b>	<b>37</b>
5.1 Permissions on Files and Directories.....	37
5.2 umask Values.....	37

# Terms of Use

---

**Copyright © Huawei Technologies Co., Ltd. 2020. All rights reserved.**

Your replication, use, modification, and distribution of this document are governed by the Creative Commons License Attribution-ShareAlike 4.0 International Public License (CC BY-SA 4.0). You can visit <https://creativecommons.org/licenses/by-sa/4.0/> to view a human-readable summary of (and not a substitute for) CC BY-SA 4.0. For the complete CC BY-SA 4.0, visit <https://creativecommons.org/licenses/by-sa/4.0/legalcode>.

## **Trademarks and Permissions**

openEuler is a trademark or registered trademark of Huawei Technologies Co., Ltd. All other trademarks and registered trademarks mentioned in this document are the property of their respective holders.

## **Disclaimer**

This document is used only as a guide. Unless otherwise specified by applicable laws or agreed by both parties in written form, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, including but not limited to non-infringement, timeliness, and specific purposes.

---

# Preface

---

## Overview

This document describes how to perform security hardening for openEuler.

## Intended Audience

This document is intended for administrators who need to perform security hardening for openEuler. You must be familiar with the OS security architecture and technologies.

## Symbol Conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
	Indicates a potentially hazardous situation which, if not avoided, could result in equipment damage, data loss, performance deterioration, or unanticipated results. NOTICE is used to address practices not related to personal injury.
 NOTE	Supplements the important information in the main text. NOTE is used to address information not related to personal injury, equipment damage, and environment deterioration.

# 1 OS Hardening Overview

---

This chapter describes the purpose and solution of openEuler system hardening.

[1.1 Security Hardening Purpose](#)

[1.2 Security Hardening Solution](#)

[1.3 Security Hardening Impacts](#)

## 1.1 Security Hardening Purpose

The OS, as the core of the information system, manages hardware and software resources and is the basis of information system security. Applications must depend on the OS to ensure the integrity, confidentiality, availability, and controllability of information. Without the OS security protection, protective methods against hackers and virus attacks at other layers cannot meet the security requirements.

Therefore, security hardening is essential for an OS. Security hardening helps build a dynamic and complete security system, enhance product security, and improve product competitiveness.

## 1.2 Security Hardening Solution

This section describes the openEuler security hardening solution, including the hardening method and items.

### Security Hardening Method

You can manually modify security hardening configurations or run commands to harden the system, or use the security hardening tool to modify security hardening items in batches. The openEuler security hardening tool runs as `openEuler-security.service`. When the system is started for the first time, the system automatically runs the service to execute the default hardening policy, and automatically set the service not to start as the system starts.

You can modify the `security.conf` file and use the security hardening tool to implement user-defined security hardening.

## Security Hardening Items

openEuler security hardening includes the following parts:

- System services
- File permissions
- Kernel parameters
- Authentication and authorization
- Account passwords

### 1.3 Security Hardening Impacts

Security hardening on file permissions and account passwords may change user habits, affecting system usability. For details about common hardening items that affect system usability, see Table 1-1.

**Table 1-1** Security hardening impacts

Item	Suggestion	Impact	Configured By Default
Timeout setting on the text-based user interface (TUI)	When the TUI is idle for a long period of time, it automatically exits.  <b>NOTE</b> When a user logs in to the system using SSH, the timeout period is determined by the smaller value of the <b>TMOUT</b> field in the <b>/etc/profile</b> file and the <b>ClientAliveInterval</b> field in the <b>/etc/ssh/sshd_config</b> file. You are advised to set this parameter to 300 seconds.	If you do not perform any operation on the TUI for a long time, TUI automatically exits.	No
Password complexity	The password is a string containing at least eight characters chosen from three or four of the following types: uppercase letters, lowercase letters, digits, and special characters.	All passwords must comply with the complexity requirements.	No
Password retry limits	If a user fails to enter the correct password for three consecutive times when logging in to the OS, the user account will be locked for 60 seconds.	After the account is locked, the user can log in to the system only after 60 seconds.	Yes
Default <b>umask</b> value	The default <b>umask</b> value of all users is set to <b>077</b> so that the default permission on files created by users is <b>600</b> and the default permission on directories is <b>700</b> .	Users must modify the permission on specified files or directories as required.	Yes
Password	The password validity period can be	When a user	No

Item	Suggestion	Impact	Configured By Default
validity period	modified in the <b>/etc/login.defs</b> file and is set to <b>90 days</b> by default. It can be modified in any time. An expiration notification will be displayed seven days before a password is to expire.	attempts to log in after the password expires, the user will be informed of the password expiry and is required to change the password. If the user does not change the password, the user cannot access the system.	
<b>su</b> permission control	The <b>su</b> command is used to switch user accounts. To improve system security, only the user <b>root</b> and users in the wheel group can use the <b>su</b> command.	Common users can successfully run the <b>su</b> command only after joining in the wheel group.	Yes
Disabling user <b>root</b> from logging in using SSH	Set the value of the <b>PermitRootLogin</b> field in the <b>/etc/ssh/sshd_config</b> file to <b>no</b> . In this way, user <b>root</b> cannot directly log in to the system using SSH.	You need to log in to the system as a common user in SSH mode and then switch to user <b>root</b> .	No
Strong SSH encryption algorithm	The MACs and Ciphers configurations of SSH services support the CTR and SHA2 algorithms and do not support the CBC, MD5, and SHA1 algorithms.	Some early Xshell and PuTTY versions do not support aes128-ctr, aes192-ctr, aes256-ctr, hmac-sha2-256, and hmac-sha2-512 algorithms. Ensure that the latest PuTTY (0.63 or later) and Xshell (5.0 or later) are used.	Yes

# 2 Security Hardening Guide

---

You can modify the hardening policy configuration file or script to harden the system. This chapter describes the hardening items, whether the items are hardened by default, and how to perform security hardening.

- [2.1 System Services](#)
- [2.2 File Permissions](#)
- [2.3 Kernel Parameters](#)
- [2.4 Authentication and Authorization](#)
- [2.5 Account Passwords](#)

## 2.1 System Services

### 2.1.1 Hardening the SSH Service

#### Description

The Secure Shell (SSH) is a reliable security protocol for remote logins and other network services. SSH prevents information disclosure during remote management. SSH encrypts transferred data to prevent domain name server (DNS) spoofing and IP spoofing. OpenSSH was created as an open source alternative to the proprietary SSH protocol.

Hardening the SSH service is to modify configurations of the SSH service to set the algorithm and authentication parameters when the system uses the OpenSSH protocol, improving the system security. Table 2-1 describes the hardening items, recommended hardening values, and default policies.

#### Implementation

To harden a server, perform the following steps:

- Step 1** Open the configuration file `/etc/ssh/sshd_config` of the SSH service on the server, and modify or add hardening items and values in the file.
- Step 2** Save the `/etc/ssh/sshd_config` file.

**Step 3** Run the following command to restart the SSH service:

```
systemctl restart sshd
```

----End

To harden a client, perform the following steps:

**Step 1** Open the configuration file `/etc/ssh/ssh_config` of the SSH service on the client, and modify or add hardening items and values in the file.

**Step 2** Save the `/etc/ssh/ssh_config` file.

**Step 3** Run the following command to restart the SSH service:

```
systemctl restart sshd
```

----End

## Hardening Items

- Server hardening policies

All SSH service hardening items are stored in the `/etc/ssh/sshd_config` configuration file. For details about the server hardening items, hardening suggestions, and whether the hardening items are configured as suggested, see Table 2-1.

**Table 2-1** SSH hardening items on a server

Item	Description	Suggestion	Configured as Suggested
Protocol	SSH protocol version.	2	Yes
SyslogFacility	Log type of the SSH service. The item is set to <b>AUTH</b> , indicating authentication logs.	AUTH	Yes
LogLevel	Level for recording SSHD logs.	VERBOSE	Yes
X11Forwarding	Specifies whether a GUI can be used after login using SSH.	no	Yes
MaxAuthTries	Maximum number of authentication attempts.	3	No
PubkeyAuthentication	Specifies whether public key authentication is allowed.	yes	Yes
RSAAuthentication	Specifies whether only RSA security authentication is allowed.	yes	Yes
IgnoreRhosts	Specifies whether the <b>rhosts</b> and <b>shosts</b> files are used for authentication. The <b>rhosts</b> and <b>shosts</b> files record the names of the servers that support	yes	Yes

Item	Description	Suggestion	Configured as Suggested
	remote access and related login names.		
RhostsRSA Authentication	Specifies whether the RSA algorithm security authentication based on the <b>rhosts</b> file is used. The <b>rhosts</b> file records the names of the servers that support remote access and related login names.	no	Yes
HostbasedAuthentication	Specifies whether host-based authentication is used. Host-based authentication indicates that any user of a trusted client can use the SSH service.	no	Yes
PermitRootLogin	Specifies whether to allow user <b>root</b> to log in to the system using SSH. <b>NOTE</b> If you want to log in to the system using SSH as user <b>root</b> , set the value of the <b>PermitRootLogin</b> field in the <b>/etc/ssh/sshd_config</b> file to <b>yes</b> .	no	No
PermitEmptyPasswords	Specifies whether accounts with empty passwords can log in.	no	Yes
PermitUserEnvironment	Specifies whether to resolve the environment variables set in <b>~/.ssh/environment</b> and <b>~/.ssh/authorized_keys</b> .	no	Yes
Ciphers	Encryption algorithm of SSH data transmission.	aes128-ctr,aes192-ctr,aes256-ctr,ChaCha20-Poly1305@openssh.com,aes128-gcm@openssh.com,aes256-gcm@openssh.com	Yes
ClientAliveInterval	Timeout period of the system (in seconds). If no response from the client is received in the specific period, the server automatically disconnects from the client.	300	No
ClientAliveCountMax	Timeout count. After the server sends a request, if the number of times that the client does not respond reaches a	0	No

Item	Description	Suggestion	Configured as Suggested
	specified value, the server automatically disconnects from the client.		
Banner	File of the prompt information displayed before and after SSH login.	/etc/issue.net	Yes
MACs	Hash algorithm for SSH data verification.	hmac-sha2-512,hmac-sha2-512-etm@openssh.com,hmac-sha2-256,hmac-sha2-256-etm@openssh.com,hmac-sha1,hmac-sha1-etm@openssh.com	Yes
StrictModes	Specifies whether to check the permission on and ownership of the home directory and <b>rhosts</b> file before SSH receives login requests.	yes	Yes
UsePAM	Specifies whether to use PAM for login authentication.	yes	Yes
AllowTcpForwarding	Specifies whether to allow TCP forwarding.	no	Yes
Subsystem sftp /usr/libexec/openssh/sftp-server	SFTP log record level, which records the INFO level and authentication logs.	-l INFO -f AUTH	Yes
AllowAgentForwarding	Specifies whether to allow SSH Agent forwarding.	no	Yes
GatewayPorts	Specifies whether SSH can connect to ports on the forwarding client.	no	Yes
PermitTunnel	Specifies whether Tunnel devices are allowed.	no	Yes
KexAlgorithms	SSH key exchange algorithms.	curve25519-sha256,curve25519-sha	Yes

Item	Description	Suggestion	Configured as Suggested
		256@@1ibssh.org,diffie-hellman-group14-sha1,diffie-hellman-group-exchange-sha1,diffie-hellman-group-exchange-sha256	
LoginGrace Time	Time limit for users passing the authentication. <b>0</b> indicates no limit. The default value is <b>60</b> seconds.	60	No

 **NOTE**

By default, the messages displayed before and after SSH login are saved in the `/etc/issue.net` file. The default information in the `/etc/issue.net` file is **Authorized users only. All activities may be monitored and reported.**

- Client hardening policies

All SSH service hardening items are stored in the `/etc/ssh/ssh_config` configuration file. For details about the client hardening items, hardening suggestions, and whether the hardening items are configured as suggested, see Table 2-2.

**Table 2-2** SSH hardening items on a client

Item	Description	Suggestion	Configured as Suggested
KexAlgorithms	SSH key exchange algorithms.	ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256,diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1	No
VerifyHostKeyDNS	Specifies whether to verify Host	ask	No

Item	Description	Suggestion	Configured as Suggested
	Key files by using DNS or SSHFP.		

 **NOTE**

Third-party clients and servers that use the Diffie-Hellman algorithm are required to allow at least 2048-bit connection.

## Other Security Suggestions

- The SSH service only listens on specified IP addresses.  
For security purposes, you are advised to only listen on required IP addresses rather than listen on 0.0.0.0 when using the SSH service. You can specify the IP addresses that SSH needs to listen on in the ListenAddress configuration item in the `/etc/ssh/sshd_config` file.
  - a. Open and modify the `/etc/ssh/sshd_config` file.  

```
vi /etc/ssh/sshd_config
```

The following information indicates that the bound listening IP address is **192.168.1.100**. You can change the listening IP address based on the site requirements.

```
...
ListenAddress 192.168.1.100
...
```
  - b. Restart the SSH service.  

```
systemctl restart sshd.service
```
- SFTP users are restricted from access to upper-level directories.  
SFTP is a secure FTP designed to provide secure file transfer over SSH. Users can only use dedicated accounts to access SFTP for file upload and download, instead of SSH login. In addition, directories that can be accessed over SFTP are limited to prevent directory traversal attacks. The configuration process is as follows:

 **NOTE**

In the following configurations, `sftpgroup` is an example user group name, and `sftpuser` is an example username.

- a. Create an SFTP user group.  

```
groupadd sftpgroup
```
- b. Create an SFTP root directory.

- ```
mkdir /sftp
```
- c. Modify the ownership of and permission on the SFTP root directory.
- ```
chown root:root /sftp
chmod 755 /sftp
```
- d. Create an SFTP user.
- ```
useradd -g sftpgroup -s /sbin/nologin sftpuser
```
- e. Set the password of the SFTP user.
- ```
passwd sftpuser
```
- f. Create a directory used to store files uploaded by the SFTP user.
- ```
mkdir /sftp/sftpuser
```
- g. Modify the ownership of and permission on the upload directory of the SFTP user.
- ```
chown root:root /sftp/sftpuser
chmod 777 /sftp/sftpuser
```
- h. Modify the `/etc/ssh/sshd_config` file.
- ```
vi /etc/ssh/sshd_config
```
- Modify the following information:
- ```
#Subsystem sftp /usr/libexec/openssh/sftp-server -l INFO -f AUTH
Subsystem sftp internal-sftp -l INFO -f AUTH
...

Match Group sftpgroup
    ChrootDirectory /sftp/%u
    ForceCommand internal-sftp
```

#### NOTE

- `%u` is a wildcard character. Enter `%u` to represent the username of the current SFTP user.
- The following content must be added to the end of the `/etc/ssh/sshd_config` file:

```
Match Group sftpgroup
    ChrootDirectory /sftp/%u
    ForceCommand internal-sftp
```

- i. Restart the SSH service.

```
systemctl restart sshd.service
```

- Remotely execute commands using SSH.

When a command is executed remotely through OpenSSH, TTY is disabled by default. If a password is required during command execution, the password is displayed in plain text. To ensure password input security, you are advised to add the `-t` option to the command. Example:

```
ssh -t testuser@192.168.1.100 su
```

#### NOTE

`192.168.1.100` is an example IP address, and `testuser` is an example username.

## 2.2 File Permissions

### 2.2.1 Setting the Permissions on and Ownership of Files

#### Description

In Linux, all objects are processed as files. Even a directory will be processed as a large file containing many files. Therefore, the most important thing in Linux is the security of files and directories. Their security is ensured by permissions and owners.

By default, the permissions and ownership of common directories, executable files, and configuration files in the system are set in openEuler.

#### Implementation

The following uses the **/bin** directory as an example to describe how to change the permission and ownership of a file:

- Modify the file permission. For example, set the permission on the **/bin** directory to **755**.

```
chmod 755 /bin
```

- Change the ownership of the file. For example, set the ownership and group of the **/bin** directory to **root:root**.

```
chown root:root /bin
```

### 2.2.2 Deleting Unowned Files

#### Description

When deleting a user or group, the system administrator may forget to delete the files of the user or group. If the name of a new user or group is the same as that of the deleted user or group, the new user or group will own files on which it has no permission. You are advised to delete these files.

#### Implementation

Delete the file whose user ID does not exist.

- Step 1** Search for the file whose user ID does not exist.

```
find / -nouser
```

- Step 2** Delete the found file. In the preceding command, *filename* indicates the name of the file whose user ID does not exist.

```
rm -f filename
```

----End

Delete the file whose group ID does not exist.

- Step 1** Search for the file whose user ID does not exist.

```
find / -nogroup
```

- Step 2** Delete the found file. In the preceding command, *filename* indicates the name of the file whose user ID does not exist.

```
rm -f filename
```

----End

## 2.2.3 Removing a Symbolic Link to /dev/null

### Description

A symbolic link to **/dev/null** may be used by malicious users. This affects system security. You are advised to delete these symbolic links to improve system security.

### Special Scenario

After openEuler is installed, symbolic links to **/dev/null** may exist. These links may have corresponding functions. (Some of them are preconfigured and may be depended by other components.) Rectify the fault based on the site requirements. For details, see [Implementation](#).

For example, openEuler supports UEFI and legacy BIOS installation modes. The GRUB packages supported in the two boot scenarios are installed by default. If you select the legacy BIOS installation mode, a symbolic link **/etc/grub2-efi.cfg** is generated. If you select the UEFI installation mode, a symbolic link **/etc/grub2.cfg** is generated. You need to process these symbolic links based on the site requirements.

### Implementation

- Step 1** Run the following command to search for symbolic links to **/dev/null**:

```
find dirname -type l -follow 2>/dev/null
```

#### NOTE

*dirname* indicates the directory to be searched. Normally, key system directories, such as **/bin**, **/boot**, **/usr**, **/lib64**, **/lib**, and **/var**, need to be searched.

- Step 2** If these symbolic links are useless, run the following command to delete them:

```
rm -f filename
```

#### NOTE

*filename* indicates the file name obtained in [Step 1](#).

----End

## 2.2.4 Setting the umask Value for a Daemon

### Description

The **umask** value is used to set default permission on files and directories. If the **umask** value is not specified, the file has the globally writable permission. This brings risks. A daemon provides a service for the system to receive user requests or network customer requests. To improve the security of files and directories created by the daemon, you are advised to set

**umask** to **0027**. The **umask** value indicates the complement of a permission. For details about how to convert the **umask** value to a permission, see 5.2 umask Values.

 **NOTE**

By default, the **umask** value of the daemon is set to **0027** in openEuler.

## Implementation

In configuration file `/etc/sysconfig/init`, add **umask 0027** as a new row.

## 2.2.5 Adding a Sticky Bit Attribute to Globally Writable Directories

### Description

Any user can delete or modify a file or directory in a globally writable directory, which leads to unauthorized file or directory deletion. Therefore, the sticky bit attribute is required for globally writable directories.

### Implementation

**Step 1** Search for globally writable directories.

```
find / -type d -perm -0002 ! -perm -1000 -ls | grep -v proc
```

**Step 2** Add the sticky bit attribute to globally writable directories. *dirname* indicates the name of the directory that is found.

```
chmod +t dirname
```

----End

## 2.2.6 Disabling the Globally Writable Permission on Unauthorized Files

### Description

Any user can modify globally writable files, which affects system integrity.

### Implementation

**Step 1** Search for all globally writable files.

```
find / -type d \( -perm -o+w \) | grep -v proc  
find / -type f \( -perm -o+w \) | grep -v proc
```

**Step 2** View the settings of files (excluding files and directories with sticky bits) listed in step 1, and delete the files or disable the globally writable permission on them. Run the following command to remove the permission. In the command, *filename* indicates the file name.

```
chmod o-w filename
```

### NOTE

You can run the following command to check whether the sticky bit is set for the file or directory. If the command output contains the **T** flag, the file or directory is with a sticky bit. In the command, *filename* indicates the name of the file or directory to be queried.

```
ls -l filename
```

----End

## 2.2.7 Restricting Permissions on the at Command

### Description

The **at** command is used to create a scheduled task. Users who can run the **at** command must be specified to protect the system from being attacked.

### Implementation

**Step 1** Delete the `/etc/at.deny` file.

```
rm -f /etc/at.deny
```

**Step 2** Run the following command to change the ownership of file `/etc/at.allow` file to **root:root**.

```
chown root:root /etc/at.allow
```

**Step 3** Set that only user **root** can operate file `/etc/at.allow`.

```
chmod og-rwx /etc/at.allow
```

----End

## 2.2.8 Restricting Permissions on the cron Command

### Description

The **cron** command is used to create a routine task. Users who can run the **cron** command must be specified to protect the system from being attacked.

### Implementation

**Step 1** Delete the `/etc/cron.deny` file.

```
rm -f /etc/at.deny
```

**Step 2** Run the following command to change the ownership of the `/etc/cron.allow` file to **root:root**:

```
chown root:root /etc/cron.allow
```

**Step 3** Set that only user **root** can operate file `/etc/cron.allow`.

```
chmod og-rwx /etc/cron.allow
```

----End

## 2.2.9 Restricting Permissions on the sudo Command

### Description

A common user can use the **sudo** command to run commands as the user **root**. To harden system security, it is necessary to restrict permissions on the **sudo** command. Only user **root** can use the **sudo** command.

### Implementation

Modify the `/etc/sudoers` file to restrict permissions on the **sudo** command. Comment out the following configuration line:

```
#%wheel ALL=(ALL) ALL
```

## 2.3 Kernel Parameters

### 2.3.1 Hardening the Security of Kernel Parameters

#### Description

Kernel parameters specify the status of network configurations and application privileges. The kernel provides system control which can be fine-tuned or configured by users. This function can improve the security of the OS by controlling configurable kernel parameters. For example, you can fine-tune or configure network options to improve system security.

#### Implementation

**Step 1** Write the hardening items in Table 2-3 to the `/etc/sysctl.conf` file.

#### NOTE

Record security hardening items as follows:

```
net.ipv4.icmp_echo_ignore_broadcasts = 1  
net.ipv4.conf.all.rp_filter = 1  
net.ipv4.conf.default.rp_filter = 1
```

**Table 2-3** Policies for hardening the security of kernel parameters

Item	Description	Suggestion	Configured as Suggested
net.ipv4.icmp_echo_ignore_broadcasts	Specifies whether ICMP broadcast packets are accepted. They are not accepted according to the hardening policy.	1	Yes
net.ipv4.conf.all.rp_filter	Specifies whether the actual source IP address used by a data packet is	1	Yes

Item	Description	Suggestion	Configured as Suggested
net.ipv4.conf.default.rp_filter	related to a routing table and whether the data packet receives responses through interfaces. The item is enabled according to the hardening policy.	1	Yes
net.ipv4.ip_forward	The IP forwarding function prevents unauthorized IP address packets from being transferred to a network. The item is disabled according to the hardening policy.	0	Yes
net.ipv4.conf.all.accept_source_route	<b>accept_source_route</b> indicates that a packet sender can specify a path for sending the packet and a path for receiving a response. The item is disabled according to the hardening policy.	0	Yes
net.ipv4.conf.default.accept_source_route		0	Yes
net.ipv4.conf.all.accept_redirects	Specifies whether a redirected ICMP packet is sent. The packet is not sent according to the hardening policy.	0	Yes
net.ipv4.conf.default.accept_redirects		0	Yes
net.ipv6.conf.all.accept_redirects		0	Yes
net.ipv6.conf.default.accept_redirects		0	Yes
net.ipv4.conf.all.send_redirects	Specifies whether a redirected ICMP packet is sent to another server. This item is enabled only when the host functions as a route. The item is disabled according to the hardening policy.	0	Yes
net.ipv4.conf.default.send_redirects		0	Yes
net.ipv4.icmp_ignore_bogus_error_responses	Fake ICMP packets are not recorded to logs, which saves disk space. The item is enabled according to the hardening policy.	1	Yes
net.ipv4.tcp_syncookies	SYN attack is a DoS attack that forces system restart by occupying system resources. TCP-SYN cookie protection is enabled according to the hardening policy.	1	Yes
kernel.dmesg_restrict	Hardens dmesg messages. Only the administrator is allowed to view the messages.	1	Yes

Item	Description	Suggestion	Configured as Suggested
kernel.sched_autogroup_enabled	Determines whether the kernel automatically groups and schedules threads. After this item is enabled, scheduling groups compete for time slices, and threads in a scheduling group compete for the time slices allocated to the scheduling group. The item is disabled according to the hardening policy.	0	No
kernel.sysrq	Disables the magic key. <b>NOTE</b> You are advised to disable the magic key so that commands cannot be directly passed to the kernel.	0	Yes
net.ipv4.conf.all.secure_redirects	Specifies whether redirected ICMP messages sent from any servers or from gateways listed in the default gateway list are accepted. Redirected ICMP messages are received from any servers according to the hardening policy.	0	Yes
net.ipv4.conf.default.secure_redirects		0	Yes

**Step 2** Run the following command to load the kernel parameters set in the `sysctl.conf` file:

```
sysctl -p /etc/sysctl.conf
```

----End

## Other Security Suggestions

- net.ipv4.icmp\_echo\_ignore\_all**: ignores ICMP requests.

For security purposes, you are advised to enable this item. The default value is **0**. Set the value to **1** to enable this item.

After this item is enabled, all incoming ICMP Echo request packets will be ignored, which will cause failure to ping the target host. Determine whether to enable this item based on your actual networking condition.
- net.ipv4.conf.all.log\_martians/net.ipv4.conf.default.log\_martians**: logs spoofed, source routed, and redirect packets.

For security purposes, you are advised to enable this item. The default value is **0**. Set the value to **1** to enable this item.

After this item is enabled, data from forbidden IP addresses will be logged. Too many new logs will overwrite old logs because the total number of logs allowed is fixed. Determine whether to enable this item based on your actual usage scenario.
- net.ipv4.tcp\_timestamps**: disables tcp\_timestamps.

For security purposes, you are advised to disable `tcp_timestamps`. The default value is `1`. Set the value to `0` to disable `tcp_timestamps`.

After this item is disabled, TCP retransmission timeout will be affected. Determine whether to disable this item based on the actual usage scenario.

- **net.ipv4.tcp\_max\_syn\_backlog**: determines the number of queues that is in `SYN_RECV` state.

This parameter determines the number of queues that is in `SYN_RECV` state. When this number is exceeded, new TCP connection requests will not be accepted. This to some extent prevents system resource exhaustion. Configure this parameter based on your actual usage scenario.

## 2.4 Authentication and Authorization

### 2.4.1 Setting a Warning for Remote Network Access

#### Description

A warning for remote network access is configured and displayed for users who attempt to remotely log in to the system. The warning indicates the penalty for authorized access and is used to threaten potential attackers. When the warning is displayed, system architecture and other system information are hidden to protect the system from being attacked.

#### Implementation

This setting can be implemented by modifying the `/etc/issue.net` file. Replace the original content in the `/etc/issue.net` file with the following information (which has been set by default in openEuler):

```
Authorized users only. All activities may be monitored and reported.
```

### 2.4.2 Forestalling Unauthorized System Restart by Holding Down Ctrl, Alt, and Delete

#### Description

By default, you can restart the OS by holding down **Ctrl**, **Alt**, and **Delete**. Disabling this feature can prevent data loss caused by misoperations.

#### Implementation

Shield the **Ctrl+Alt+Del** response function of the kernel keyboard.

```
rm -f /etc/systemd/system/ctrl-alt-del.target
rm -f /usr/lib/systemd/system/ctrl-alt-del.target
```

#### NOTE

The following file is reserved because the Xen driver needs to be invoked and the system cannot respond to the **Ctrl+Alt+Del** operation. Therefore, there is no impact.

/usr/lib/systemd/system/ctrl-alt-del.target

## 2.4.3 Setting an Automatic Exit Interval for Shell

### Description

An unattended shell is prone to listening or attacks. Therefore, a mechanism must be configured to ensure that a shell can automatically exit when it does not run for a period.

### Implementation

At the end of file `/etc/profile`, set the **TMOUT** field (unit: second) that specifies the interval for automatic exit as follows:

```
export TMOUT=300
```

## 2.4.4 Setting the Default umask Value for Users to 0077

### Description

The **umask** value is used to set default permission on files and directories. A smaller **umask** value indicates that group users or other users have incorrect permission, which brings system security risks. Therefore, the default **umask** value must be set to **0077** for all users, that is, the default permission on user directories is **700** and the permission on user files is **600**. The **umask** value indicates the complement of a permission. For details about how to convert the **umask** value to a permission, see 5.2 umask Values.

#### NOTE

By default, the **umask** value of the openEuler user is set to **0077**.

### Implementation

**Step 1** Add **umask 0077** to the `/etc/bashrc` file and all files in the `/etc/profile.d/` directory.

```
echo "umask 0077" >> $FILE
```

#### NOTE

`$FILE` indicates the file name, for example, `echo "umask 0077" >> /etc/bashrc`.

**Step 2** Set the ownership and group of the `/etc/bashrc` file and all files in the `/etc/profile.d/` directory to **root**.

```
chown root.root $FILE
```

#### NOTE

`$FILE` indicates the file name, for example, `chown root.root /etc/bashrc`.

----End

## 2.4.5 Setting the GRUB2 Encryption Password

### Description

GRand Unified Bootloader (GRUB) is an operating system boot manager used to boot different systems (such as Windows and Linux). GRUB2 is an upgraded version of GRUB.

When starting the system, you can modify the startup parameters of the system on the GRUB2 screen. To ensure that the system startup parameters are not modified randomly, you need to encrypt the GRUB2 screen. The startup parameters can be modified only when the correct GRUB2 password is entered.

#### NOTE

The default password of GRUB2 is **openEuler#12**. You are advised to change the default password upon the first login and periodically update the password. If the password is leaked, startup item configurations may be modified, causing the system startup failure.

### Implementation

**Step 1** Run the **grub2-mkpasswd-pbkdf2** command to generate an encrypted password.

#### NOTE

SHA-512 is used as the GRUB2 encryption algorithm.

```
# grub2-mkpasswd-pbkdf2
Enter password:
Reenter password:
PBKDF2 hash of your password is
grub.pbkdf2.sha512.10000.5A45748D892672FDA02DD3B6F7AE390AC6E6D532A600D4AC477D25C7D
087644697D8A0894DFED9D86DC2A27F4E01D925C46417A225FC099C12DBD3D7D49A7425.2BD2F5BF49
07DCC389CC5D165DB85CC3E2C94C8F9A30B01DACC9CD552B731BA1DD3B7CC2C765704D55B8CD962D2
AEF19A753CBE9B8464E2B1EB39A3BB4EAB08
```

#### NOTE

Enter the same password in the **Enter password** and **Reenter password** lines.

After **openEuler#12** is encrypted by **grub2-mkpasswd-pbkdf2**, the output is **grub.pbkdf2.sha512.10000.5A45748D892672FDA02DD3B6F7AE390AC6E6D532A600D4AC477D25C7D087644697D8A0894DFED9D86DC2A27F4E01D925C46417A225FC099C12DBD3D7D49A7425.2BD2F5BF4907DCC389CC5D165DB85CC3E2C94C8F9A30B01DACC9CD552B731BA1DD3B7CC2C765704D55B8CD962D2AEF19A753CBE9B8464E2B1EB39A3BB4EAB08**. The ciphertext is different each time.

**Step 2** Open **/boot/efi/EFI/openEuler/grub.cfg** in a vi editor. Append the following fields to the beginning of **/boot/efi/EFI/openEuler/grub.cfg**.

```
set superusers="root"
password pbkdf2 root
grub.pbkdf2.sha512.10000.5A45748D892672FDA02DD3B6F7AE390AC6E6D532A600D4AC477D25C7D
087644697D8A0894DFED9D86DC2A27F4E01D925C46417A225FC099C12DBD3D7D49A7425.2BD2F5BF49
07DCC389CC5D165DB85CC3E2C94C8F9A30B01DACC9CD552B731BA1DD3B7CC2C765704D55B8CD962D2
AEF19A753CBE9B8464E2B1EB39A3BB4EAB08
```

#### NOTE

- The **superusers** field is used to set the account name of the super GRUB2 administrator.
- The first parameter following the **password\_pbkdf2** field is the GRUB2 account name, and the second parameter is the encrypted password of the account.

----End

## 2.4.6 Setting the Secure Single-user Mode

### Description

When you log in to the system as user **root** in single-user mode, if the **root** password is not set, high security risks exist.

### Implementation

This setting can be implemented by modifying the `/etc/sysconfig/init` file. Set **SINGLE** to **SINGLE=/sbin/sulogin**.

## 2.4.7 Disabling Interactive Startup

### Description

With interactive guidance, console users can disable audit, firewall, or other services, which compromises system security. Users can disable interactive startup to improve security. This item is disabled by default in openEuler.

### Implementation

This setting can be implemented by modifying the `/etc/sysconfig/init` file. Set **PROMPT** to **no**.

## 2.5 Account Passwords

### 2.5.1 Shielding System Accounts

#### Description

Accounts excluding user accounts are system accounts. System accounts cannot be used for logins or performing other operations. Therefore, system accounts must be shielded.

#### Implementation

Modify the shell of a system account to **/sbin/nologin**.

```
usermod -L -s /sbin/nologin $systemaccount
```

#### NOTE

*\$systemaccount* indicates the system account.

## 2.5.2 Restricting Permissions on the su Command

### Description

The **su** command is used to switch user accounts. To improve system security, only the user **root** and users in the wheel group can use the **su** command.

### Implementation

Modify the `/etc/pam.d/su` file as follows:

```
auth      required      pam_wheel.so use_uid
```

**Table 2-4** Configuration item in pam\_wheel.so

Item	Description
use_uid	UID of the current account.

## 2.5.3 Setting Password Complexity

### Description

You can set the password complexity requirements by modifying the corresponding configuration file. You are advised to set the password complexity based on the site requirements.

### Implementation

The password complexity is implemented by the **pam\_pwquality.so** and **pam\_pwhistory.so** modules in the `/etc/pam.d/password-auth` and `/etc/pam.d/system-auth` files. You can modify the configuration items of the two modules to change the password complexity requirements.

### Example

This section provides an example for configuring password complexity.

#### Password Complexity Requirements

1. Contains at least eight characters.
2. Contains at least three types of the following characters:
  - At least one lowercase letter
  - At least one uppercase letter
  - At least one digit
  - At least one space or one of the following special characters: ` ~ ! @ # \$ % ^ & \* ( ) - \_ = + \ | [ { } ] ; : ' " , < . > / ?
3. Cannot be the same as an account or the account in reverse order.

- Cannot be the last five passwords used.

### Implementation

Add the following content to the `/etc/pam.d/password-auth` and `/etc/pam.d/system-auth` files:

```
password requisite pam_pwquality.so minlen=8 minclass=3 enforce_for_root
try first pass local users only retry=3 dcredit=0 ucredit=0 lcredit=0 ocredit=0
password required pam_pwhistory.so use_authtok remember=5 enforce_for_root
```

### Configuration Item Description

For details about the configuration items of `pam_pwquality.so` and `pam_pwhistory.so`, see Table 2-5 and Table 2-6, respectively.

**Table 2-5** Configuration items in `pam_pwquality.so`

Item	Description
<code>minlen=8</code>	A password must contain at least eight characters.
<code>minclass=3</code>	A password must contain at least three of the following types: uppercase letters, lowercase letters, digits, and special characters.
<code>ucredit=0</code>	A password contains any number of uppercase letters.
<code>lcredit=0</code>	A password contains any number of lowercase letters.
<code>dcredit=0</code>	A password contains any number of digits.
<code>ocredit=0</code>	A password contains any number of special characters.
<code>retry=3</code>	Each time a maximum of three password changes is allowed.
<code>enforce_for_root</code>	This configuration is also effective for user <b>root</b> .

**Table 2-6** Configuration items in `pam_pwhistory.so`

Item	Description
<code>remember=5</code>	A password must be different from the last five passwords used.
<code>enforce_for_root</code>	This configuration is also effective for user <b>root</b> .

## 2.5.4 Setting the Password Validity Period

### Description

To ensure system security, you are advised to set the password validity period and notify users to change passwords before the passwords expire.

## Implementation

The password validity period is set by modifying the `/etc/login.defs` file. Table 2-7 describes the hardening items. All hardening items in the table are in the `/etc/login.defs` file. You can directly modify the items in the configuration file.

**Table 2-7** Configuration items in login.defs

Item	Description	Suggestion	Configured as Suggested
PASS_MAX_DAYS	Maximum validity period of a password.	90	No
PASS_MIN_DAYS	Minimum interval between password changes.	0	No
PASS_WARN_AGE	Number of days before the password expires.	7	No

### NOTE

The `login.defs` file is used to set restrictions on user accounts, such as setting the maximum password validity period and maximum length. The configuration in this file is invalid for the user `root`. If the `/etc/shadow` file contains the same items, the `/etc/shadow` configuration takes precedence over the `/etc/login.defs` configuration. When a user attempts to log in after the password expires, the user will be informed of the password expiry and is required to change the password. If the user does not change the password, the user cannot access the system.

## 2.5.5 Setting Password Encryption Algorithms

### Description

For system security, passwords cannot be stored in plaintext in the system and must be encrypted. The passwords that do not need to be restored must be encrypted using irreversible algorithms. Set the password encryption algorithm to SHA-512. This item has been set by default in openEuler. The preceding settings can effectively prevent password disclosure and ensure password security.

### Implementation

To set the password encryption algorithm, add the following configuration to the `/etc/pam.d/password-auth` and `/etc/pam.d/system-auth` files:

```
password sufficient pam_unix.so sha512 shadow nullok try_first_pass use_authok
```

**Table 2-8** Configuration items in pam\_unix.so

Item	Description
------	-------------

Item	Description
sha512	The SHA-512 algorithm is used for password encryption.

## 2.5.6 Locking an Account After Three Login Failures

### Description

To ensure user system security, you are advised to set the maximum number of incorrect password attempts (three attempts are recommended) and the automatic unlocking time (300 seconds are recommended) for a locked account.

If an account is locked, any input is invalid but does not cause the locking timer to recount. Records of the user's invalid inputs are cleared once unlocked. The preceding settings protect passwords from being forcibly cracked and improve system security.

#### NOTE

By default, the maximum number of incorrect password attempts is 3 in openEuler. After the system is locked, the automatic unlock time is 60 seconds.

### Implementation

The password complexity is set by modifying the `/etc/pam.d/password-auth` and `/etc/pam.d/system-auth` files. The maximum number of incorrect password attempts is set to **3**, and the unlocking time after the system is locked is set to **300** seconds. The configuration is as follows:

```
auth      required      pam faillock.so preauth audit deny=3 even deny root
unlock   time=300
auth      [default=die] pam faillock.so authfail audit deny=3 even deny root
unlock   time=300
auth      sufficient    pam faillock.so authsucc audit deny=3 even deny root
unlock   time=300
```

**Table 2-9** Configuration items in `pam_faillock.so`

Item	Description
authfail	Captures account login failure events.
deny=3	A user account will be locked after three login attempts.
unlock_time=300	A locked common user account is automatically unlocked in 300 seconds.
even_deny_root	This configuration is also effective for user <b>root</b> .

## 2.5.7 Hardening the su Command

### Description

To enhance system security and prevent the environment variables of the current user from being brought into other environments when you run the **su** command to switch to another user, this item has been configured by default in openEuler. The **PATH** variable is always initialized when the **su** command is used to switch users.

### Implementation

Modify the **/etc/login.defs** file. The configuration is as follows:

```
ALWAYS_SET_PATH=yes
```

---

# 3 Security Hardening Tools

---

[3.1 Security Hardening Procedure](#)

[3.2 Hardening Items Taking Effect](#)

## 3.1 Security Hardening Procedure

### Overview

You need to modify the **usr-security.conf** file so that the security hardening tool can set hardening policies based on the **usr-security.conf** file. This section describes rules for modifying the **usr-security.conf** file. For details about the configurable security hardening items, see 2 Security Hardening Guide.

### Precautions

- After modifying the items, restart the security hardening service for the modification to take effect. For details about how to restart the service, see 3.2 Hardening Items Taking Effect.
- When modifying security hardening items, you only need to modify the **/etc/openEuler\_security/usr-security.conf** file. You are not advised to modify the **/etc/openEuler\_security/security.conf** file. The **security.conf** file contains basic hardening items which are executed only once.
- After the security hardening service is restarted for the configuration to take effect, the previous configuration cannot be deleted by deleting the corresponding hardening items from the **usr-security.conf** file and restarting the security hardening service.
- Security hardening operations are recorded in the **/var/log/openEuler-security.log** file.

### Configuration Format

Each line in the **usr-security.conf** file indicates a configuration item. The configuration format varies according to the configuration content. The following describes the format of each configuration item.

#### NOTE

- All configuration items start with an execution ID. The execution ID is a positive integer and can be customized.

- Contents of a configuration item are separated by an at sign (@).
- If the actual configuration content contains an at sign (@), use two at signs (@@) to distinguish the content from the separator. For example, if the actual content is **xxx@yyy**, set this item to **xxx@@yyy**. Currently, an at sign (@) cannot be placed at the beginning or end of the configuration content.

- **d**: comment

Format: *Execution ID@d@Object file@Match item*

Function: Comment out lines starting with the match item (the line can start with a space) in an object file by adding a number sign (#) at the beginning of the line.

Example: If the execution ID is **401**, comment out lines starting with **%wheel** in the **/etc/sudoers** file.

```
401@d@/etc/sudoers@%wheel
```

- **m**: replacement

Format: *Execution ID@m@Object file@Match item@Target value*

Function: Replace lines starting with the match item (the line can start with a space) in an object file with *match item* and *target value*. If the match line starts with spaces, the spaces will be deleted after the replacement.

Example: If the execution ID is **101**, replace lines starting with **Protocol** in the **/etc/ssh/sshd\_config** file with **Protocol 2**. The spaces after **Protocol** are matched and replaced.

```
101@m@/etc/ssh/sshd_config@Protocol @2
```

- **sm**: accurate modification

Format: *Execution ID@sm@Object file@Match item@Target value*

Function: Replace lines starting with the match item (the line can start with a space) in an object file with *match item* and *target value*. If the match line starts with spaces, the spaces are retained after the replacement. This is the difference between **sm** and **m**.

Example: If the execution ID is **201**, replace lines starting with **size** in the **/etc/audit/hzqtest** file with **size 2048**.

```
201@sm@/etc/audit/hzqtest@size @2048
```

- **M**: subitem modification

Format: *Execution ID@M@Object file@Match item@Match subitem[@Value of the match subitem]*

Function: Match lines starting with the match item (the line can start with a space) in an object file and replace the content starting with the match subitem in these lines with the *match subitem* and *value of the match subitem*. The value of the match subitem is optional.

Example: If the execution ID is **101**, find lines starting with **key** in the file and replace the content starting with **key2** in these lines with **key2value2**.

```
101@M@file@key@key2@value2
```

- **systemctl**: service management

Format: *Execution ID@systemctl@Object service@Operation*

Function: Use **systemctl** to manage object services. The value of **Operation** can be **start**, **stop**, **restart**, or **disable**.

Example: If the execution ID is **218**, stop the **cups.service**. This provides the same function as running the **systemctl stop cups.service** command.

```
218@systemctl @cups.service@stop
```

- Other commands

Format: *Execution ID@Command@Object file*

Function: Run the corresponding command, that is, run the command line *Command Object file*.

Example 1: If the execution ID is **402**, run the **rm -f** command to delete the **/etc/pki/ca-trust/extracted/pem/email-ca-bundle.pem** file.

```
402@rm -f @/etc/pki/ca-trust/extracted/pem/email-ca-bundle.pem
```

Example 2: If the execution ID is **215**, run the **touch** command to create the **/etc/cron.allow** file.

```
215@touch @/etc/cron.allow
```

Example 3: If the execution ID is **214**, run the **chown** command to change the owner of the **/etc/at.allow** file to **root:root**.

```
214@chown root:root @/etc/at.allow
```

Example 4: If the execution ID is **214**, run the **chmod** command to remove the **rwX** permission of the group to which the owner of the **/etc/at.allow** file belongs and other non-owner users.

```
214@chmod og-rwx @/etc/at.allow
```

## 3.2 Hardening Items Taking Effect

After modifying the **usr-security.conf** file, run the following command for the new configuration items to take effect:

```
systemctl restart openEuler-security.service
```

---

# 4 SELinux Configuration

---

## Overview

Discretionary access control (DAC) determines whether a resource can be accessed based on users, groups, and other permissions. It does not allow the system administrator to create comprehensive and fine-grained security policies. SELinux (Security-Enhanced Linux) is a module of the Linux kernel and a security subsystem of Linux. SELinux implements mandatory access control (MAC). Each process and system resource has a special security label. In addition to the principles specified by the DAC, the SELinux needs to determine whether each type of process has the permission to access a type of resource.

By default, openEuler uses SELinux to improve system security. SELinux has three modes:

- **permissive:** The SELinux outputs alarms but does not forcibly execute the security policy.
- **enforcing:** The SELinux security policy is forcibly executed.
- **disabled:** The SELinux security policy is not loaded.

## Configuration Description

SELinux is enabled for openEuler by default and the default mode is enforcing. You can change the SELinux mode by changing the value of **SELINUX** in **/etc/selinux/config**.

- To disable the SELinux policy, run the following command:

```
SELINUX=disabled
```

- To use the permissive policy, run the following command:

```
SELINUX=permissive
```

### NOTE

When you switch between the disabled mode and the other mode, you need to restart the system for the switch to take effect.

```
# reboot
```

## SELinux Commands

- Query the SELinux mode. For example, the following shows that the SELinux mode is permissive.

```
# getenforce  
Permissive
```

- Set the SELinux mode. **0** indicates the permissive mode, and **1** indicates the enforcing mode. For example, run the following command to set the SELinux mode to enforcing. This command cannot be used to set the disabled mode. After the system is restarted, the mode set in **/etc/selinux/config** is restored.

```
# setenforce 1
```

- Query the SELinux status. **SELinux status** indicates the SELinux status. **enabled** indicates that SELinux is enabled, and **disabled** indicates that SELinux is disabled. **Current mode** indicates the current security policy of the SELinux.

```
# sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:       /etc/selinux
Loaded policy name:            targeted
Current mode:                  enforcing
Mode from config file:         enforcing
Policy MLS status:             enabled
Policy deny unknown status:    allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     31
```

---

# 5 Appendix

---

This chapter describes the file permissions and **umask** values.

## 5.1 Permissions on Files and Directories

### 5.2 umask Values

## 5.1 Permissions on Files and Directories

Permission on files and directories in Linux specifies the users who can access and perform operations on files and directories and the access and operation modes. Permissions on files and directories include read only, write only, and execute.

The following types of users can access files and directories:

- File creator
- Users in the same group as a file creator
- Users not in the same group as a file creator

An example of permission on files and directories is described as follows:

If the permission on **/usr/src** is set to **755** which is 111101101 in binary mode, permissions for each type of users are described as follows:

- The left-most **111** indicates that the file owner can read, write, and execute the file.
- The middle **101** indicates the group users can read and execute but cannot write the file.
- The right-most **101** indicates that other users can read and execute but cannot write the file.

## 5.2 umask Values

When a user creates a file or directory, the file or directory has a default permission. The default permission is specified by the **umask** value.

The **umask** value is the complement of the permission value. The actual permission value is obtained by subtracting the **umask** value from the default maximum permission value. The default maximum permission of a file is readable and writable. The default maximum permission of a directory is readable, writable, and executable. The default permission of a

file is 666 minus the **umask** value. The default permission of a directory is 777 minus the **umask** value.