EulerOS V2.0SP5 Administrators Guide

Issue 01

Date 2019-08-14

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

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

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

Huawei Technologies Co., Ltd.

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

Website: http://www.huawei.com

Email: support@huawei.com

Contents

1 Preface

Overview

Huawei EulerOS V2.0 is the Huawei Enterprise Linux server operating system. It is easy to maintain, compatible with mainstream software and hardware, and exhibits high performance, reliability, and security.

This guide provides guidance for users who first use Huawei EulerOS V2.0 to complete routine maintenance and relevant configurations. This guide covers the basic configuration, software package management, and user management.

The EulerOS V2.0 is short for the OS software V2.0 of Huawei EulerOS server. The name EulerOS V2.0 is used in this guide and interfaces of software.

Intended Audience

This guide is intended for EulerOS V2.0 users with a basic understanding of Linux system management, and is also recommended for administrators, system engineers, and maintenance personnel.

This guide assumes that you have a basic understanding of Linux system management.

Symbol Conventions

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

Symbol	Description
⚠ DANGER	Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.
⚠ WARNING	Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.
⚠ CAUTION	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

Symbol	Description
⚠ NOTICE	Indicates a potentially hazardous situation that, 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	Calls attention to important information, best practices, and tips. NOTE is used to address information not related to personal injury, equipment damage, and environment deterioration.

2 Basic Configuration

2.1 Using Commands

2.1.1 Setting the System Locale

System locale settings are stored in the /etc/locale.conf file and can be modified by the localectl command. These settings are read at early boot by the systemd daemon.

Displaying the Current Locale Status

To display the current locale status, run the following command:

```
localectl status
```

Example command output:

```
$ localectl status
   System Locale: LANG=zh_CN.UTF-8
       VC Keymap: cn
      X11 Layout: cn
```

Listing Available Locales

To list available locales, run the following command:

```
localectl list-locales
```

You can check that by listing all Chinese locales with the following command:

```
$ localectl list-locales | grep zh
zh_CN
zh_CN.gb18030
zh_CN.gb2312
zh_CN.gbk
zh_CN.utf8
zh_HK
zh_HK.big5hkscs
zh_HK.utf8
```

```
zh_SG
zh_SG.gb2312
zh_SG.gbk
zh_SG.utf8
zh_TW
zh_TW.big5
zh_TW.euctw
zh_TW.utf8
```

Setting the Locale

To set the locale, run the following command as the root user:

```
localectl set-locale LANG=locale
```

For example, if you want to use Simplified Chinese as the locale, run the following command as the root user:

```
# localectl set-locale LANG=zh_CN.utf8
```

2.1.2 Setting the Keyboard Layout

Keyboard layout settings are stored in the /etc/locale.conf file and can be modified by the localectl command. These settings are read at early boot by the systemd daemon.

Displaying the Current Settings

To display the current keyboard layout settings, run the following command:

```
localectl status
```

Example command output:

```
$ localectl status
   System Locale: LANG=zh_CN.UTF-8
       VC Keymap: cn
      X11 Layout: cn
```

Listing Available Keyboard Layouts

To list all available keyboard layouts that can be configured on EulerOS, run the following command:

```
localectl list-keymaps
```

To list keyboard layouts compatible with your current locale (for example, Chinese), run the following command:

```
$ localectl list-keymaps | grep cn
cn
```

Setting the Keyboard Layout

To set the keyboard layout, run the following command as the root user:

```
localectl set-keymap map
```

The keyboard layout will be equally applied to graphical user interfaces.

Then you can verify if your setting was successful by checking the current status:

```
$ localectl status
   System Locale: LANG=zh_CN.UTF-8
```

```
      VC Keymap: cn
     X11 Layout: us
```

2.1.3 Setting the Date and Time

This topic describes how to set the system date, time, and time zone by using timedatectl, date, and hwclock commands.

2.1.3.1 Using the timedatectl Command

Displaying the Current Date and Time

To display the current date and time, run the following command:

```
timedatectl
```

Example command output:

```
$ timedatectl
Local time: 2015-08-14 15:57:24 CST
Universal time: 2015-08-14 07:57:24 UTC
RTC time: 2015-08-14 07:57:24
        Timezone: Asia/Shanghai (CST, +0800)
     NTP enabled: yes
NTP synchronized: no
 RTC in local TZ: no
      DST active: n/a
```

Changing the Current Time

To change the current time, run the following command as the root user:

```
timedatectl set-time HH:MM:SS
```

For example, to change the current time to 15:57:24 pm, run the following command as the root user:

```
# timedatectl set-time 15:57:24
```

Changing the Current Date

To change the current date, run the following command as the root user:

```
timedatectl set-time YYYY-MM-DD
```

For example, to change the current date to 14 August 2015, run the following command as the root user:

```
# timedatectl set-time '2015-08-14'
```

Changing the Time Zone

To list all available time zones, run the following command:

```
timedatectl list-timezones
```

To change the current time zone, run the following command as the root user:

```
timedatectl set-timezone time_zone
```

Imagine you want to identify which time zone is closest to your present location while you are in Asia. You can check that by listing all available time zones in Asia with the following command:

```
# timedatectl list-timezones | grep Asia
Asia/Aden
Asia/Almaty
Asia/Amman
Asia/Anadyr
Asia/Aqtau
Asia/Aqtobe
Asia/Ashgabat
Asia/Baghdad
Asia/Bahrain
......

Asia/Seoul
Asia/Shanghai
Asia/Singapore
Asia/Srednekolymsk
Asia/Taipei
Asia/Tashkent
Asia/Tbilisi
Asia/Tehran
Asia/Thimphu
Asia/Tokyo
```

To change the time zone to shanghai, run the following command:

```
# timedatectl set-timezone Asia/Shanghai
```

Synchronizing the System Clock with a Remote Server

Your system clock can be automatically synchronized with a remote server using the Network Time Protocol (NTP). To enable or disable this feature, run the following command as the root user:

```
timedatectl set-ntp boolean
```

For example, to automatic synchronization of the system clock with a remote server, run the following command:

```
# timedatectl set-ntp yes
```

2.1.3.2 Using the date Command

Displaying the Current Date and Time

To display the current date and time, run the following command:

```
date
```

By default, the date command displays the local time. To display the time in Coordinated Universal Time (UTC), run the command with the --utc or -u command line option:

```
date --utc
```

You can also customize the format of the displayed information by providing the + "format" option on the command line:

```
date +"format"
```

Table 2-1 Formatting options

Format Option	Description
%H	The hour in the HH format (for example, 17)
%M	The minute in the MM format (for example, 37)
%S	The second in the SS format (for example, 25)
%d	The day of the month in the DD format (for example, 15)
%m	The month in the MM format (for example, 07)
%Y	The year in the YYYY format (for example, 2015)
%Z	The time zone abbreviation (for example, CEST)
%F	The full date in the YYYY-MM-DD format (for example, 2015-7-15). This option is equal to %Y-%m-%d.
%T	The full time in the HH:MM:SS format (for example, 18:30:25). This option is equal to %H:%M:%S.

Example commands and outputs:

- To display the current date and time:
```
$ date
2015 08 17 Monday 17:26:34 CST
```
- To display the current date and time in UTC:
```
$ date --utc
2015 08 17 Monday 09:26:18 UTC
```
- To customize the output of the date command:
```
$ date +"%Y-%m-%d %H:%M"
2015-08-17 17:24
```

Changing the Current Time

To change the current time, run the date command with the --set or -s option as the root user:

```
date --set HH:MM:SS
```

By default, the date command sets the local time. To set the system clock in UTC instead, run the command with the --utc or -u command line option:

```
date --set HH:MM:SS --utc
```

For example, to change the current time to 23:26:00 p.m, run the following command as the root user:

```
# date --set 23:26:00
```

Changing the Current Date

To change the current date, run the following command with the **--set** or **-s** option as the **root** user:

```
date --set YYYY-MM-DD
```

For example, to change the current date to 2 November 2015, run the following command as the root user:

```
# date --set 2015-11-02
```

2.1.3.3 Using the hwclock Command

The hwclock command is used to set the real-time clock (RTC).

Real-Time Clock and System Clock

Linux distinguishes between the system clock and the real-time clock. The system clock is maintained by Linux kernel, whereas the real-time clock is an integrated clock on the system board and is battery-powered. The real-time clock is defined in the Standard BIOS Feature option of BIOS.

When Linux starts, it reads the real-time clock time and sets the system clock time based on the real-time clock time.

Displaying the Current Date and Time

To display the current RTC date and time, run the following command as the root user:

```
hwclock
```

Example command output:

```
# hwclock
2015-08-17, Monday, 14:34:42, -0.094973s
```

Setting the Date and Time

To change the RTC date and time, run the following command as the root user:

```
hwclock --set --date "dd mmm yyyy HH:MM"
```

For example, to change the RTC time to 21:17, 21 October 2015, run the following command as the root user:

```
# hwclock --set --date "21 Oct 2015 21:17" --utc
```

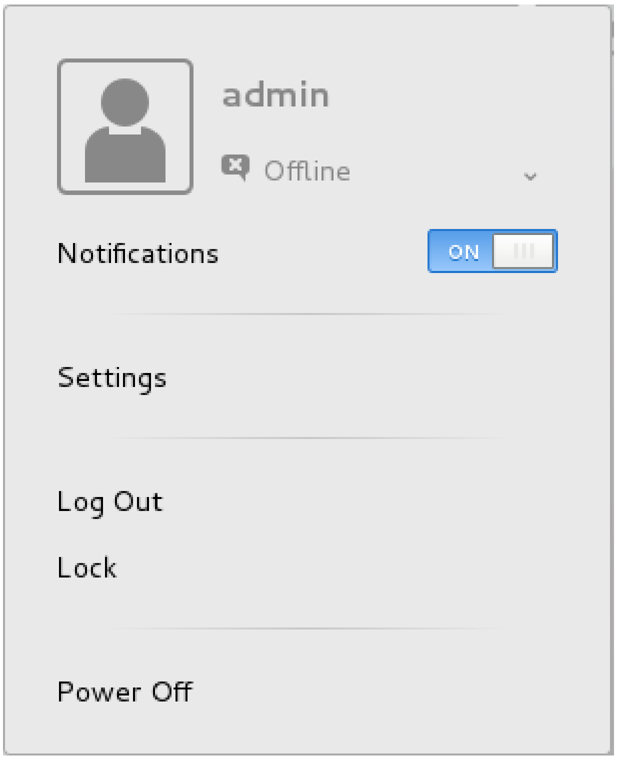
2.2 Using GUI

This topic guides you through configuring basic system options on graphical user interface (GUI).

2.2.1 Opening the Settings Page

Click the user name in the upper right corner of desktop. On the displayed page, click **Settings**, as shown in **Figure 2-1**.

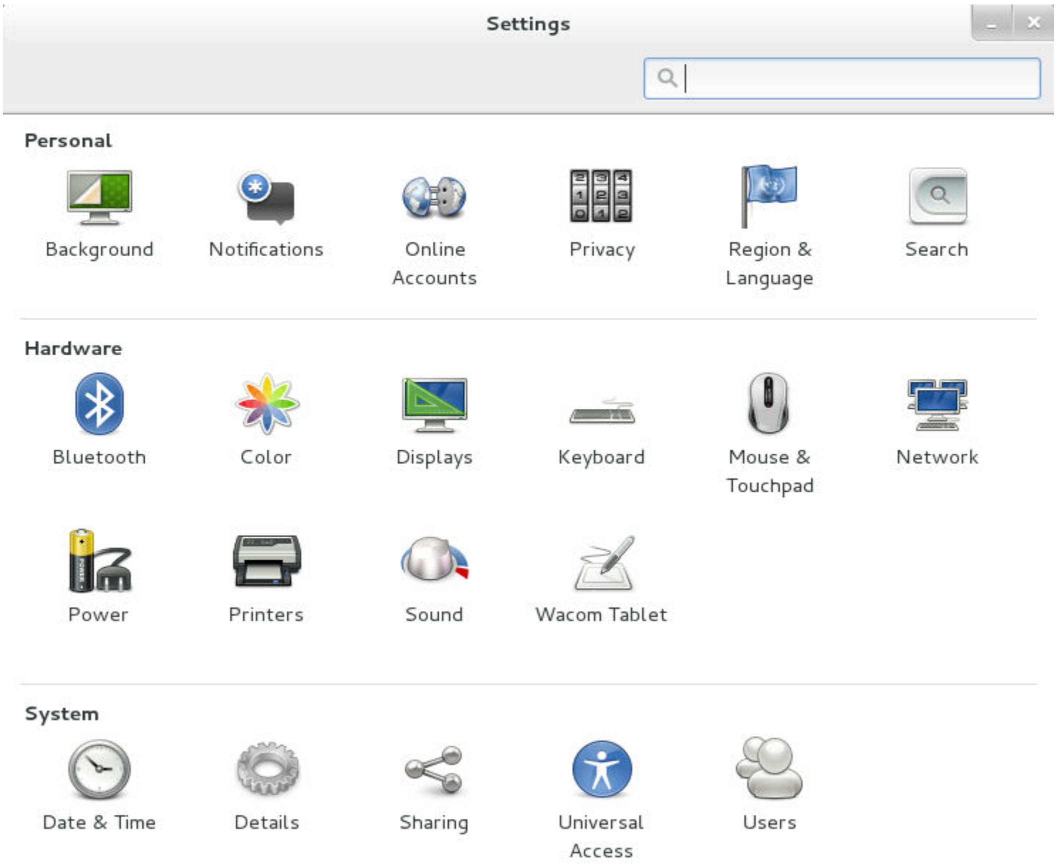
Figure 2-1 Opening the Settings page



The Settings page is displayed, as shown in **Figure 2-2**. You can set items, such as background, language, and network.

EulerOS V2.0SP5 Administrators Guide

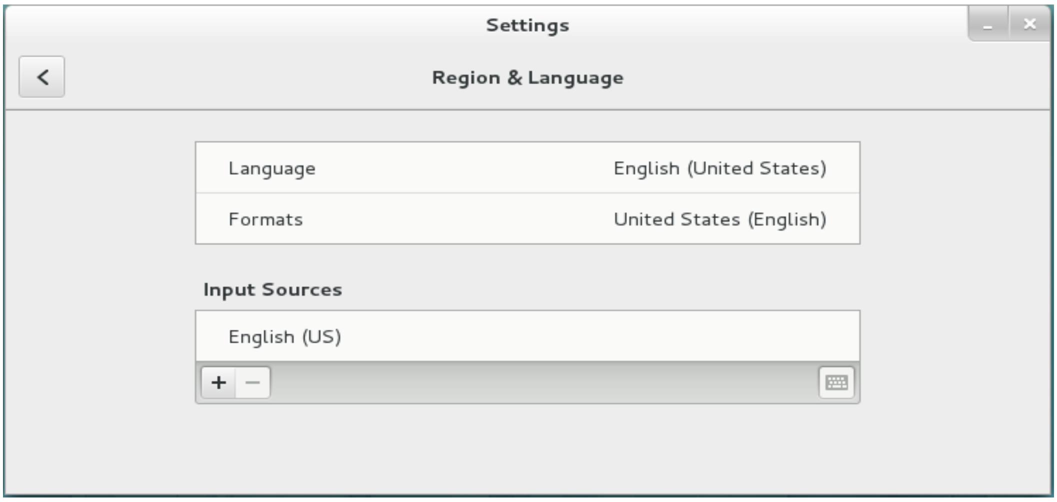
Figure 2-2 Settings page



2.2.2 Setting the Language

Click **Region & Language** in the **Personal** area of the **Settings page**. The **Region & Language** page is displayed, as shown in **Figure 2-3**. You can set the language, formats, and input sources as required.

Figure 2-3 Region & Language page



Language English (United States)

Formats United States (English)

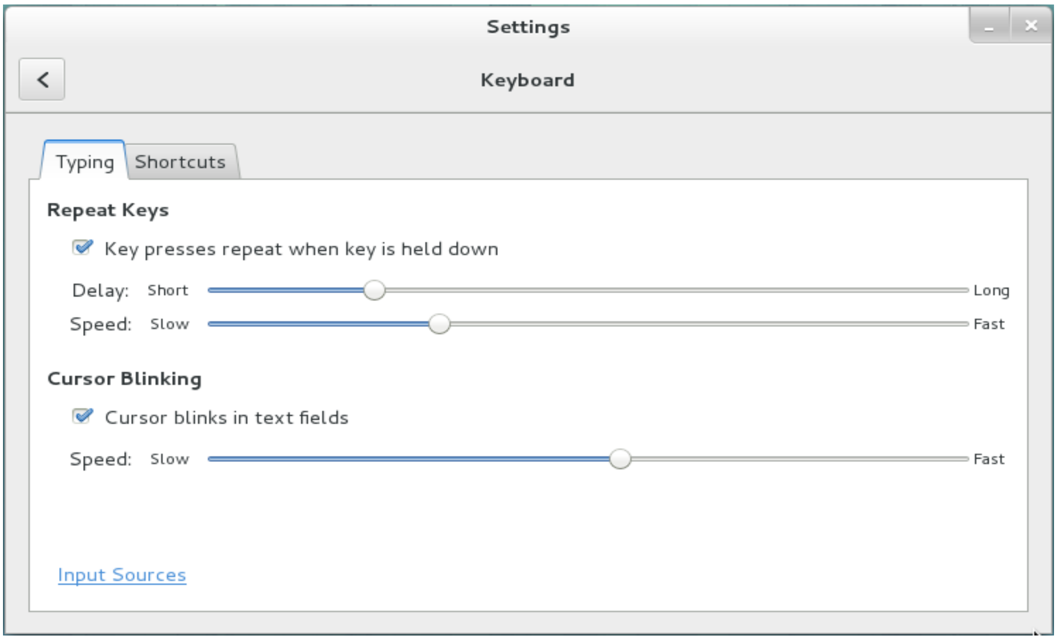
Input Sources

English (US)

2.2.3 Setting the Keyboard Layout

In the Settings page, click **Keyboard**. The **Keyboard** page is displayed, as shown in **Figure 2-4**. You can set the repeat keys, cursor blinking, and shortcuts.

Figure 2-4 Keyboard



Repeat Keys

☑ Key presses repeat when key is held down

Delay: Short — Long

Speed: Slow — Fast

Cursor Blinking

☑ Cursor blinks in text fields

Speed: Slow — Fast

2.2.4 Setting the Date and Time

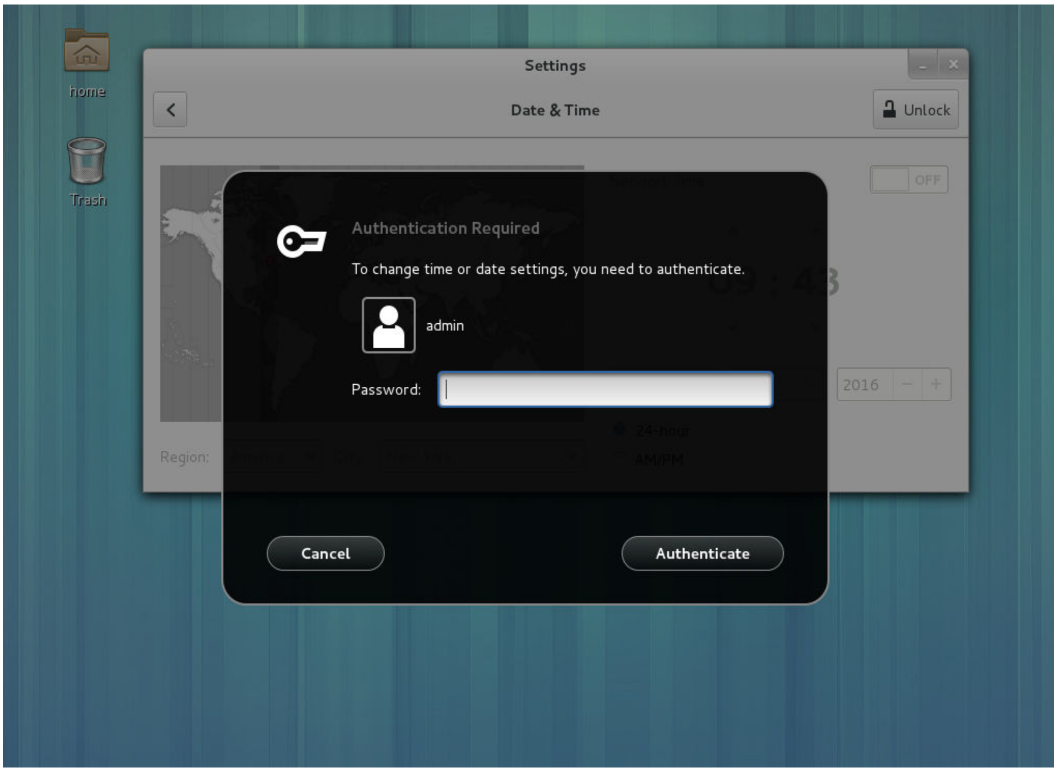
In the Settings page, click **Date & Time**. The **Date & Time** page is displayed.

If you have logged in to EulerOS as a non-root user, before setting the date and time, you must click **Unlock** in the upper right corner of the **Date & Time** page and then enter the admin password, as shown in **Figure 2-5**.

NOTE

If a common user instead of an admin user is created during installation, enter the password of the **root** user.

Figure 2-5 Date & Time page

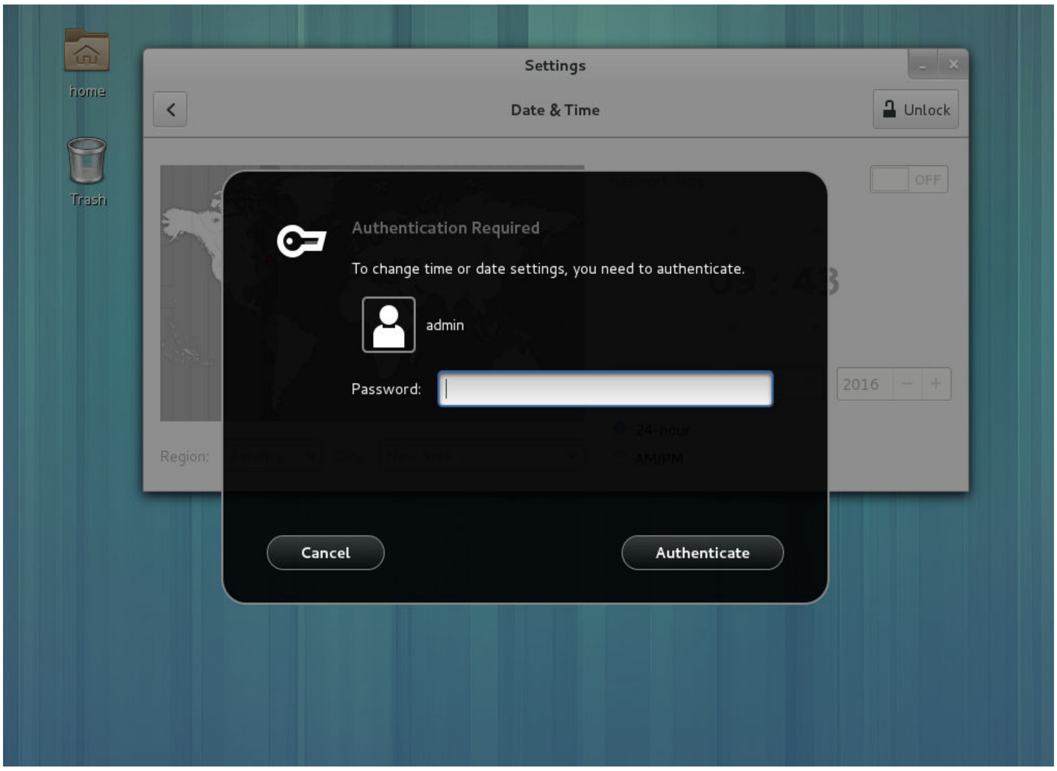


Authentication Required

To change time or date settings, you need to authenticate.

admin

Password:



Authentication Required

To change time or date settings, you need to authenticate.

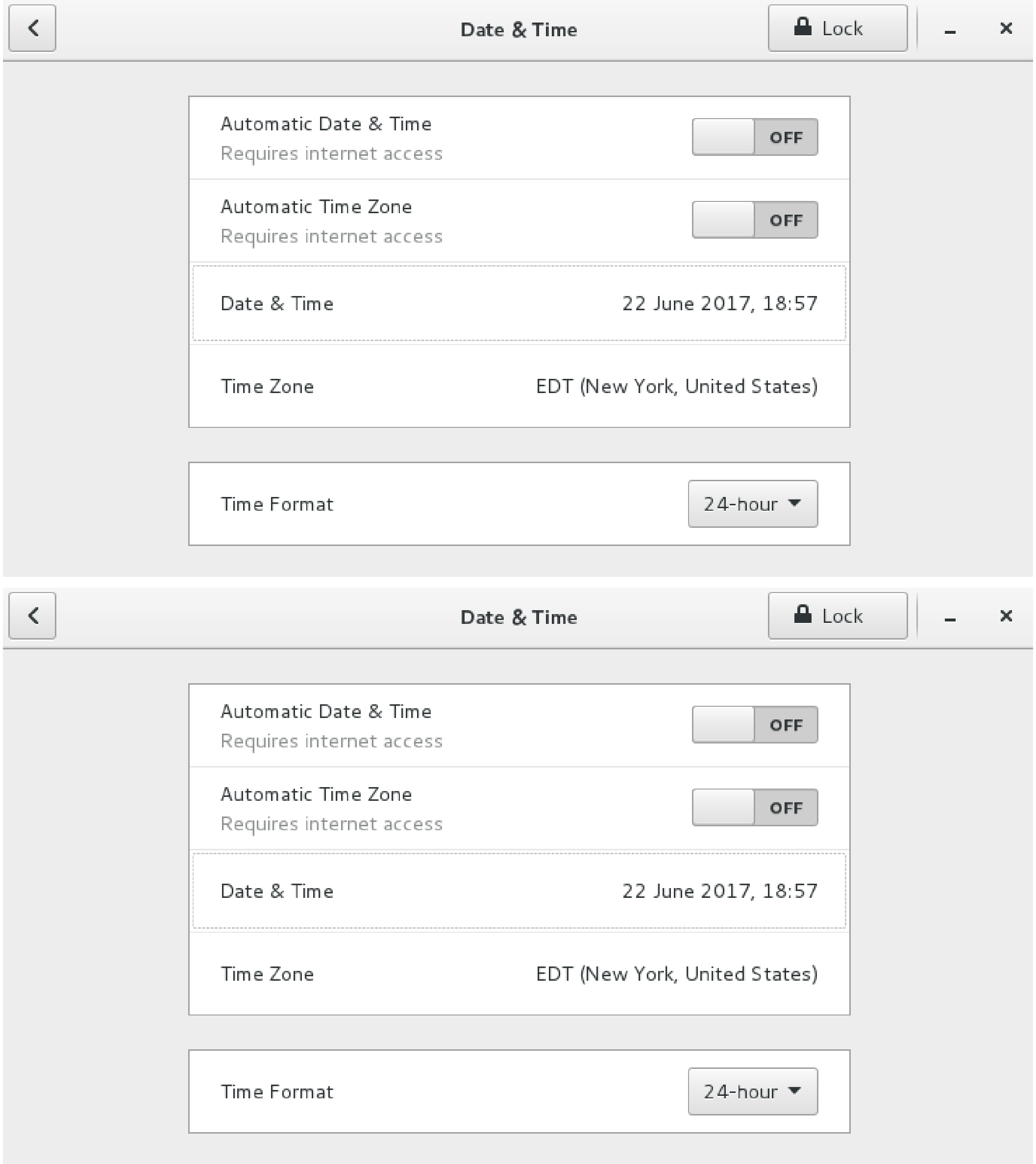
admin

Password:

After the authentication is complete, you can modify the time zone, date, and time, as shown in **Figure 2-6**, **Figure 2-7**, and **Figure 2-8**.

EulerOS V2.0SP5 Administrators Guide

Figure 2-6 Date & Time 1



Automatic Date & Time
Requires internet access
OFF

Automatic Time Zone
Requires internet access
OFF

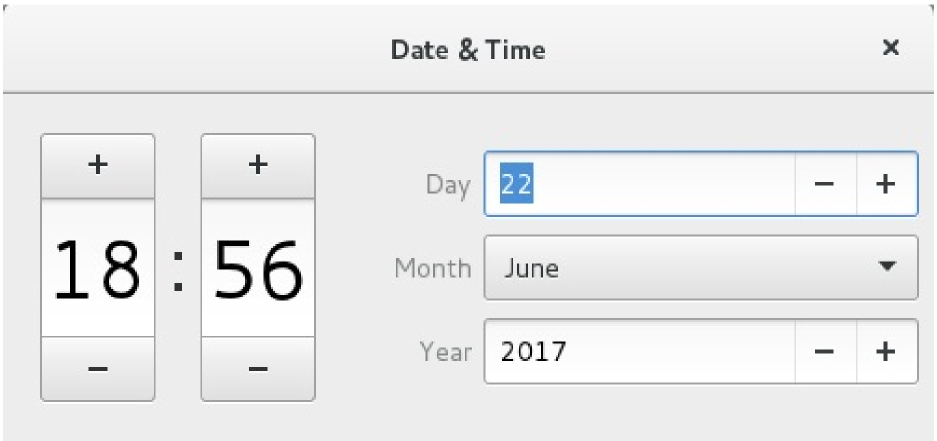
Date & Time 22 June 2017, 18:57

Time Zone EDT (New York, United States)

Time Format 24-hour

Date & Time
Lock
Automatic Date & Time
Requires internet access
OFF
Automatic Time Zone
Requires internet access
OFF
Date & Time 22 June 2017, 18:57
Time Zone EDT (New York, United States)
Time Format 24-hour

Figure 2-7 Date & Time 2

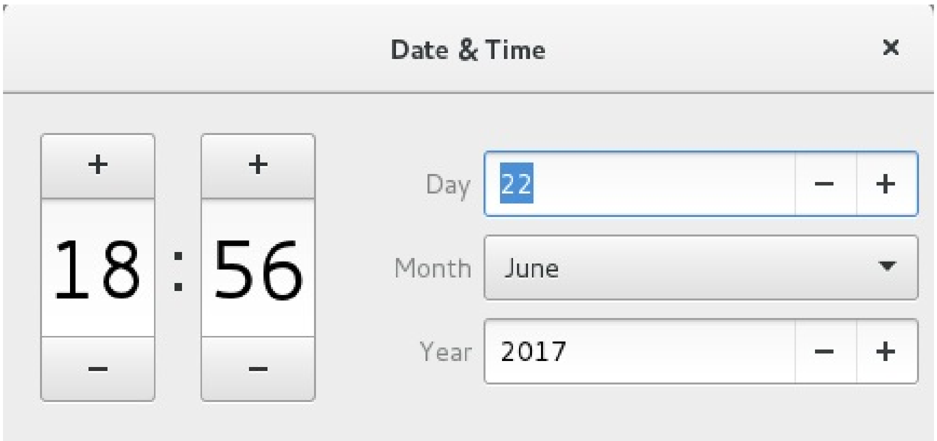


18 : 56

Day 22

Month June

Year 2017



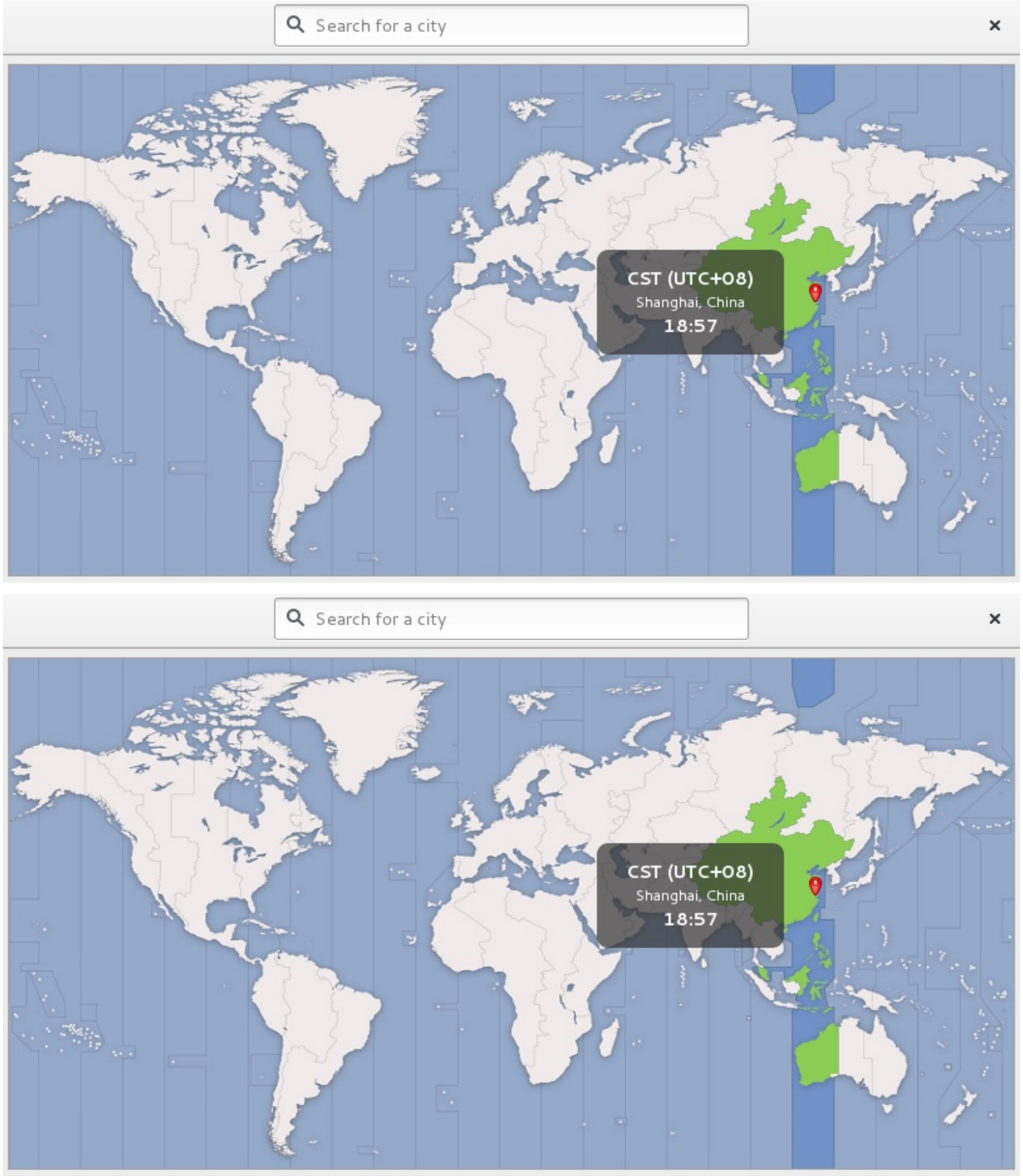
18 : 56

Day 22

Month June

Year 2017

Figure 2-8 Setting the time zone



3 User Management

In Linux, each common user has an account, including the user name, password, and home directory. There also exist special users created for specific purposes, and the most important special user is the admin account whose default user name is root. The concept of user group is introduced to make privilege management easier. Each user belongs to at least one user group.

The control of users and groups is a core element of EulerOS security management. This topic explains how to create multiple admin accounts and assign privileges to common users in graphical user interface and on command lines.

3.1 Adding a User

useradd Command

The useradd command is used to add a new user to EulerOS.

```
useradd [options] user_name
```

User Information Files

The following files contain user account information:

- /etc/passwd: user account information
- /etc/shadow file: user account encryption information
- /etc/group file: group information
- /etc/defaut/useradd: default configurations
- /etc/login.defs: system wide settings

- /etc/skel: default directory that holds initial configuration files

Example

To create a user XXX, run the following command:

```
[root@localhost ~]# useradd XXX
```

NOTE

> If no prompt is displayed, the user XXX is successfully created. After the user XXX is created, run the passwd command to assign a password to the user. A new account without a password will be banned.

To view information about the new user, run the id command:

```
[root@localhost ~]# id user_example
uid=502(user_example)    gid=502(user_example)
```

To change the password of the user_example, run the following command:

```
[root@localhost ~]# passwd user_example
```

Then, type the password and confirm it as prompted:

```
[root@localhost ~]# passwd user_example
Changing password for user user_example.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

3.2 Modifying a User Account

Changing a Password

Common users can change their passwords using the passwd command. Only the admin is allowed to use the passwd usename command to change passwords for other users.

Changing User's Login Shell

Common users can change their login shell using either the chsh or usermod command. Only the admin is allowed to run the chsh usename command to change login shell for other users.

usermod command for changing user's shell:

```
usermod -s [new_shell_path] usename
```

The values of **new_shell_path** and **usename** must be the set.

For example, to change the shell of user_example to csh, run the following command:

```
[root@localhost ~]# usermod -s /bin/csh user_example
```

Changing the Home Directory

```
usermod -d [new_home_directory] usename
```

For example, to change the home directory of user_example to /home/user_example, run the following command:

```
[root@localhost ~]# usermod -d /home/user_example user_example
```

To move the content in the current home directory to a new one, run the usermod command with the -m option:

```
usermod -d /new/home -m usename
```

Changing a UID

```
usermod -u UID usename
```

The usermod command can change a user's UID in all files and directories under the user's home directory. However, for files outside the user's home directory, their owners can only be changed using the chown command.

Changing Account Expiry Date

To change account expiry date, run the following command. A prerequisite for using this command is that a shadow password is in use.

```
usermod -e MM/DD/YY usename
```

3.3 Deleting a User

The userdel command is used to delete a user.

For example, to delete the user Test, run the following command:

```
[root@localhost ~]# userdel Test
```

If you need to also delete the user's home directory and all contents in the directory, run the userdel command with the -r option to delete them recursively.

NOTE

If a user has logged in to EulerOS, the user cannot be deleted unless you have first killed relevant processes.

3.4 Authorizing Administrator Accounts

3.4.1 Creating Multiple Administrator Accounts

Most new system administrators believe that the **root** user is the only administrator account. However, the **root** user is the default system administrator account. Open the **/etc/passwd** file. You will find the following information:

```
root:x:0:0:root: /root:/bin/bash
bin:x:1:1:bin: /bin: /sbin/nologin
daemon:x:2:2:daemon: /sbin: /sbin/nologin
adm:x:3:4:acim: /var/adm:/sbin/nologin
lp:x:4:7:lp: /va r/spool/ lpd: /sbin/nologin
sync:x:5:0:sync: /sbin: /bin/sync
shutdown:x:6:0:shutdown: /sbin: /sbin/shutdown
Test:x:0:0::/home/j ianCJzhonghua: /bin/bash
```

The preceding information shows that the UID and GID of the **root** user are both 0. This is the necessary and sufficient condition for the administrator account, that is, only if the user UID and GID are 0, the user functions as the **root** user. In the preceding example, **Test** is also an administrator account.

3.4.2 Granting Rights to a Common User

The **sudo** command allows common users to execute commands that can be executed only by administrator accounts.

The **sudo** command allows the user specified in the **/etc/sudoers** file to execute the administrator account commands. For example, an authorized common user can run:

```
sudo /usr/sbin/useradd newuser1
```

The **sudo** command can specify a common user that has been added to the **/etc/sudoers** file to process tasks as required.

The information configured in the **/etc/sudoers** file is as follows:

- Blank lines or comment lines starting with #: Have no specific functions.
- Optional host alias lines: Create the name of a host list. The lines must start with **Host_Alias**. The host names in the list must be separated by commas (,). For example:

  ```
  Host_Alias  linux=ted1,ted2
  ```

  ted1 and **ted2** are two host names, which can be called **linux**.
- Optional user alias lines: Create the name of a user list. The lines must start with **User_Alias**. The user names in the list must be separated by commas (,). The user alias lines have the same format as the host alias lines.
- Optional command alias lines: Create the name of a command list. The lines must start with **Cmnd_Alias**. The commands in the list must be separated by commas (,).
- Optional running mode alias lines: Create the name of a user list. The difference is that such alias can enable a user in the list to run the **sudo** command.
- Necessary declaration lines for user access:

  The declaration syntax for user access is as follows:

  ```
  user host = [ run as user ] command list
  ```

  Set the user to a real user name or a defined user alias, and set the host to a real host name or a defined host alias. By default, all the commands executed by sudo are executed as user **root**. If you want to use another account, you can specify it. **command list** is either a command list separated by commas (,) or a defined command alias. For example:

  ```
  ted1   ted2=/sbin/shutdown
  ```

  In this example, ted1 can run the shutdown command on ted2.

  NOTE

  You can define multiple aliases in a line and separate them with colons (:).

  You can add an exclamation mark (!) before a command or a command alias to make the command or the command alias invalid.

  There are two keywords: ALL and NOPASSWD. ALL indicates all files, hosts, or commands, and NOPASSWD indicates that no password is required.

The following is an example of the **sudoers** file:

```
#sudoers files
#User alias specification
User_Alias ADMIN=ted1:POWERUSER=globus,ted2
#user privilege specification
ADMIN ALL=ALL
POWERUSER AIL=ALL,!/bin/su
```

EulerOS V2.0SP5 Administrators Guide

The third line defines two aliases ADMIN and POWERUSER. The fifth line indicates that ADMIN can run all commands on all hosts as user **root**. The sixth line gives the same permission as ADMIN to POWERUSER except running the **su** command.

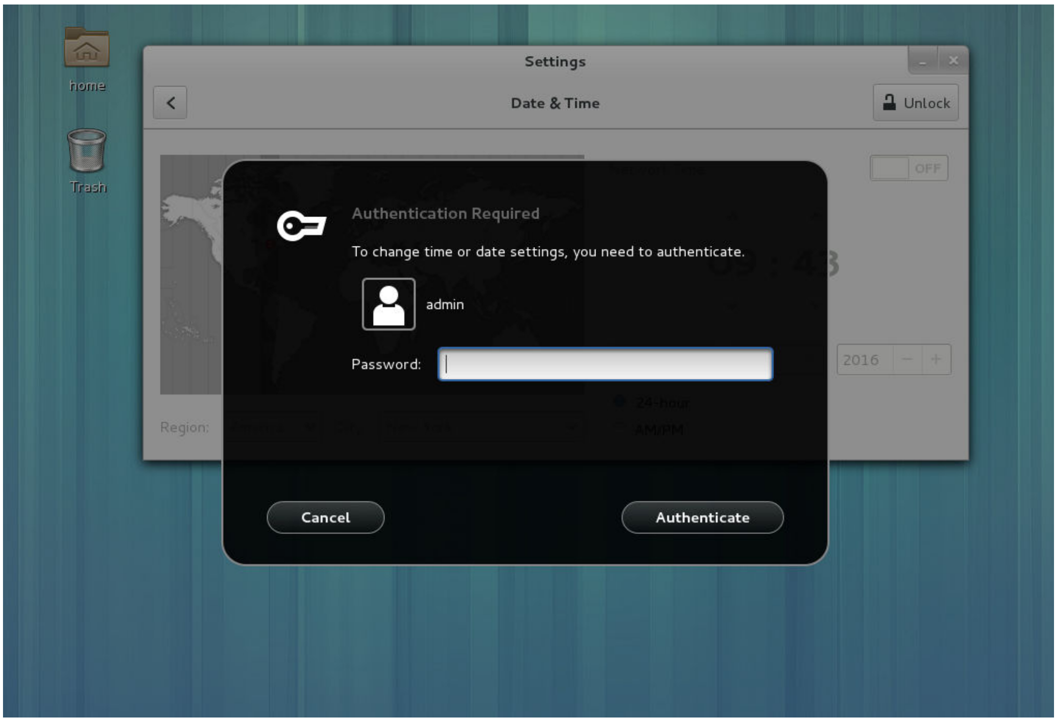
3.5 Using GUI

In the **Figure 2-2** page, click **Users**. The **Users** page is displayed.

If you have logged in to EulerOS as a non-root user, before adding or deleting a user, you must click **Unlock** in the upper right corner of the **Users** page and then type the admin password (or the root user password if no admin user is created at the time of OS installation), as shown in **Figure 3-1**.

NOTE

Figure 3-1 Authentication



Authentication Required

To change time or date settings, you need to authenticate.

admin

Password:

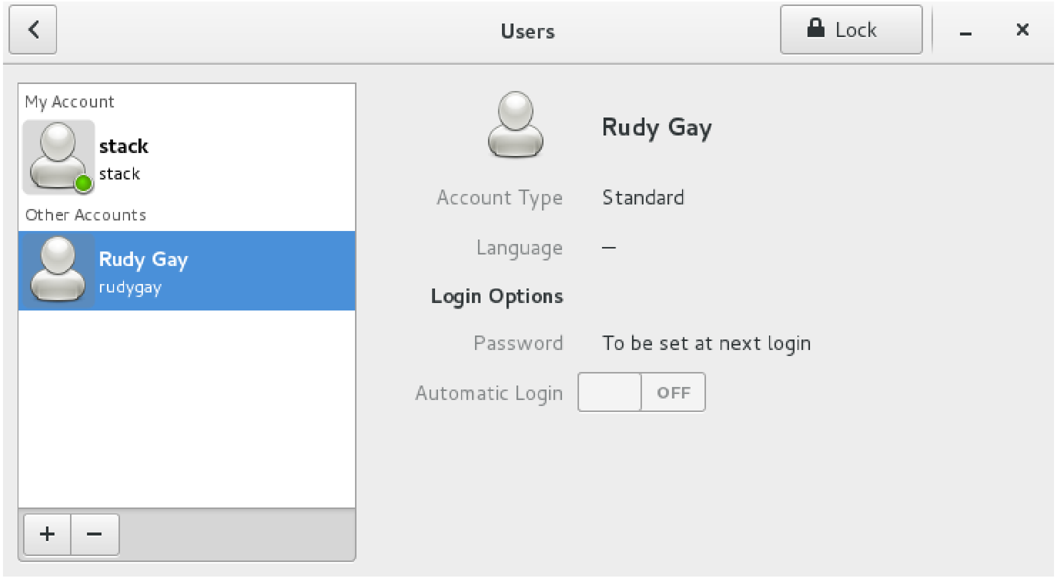
After the authentication is passed, click + or **-** in the lower left corner of the **Users** page to add or delete users.

3.5.1 Adding a User

Click + (a non-root user needs to be authenticated) in the lower left corner of **Figure 3-2** to add a user, as shown in **Figure 3-3**.

EulerOS V2.0SP5 Administrators Guide

Figure 3-2 Add account page



stack
stack

Rudy Gay
rudygay

Rudy Gay

Account Type Standard

Language —

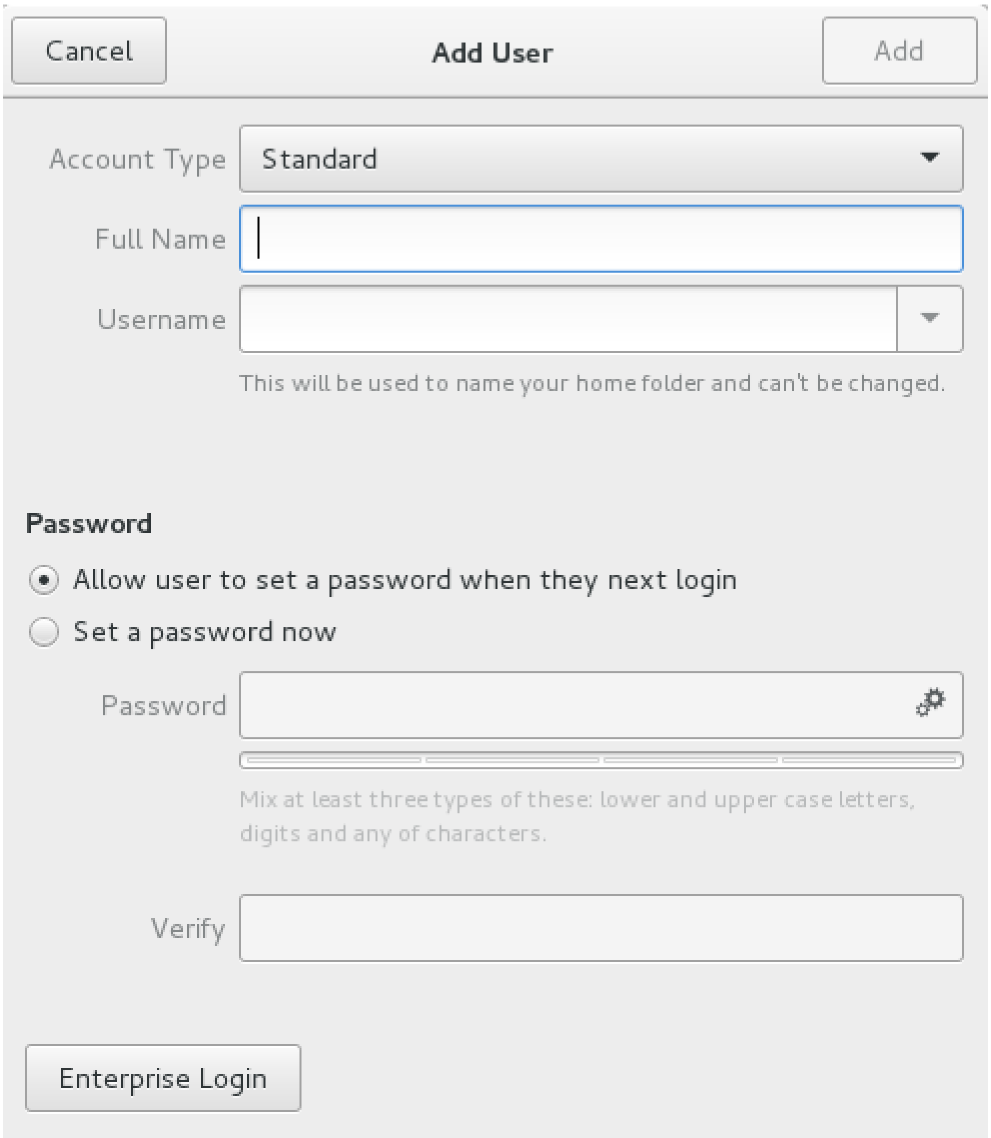
Login Options

Password To be set at next login

Automatic Login OFF

Specify the account type (standard or admin), full name, and user name. Then, click **Add**. Information about the added user is displayed, as shown in **Figure 3-3**.

Figure 3-3 Adding a user



Cancel **Add User** Add

Account Type: Standard

Full Name:

Username:

This will be used to name your home folder and can't be changed.

Password

● Allow user to set a password when they next login

○ Set a password now

Password:

Mix at least three types of these: lower and upper case letters, digits and any of characters.

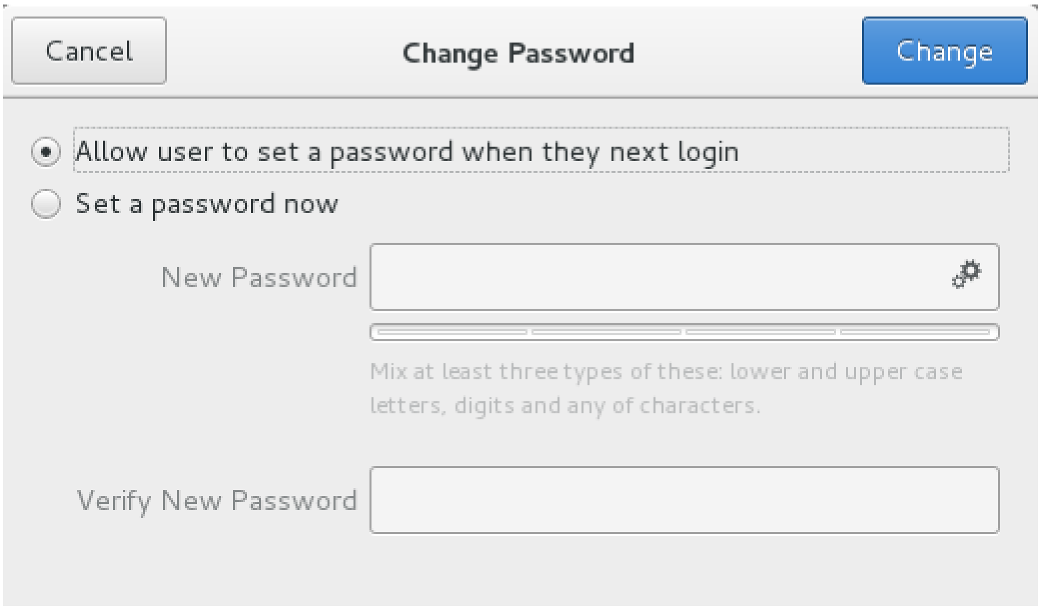
Verify:

Enterprise Login

By default, the newly added user account is disabled and its login options need to be manually specified. Click **To be set at next login** in **Figure 3-2**. In the dialog box that is displayed, you can determine whether to set the password based on site conditions, as shown in **Figure 3-4**.

EulerOS V2.0SP5 Administrators Guide

Figure 3-4 Changing the password for a newly added user



● Allow user to set a password when they next login

○ Set a password now

New Password

Mix at least three types of these: lower and upper case letters, digits and any of characters.

Verify New Password

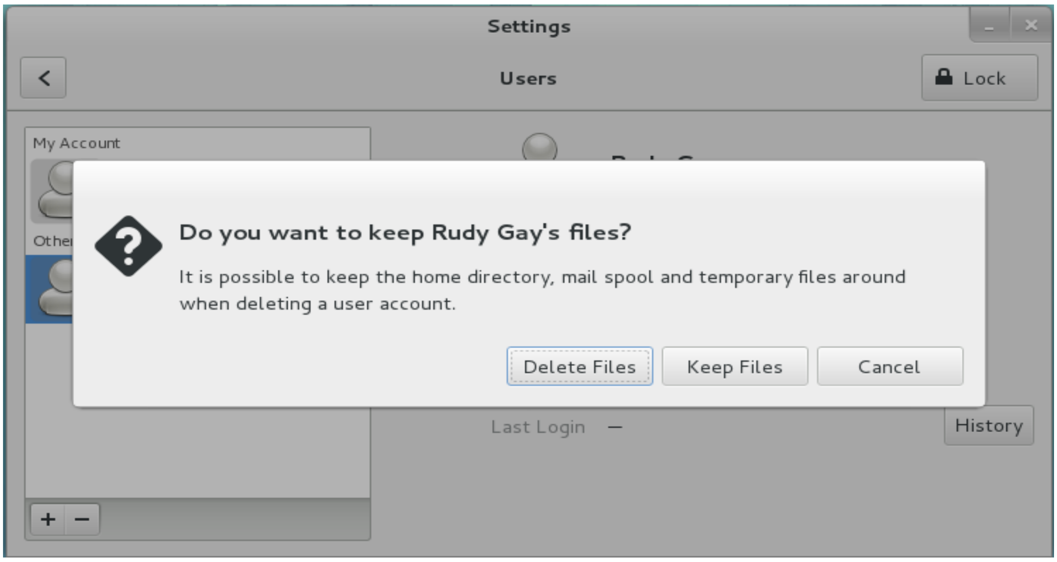
3.5.2 Deleting a User

Select the user you want to delete and click **-** in the lower left corner of the **Users** page. A message (see Figure 3-5) is then displayed, prompting you to confirm whether user files shall be deleted along with the user account.

NOTE

The user who is logging in to EulerOS cannot be deleted. Also, a common user is not allowed to delete the admin user if there is only one admin user.

Figure 3-5 Deleting a user



Do you want to keep Rudy Gay's files?

It is possible to keep the home directory, mail spool and temporary files around when deleting a user account.

4 Software Package Management by Yum

Yum is a software package manager. Based on the RedHat package manager (RPM), Yum is able to automatically download RPM packages from specified servers (repositories) and install them. Yum performs automatic dependency management on packages you are updating, installing, or removing, and thus is able to automatically determine, fetch, and install all available dependent packages.

4.1 Configuring Yum

4.1.1 Modifying the Yum Configuration File

The main configuration file for yum is located at /etc/yum.conf. This file contains one mandatory [main] section, which allows you to set yum options that have global effect, and can also contain one or more [repository] sections, which allow you to set repository-specific options. Individual repositories are defined in .repo files in the /etc/yum.repos.d directory.

Configure Yum either using the yum.conf file under the /etc directory or by adding the .repo file under the /etc/yum.repos.d directory.

Setting the [Main] Section

The following is an example [main] section:

```
[main]
cachedir=/var/cache/yum/$basearch/$releasever
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
obsoletes=1
gpgcheck=1
plugins=1
installonly_limit=3
```

NOTE

For details about the complete [main] section, see yum.conf(5) in the online help of [main].

Common options

Table 4-1 in the [main] section

Option	Description
cachedir	Yum's cache directory where downloaded RPM packages and databases are saved.
keepcache	Determines whether Yum retains the cache of RPM packages and header files after a successful installation. Value: 0 or 1. The default value is **0** (do not retain the cache).
debuglevel	Specifies the debug information generated from Yum. Value range: [0-10]. A larger value indicates more detailed debug information. The default value is **2**. The value is **0**, indicating that no debug information is displayed.
logfile	Indicates the output directory of log files.
exactarch	Determines whether the system architecture is considered when Yum upgrades installed software packages. Values: 1 or 0. The default value is **1**, indicating that the system architecture needs to be considered. If a 32-bit package is installed, the upgrade is not performed by installing the same 64-bit package.
obsoletes	Specifies whether the old RPM packages can be updated. Optional value: 1 or 0. The default value is **1**, indicating that the upgrade is allowed.
gpgcheck	Specifies whether GPG check can be performed. Value: 1 or 0. The default value is **1**, indicating that GPG check is needed.
plugins	Specifies whether to enable or disable Yum plugins. Value: 1 or 0. The default value is **1**, indicating that Yum plugins are enabled.
installonly_limit	Specifies the number of packages that can be synchronously installed and are listed in the **installonlypkgs** command. The default value is **3**. You are not advised to set the value to 2.

Setting the [repository] Sections

The [repository] sections allow you to define individual Yum repositories. Each Yum repository must have a unique name. Otherwise, a conflict between repositories occurs. The following is a bare-minimum example of a [repository] section:

```
[repository]
name=repository_name
baseurl=repository_url
```

Description about options:

Table 4-2 Options in a [repository] section

Option	Description
name=repository_name	A string describing the repository.
baseurl=repository_url	Uniform resource locator (URL) to the directory where the repository is located. ● If the repository is available over Hypertext Transfer Protocol (HTTP), the URL is http://path/to/repo. ● If the repository is available over File Transfer Protocol (FTP), the URL is ftp://path/to/repo. ● If the repository is local to the machine, the URL is file:///path/to/local/repo.

Viewing Current Configurations

To display the current values of global Yum options, run the following command:

```
yum-config-manager
```

To view the configurations of a particular section in the Yum configuration file, run the following command:

```
yum-config-manager section…
```

You can also use a global regular expression to view the configurations of all matching sections.

```
yum-config-manager glob_expression…
```

For example, to view all configuration options and their values in the [repo] section, run the following command:

```
$ yum-config-manager repo \*
================================ repo: EulerOS-base
==================================
[EulerOS-base]
async = True
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/2.0SP5
baseurl = http://developer.huawei.com/ict/site-euleros/euleros/repo/yum/2.2/os/
x86_64/
cache = 0
cachedir = /var/tmp/yum-euler-BafhEh/x86_64/2.0SP5/EulerOS-base
check_config_file_age = True
cost = 1000
deltarpm_metadata_percentage = 100
deltarpm_percentage =
enabled = True
enablegroups = True
exclude =
failovermethod = priority
......
```

4.1.2 Creating a Yum Repository

To create a Yum repository, perform the following steps:

1. Run the following command as the root user to install the createrepo package:
   ```
   yum install createrepo
   ```
2. Copy all packages that you want to have in your repository into one directory, such as /mnt/local_repo/:
3. Then, run the following command to create the necessary metadata for your Yum repository, as well as the sqlite database for speeding up Yum operations:
   ```
   createrepo --database /mnt/local_repo
   ```

4.1.3 Adding, Enabling, and Disabling a Yum Repository

This topic explains how to add, enable, and disable a repository by using the yum-config-manager command.

Adding a Yum Repository

To define a new repository, you can either add a [repository] section to the /etc/yum.conf file, or (recommended) add a .repo file in the //etc/yum.repos.d/ directory. Yum repositories commonly provide their own .repo file.

To add such a repository to your system, run the following command as the root user:

```
yum-config-manager --add-repo repository_url
```

For example, to add a repository located at http://www.example.com/example.repo, run the following command:

```
# yum-config-manager --add-repo http://www.example.com/example.repo
Loaded plugins: langpacks, product-id, subscription-manager
adding repo from: http://www.example.com/example.repo
grabbing file http://www.example.com/example.repo to /etc/yum.repos.d/example.repo
example.repo                                              |  413 B     00:00
repo saved to /etc/yum.repos.d/example.repo
```

Enabling a Yum Repository

To enable a particular repository or repositories, run the following command as the root user:

```
yum-config-manager --enable repository…
```

Alternatively, you can use a global regular expression to enable all matching Yum repositories:

```
yum-config-manager --enable glob_expression…
```

For example, to enable example, example-debuginfo, and example-source repositories, run the following command with a global regular expression:

```
# yum-config-manager --enable example\*
Loaded plugins: langpacks, product-id, subscription-manager
============================== repo: example ==============================
[example]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/6Server
baseurl = http://www.example.com/repo/6Server/x86_64/
cache = 0
cachedir = /var/cache/yum/x86_64/6Server/example
[output truncated]
```

Disabling a Yum Repository

To disable a particular repository or repositories, run the following command as the root user:

```
yum-config-manager --disable repository…
```

Alternatively, you can use a global regular expression to disable all matching Yum repositories:

```
yum-config-manager --disable glob_expression…
```

4.2 Managing Software Packages

Yum allows you to perform a complete set of operations with software packages, including searching for packages, viewing information about them, installing and removing.

Searching for Software Packages

To search for RPM packages by name, abbreviation, or description, run the following command:

```
yum search term…
```

For example:

```
$   yum search httpd
========================================== N/S matched: httpd
==========================================
httpd.x86_64 : Apache HTTP Server
httpd-devel.x86_64 : Development interfaces for the Apache HTTP server
httpd-manual.noarch : Documentation for the Apache HTTP server
httpd-tools.x86_64 : Tools for use with the Apache HTTP Server
libmicrohttpd.i686 : Lightweight library for embedding a webserver in applications
libmicrohttpd.x86_64 : Lightweight library for embedding a webserver in
applications
mod_auth_mellon.x86_64 : A SAML 2.0 authentication module for the Apache Httpd
Server
mod_dav_svn.x86_64 : Apache httpd module for Subversion server
```

Listing Software Packages

To list information on all installed and available RPM packages, run the following command:

```
yum list all
```

To list the installed and available RPM packages that match a particular global regular expression, run the following command:

```
yum list glob_expression…
```

For example:

```
$ yum list httpd
Available Packages
httpd.x86_64              2.4.6-40.4.h1           EulerOS-base
[root@localhost yum.repos.d]#
```

Displaying Software Package Information

To display information about one or more RPM packages, run the following command:

```
yum info package_name…
```

For example, to display information about the abrt package, run the following command:

```
$ yum info httpd
Available Packages
Name        : httpd
Arch        : x86_64
Version     : 2.4.6
Release     : 40.4.h1
Size        : 1.2 M
Repo        : EulerOS-base
Summary     : Apache HTTP Server
URL         : http://httpd.apache.org/
License     : ASL 2.0
Description : The Apache HTTP Server is a powerful, efficient, and extensible
            : web server.
```

Installing RPM Packages

To install a single package and all of its non-installed dependencies, run the following command as the root user:

```
yum install package_name
```

You can also install multiple packages simultaneously by appending their names as arguments. To do so, run the following command as the root user:

```
yum install package_name package_name…
```

For example:

```
yum install httpd-devel.x86_64
```

Downloading Software Packages

At certain point of installation process, you are prompted to confirm the installation with the following message:

```
...
Total size: 1.2 M
Is this ok [y/d/N]:
...
```

If you select the d option, Yum will download the packages without installing them immediately. You can install these packages later in offline mode. By default, the downloaded packages are saved in the /var/cache/yum/$basearch/$releasever/packages/ directory.

Removing Software Packages

To uninstall a particular package and any packages that depend on it, run the following command as the root user:

```
yum remove package_name…
```

For example, to remove the totem package, run the following command:

```
yum remove totem
```

4.3 Managing Software Package Groups

A package group is a collection of packages that serve a common purpose, for instance, System Tools. With Yum, you can perform an operation on a group of software packages simultaneously, saving time considerably.

Listing Software Package Groups

To view the number of installed software package groups, available groups, and available environment groups with the **summary** parameter, run the following command:

```
yum groups summary
```

Example command output:

```
$ yum groups summary
There is no installed groups file.
Maybe run: yum groups mark convert (see man yum)
EulerOS-base                                | 4.2 kB     00:00
(1/3): EulerOS-base/updateinfo              |  10 kB   00:00
(2/3): EulerOS-base/group_gz                |  15 kB   00:00
(3/3): EulerOS-base/primary_db              | 5.5 MB   00:07
Available environment groups: 4
Available Groups: 4
Done
```

To list all package groups and their group IDs, run the following command with the list ids option:

```
yum group list ids
```

Example command output:

```
$ yum group list ids
Available environment groups:
   Base System (base-sys)
   Developer Mode (developer-mode)
   Cloud Server (cloud-server)
   Server with GUI (graphical-server-environment)
Installed groups:
   Development Tools (development)
Available Groups:
   Compatibility Libraries (compat-libraries)
   Security Tools (security-tools)
   Smart Card Support (smart-card)
Done
```

Displaying Software Package Group Information

To list mandatory and optional packages contained in a particular group, run the following command:

```
yum group info glob_expression…
```

The following is an example of showing information about the **Development Tools** group:

```
$  yum group info "Development Tools"
There is no installed groups file.
Maybe run: yum groups mark convert (see man yum)

Group: Development Tools
 Group-Id: development
 Description: A basic development environment.
 Mandatory Packages:
    autoconf
    automake
    binutils
   +bison
   +flex
    gcc
   +gcc-c++
   +gcc-go
    gettext
    libtool
    make
   +patch
    pkgconfig
   +redhat-rpm-config
   +rpm-build
   +rpm-sign
 Default Packages:
   +byacc
   +cscope
   +ctags
   +diffstat
   +doxygen
    elfutils
   +gcc-gfortran
    git
   +indent
   +intltool
   +patchutils
   +rcs
   +subversion
   +swig
   +systemtap
 Optional Packages:
   ElectricFence
   ant
   babel
   bzr
   chrpath
   cmake
   compat-gcc-44
   compat-gcc-44-c++
   cvs
   dejagnu
   expect
   gcc-gnat
   gcc-objc
   gcc-objc++
   imake
   javapackages-tools
   libstdc++-docs
   mercurial
   mod_dav_svn
   nasm
   perltidy
   python-docs
   rpmdevtools
   rpmlint
   systemtap-sdt-devel
   systemtap-server
```

Installing a Software Package Group

Each software package group has a name and a groupid. You can install a package group by passing its group name or groupid to the group install command.

To install a software package group, run the following command as the root user:

```
yum group install group_name
yum group install groupid
```

For example, to install the **Development Tools** software package group, run the following commands:

```
# yum group install "Development Tools"
# yum group install development
```

Removing a Software Package Group

To remove a software package group, run the yum group remove command with either its group name or groupid as the root user:

```
yum group remove group_name
yum group remove groupid
```

For example, to delete the **Development Tools** software package group, run the following commands:

```
# yum group remove "Development Tools"
# yum group remove development
```

4.4 Checking for and Updating Software Packages

Yum checks whether your system has any updates that wait to be applied. You can list the software packages that need to be updated and update them as a whole, or you can update a selected individual package.

Checking for Updates

To see which installed packages on your system have updates available, run the following command:

```
yum check-update
```

Example command output:

```
# yum check-update
base                                        | 2.9 kB     00:00
updates                                     | 2.9 kB     00:00
(1/2): updates/primary_db                   | 2.9 MB   00:00
(2/2): base/primary_db                      | 4.4 MB   00:00

anaconda-core.x86_64        19.31.123-1.14             updates
anaconda-gui.x86_64         19.31.123-1.14             updates
anaconda-tui.x86_64         19.31.123-1.14             updates
anaconda-user-help.x86_64   19.31.123-1.14             updates
anaconda-widgets.x86_64     19.31.123-1.14             updates
bind-libs.x86_64            32:9.9.4-29.3              updates
bind-libs-lite.x86_64       32:9.9.4-29.3              updates
bind-license.noarch         32:9.9.4-29.3              updates
bind-utils.x86_64           32:9.9.4-29.3              updates
euleros-config.x86_64       1.0-6                      updates
......
```

Updating Software Packages

To update a single package, run the following command as the root user:

```
yum update package_name
```

For example, to update the rpm package, run the yum update rpm command.

```
# yum update anaconda-gui.x86_64
Resolving Dependencies
--> Running transaction check
---> Package anaconda-gui.x86_64 0:19.31.123-1.13 will be updated
---> Package anaconda-gui.x86_64 0:19.31.123-1.14 will be an update
--> Processing Dependency: anaconda-widgets = 19.31.123-1.14 for package:
anaconda-gui-19.31.123-1.14.x86_64
--> Processing Dependency: anaconda-user-help = 19.31.123-1.14 for package:
```

```
anaconda-gui-19.31.123-1.14.x86_64
--> Processing Dependency: anaconda-core = 19.31.123-1.14 for package: anaconda-
gui-19.31.123-1.14.x86_64
--> Running transaction check
---> Package anaconda-core.x86_64 0:19.31.123-1.13 will be updated
--> Processing Dependency: anaconda-core = 19.31.123-1.13 for package: anaconda-
tui-19.31.123-1.13.x86_64
---> Package anaconda-core.x86_64 0:19.31.123-1.14 will be an update
---> Package anaconda-user-help.x86_64 0:19.31.123-1.13 will be updated
---> Package anaconda-user-help.x86_64 0:19.31.123-1.14 will be an update
---> Package anaconda-widgets.x86_64 0:19.31.123-1.13 will be updated
---> Package anaconda-widgets.x86_64 0:19.31.123-1.14 will be an update
--> Running transaction check
---> Package anaconda-tui.x86_64 0:19.31.123-1.13 will be updated
---> Package anaconda-tui.x86_64 0:19.31.123-1.14 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                  Arch         Version              Repository     Size
================================================================================
Updating:
 anaconda-gui             x86_64       19.31.123-1.14       updates       461 k
Updating for dependencies:
 anaconda-core            x86_64       19.31.123-1.14       updates       1.4 M
 anaconda-tui             x86_64       19.31.123-1.14       updates       274 k
 anaconda-user-help       x86_64       19.31.123-1.14       updates       315 k
 anaconda-widgets         x86_64       19.31.123-1.14       updates       748 k

Transaction Summary
================================================================================
Upgrade  1 Package (+4 Dependent packages)

Total download size: 3.1 M
Is this ok [y/d/N]:
```

Similarly, it is possible to update a software package group by running the following command as the root user:

```
yum group update group_name
```

Updating All Software Packages and Their Dependencies

To update all software packages and their dependencies, run the following command as the root user:

```
yum update
```

EulerOS V2.0SP5 Administrators Guide

5 Service Management

This topic describes how to manage your operating system and services using the systemd.

5.1 Introduction to systemd

The systemd is a system and service manager for Linux operating systems. It is designed to be backward compatible with SysV and LSB init scripts, and provides a number of features such as Socket & D-Bus based activation of services, on-demand activation of daemons, system state snapshots, and mount & automount point management. With systemd, the service control logic and parallelization are refined.

Systemd Units

In systemd, the targets of most actions are units, which are resources systemd know how to manage. Units are categorized by the type of resources they represent and defined in unit configuration files. For example, the avahi.service unit represents the Avahi daemon and is defined in the **avahi.service** file. **Table 5-1** lists available types of systemd units.

Table 5-1 Available types of systemd units

Unit Type	File Extension	Description
Service unit	.service	A system service.
Target unit	.target	A group of systemd units.
Automount unit	.automount	A file system automount point.
Device unit	.device	A device file recognized by the kernel.

Unit Type	File Extension	Description
Mount unit	.mount	A file system mount point.
Path unit	.path	A file or directory in a file system.
Scope unit	.scope	An externally created process.
Slice unit	.slice	A group of hierarchically organized units that manage system processes.
Snapshot unit	.snapshot	A saved state of the systemd manager.
Socket unit	.socket	An inter-process communication socket.
Swap unit	.swap	A swap device or a swap file.
Timer unit	.timer	A systemd timer.

All available types of systemd units are located in one of the following directories listed in **Table 5-2**.

Table 5-2 Locations of available systemd units

Directory	Description
/usr/lib/systemd/system/	Systemd units distributed with installed RPM packages.
/run/systemd/system/	Systemd units created at runtime.
/etc/systemd/system/	Systemd units created and managed by the system administrator.

5.2 Features

Fast Activation

The systemd provides more aggressive parallelization than UpStart. The use of Socket- and D-Bus based activation reduces the time required to boot the operating system.

To accelerate system boot, systemd seeks to:

- Activate only the necessary processes
- Activate as many processes as possible in parallel

UpStart are endeavoring to do the same thing with an event-triggered mechanism. A service is not started until a trigger event occurs and it can be activated together with irrelevant services.

On-Demand Activation

SysVinit is a type of init system designed prior to systemd. During initialization, SysVinit activates all the possible background service processes that might be used, although some of them, such as CUPS and SSHD, are rarely or even never used during system runtime. Users have to wait for login until all these service processes are activated. The drawbacks in SysVinit are obvious: slow system boot and a waste of system resources in activating unnecessary service processes.

Some services may rarely or even never be used during system runtime. For example, CUPS, printing services are rarely used on most servers. SSHD is rarely accessed on many servers. It is unnecessary to spend time on starting these services and system resources.

systemd can only be activated when a service is requested. If the service request is over, systemd stops.

Service Life Cycle Management by CGroups

An important role of an init system is to track and manage the life cycle of services, and start/stop them freely. However, it is more difficult than you could ever imagine to encode an init system into stopping services freely.

Things are made simpler with CGroups, which has long been used to manage system resource quotas. The ease of use comes largely from its file-system-like user interface. When a parent service creates a child service, the latter inherits all attributes of the control group to which the parent service belongs. This means that all relevant services are put into the same control group. The systemd can find the PIDs of all relevant services simply by traversing their control group and then stop them one by one.

Mount and Automount Point Management

Traditional Linux systems use the /etc/fstab file to manage file system mount points. These mount points are automatically mounted at system boot time and usually are critical directories, such as the HOME directory. Like SysVinit, systemd also monitors and manages mount points so that they can be automatically mounted at system boot time. In systemd, you can continue to use the /etc/fstab file to manage mount points.

There are times when you need to mount or unmount on demand. This is traditionally achieved using the autofs service.

The systemd allows automatic mount without a need to install autofs.

Transactional Dependency Management

System boot involves a host of separate jobs, some of which may be dependent on each other. For example, a network file system (NFS) can be mounted only after network connectivity is activated. The systemd can run a large number of dependent jobs in parallel, but not all of them. Looking back to the NFS example, it is impossible to mount NFS and activate network at the same time. Before running a job, systemd calculates its dependencies, creates a temporary transaction, and verifies that this transaction is consistent (all relevant services can be activated without any dependency on each other).

Compatibility with SysV Init Scripts

The systemd differs from its predecessor in configuration mode and application development requirements, but is still compatible with its predecessor considering that no Linux distribution would be willing to replace the current init system with systemd at the cost of overhauling its service code.

The compatibility between systemd and SysV and LSB init scripts allows you to upgrade your system to systemd without any changes to existing services or processes, making it easier for systemd to be widely accepted by users.

System State Snapshots and System Restoration

System state varies constantly because services can be activated on demand at any given point in time. The systemd can temporarily save the current state of your operating system or restore a previous state of the operating system from a dynamically created snapshot.

Imagine a system snapshot is created while services A and B are running, and then service A is stopped and some system changes (such as activation of service C) are applied. With the system snapshot, you can return your operating system to the state in which services A and B were running. Snapshot-based restoration is very helpful in debugging scenarios — you can undo any debugging operations when the debug is completed.

5.3 Managing System Services

The systemd provides the systemctl command to start, stop, restart, view, enable, and disable system services.

Comparison Between SysVinit and systemd Commands

The **systemctl** command from the **systemd** command has the functions similar to the **SysVinit** command. Note that the **service** and **chkconfig** commands are supported in this version. For details, see **Table 1**. You are advised to manage system services by running the **systemdemd** command.

Table 5-3 Comparison between SysVinit and systemd commands

SysVinit Command	systemd Command	Description
service foo start	systemctl start foo.service	Starts a service.
service foo stop	systemctl stop foo.service	Stops a service.
service foo restart	systemctl restart foo.service	Restarts a service.
service foo reload	systemctl reload foo.service	Reloads a configuration file without interrupting an operation.
service foo condrestart	systemctl condrestart foo.service	Restarts a service only if it is running.
service foo status	systemctl status foo.service	Checks if a service is running.
chkconfig foo on	systemctl enable foo.service	Enables a service when the service activation time arrives or a trigger condition for enabling the service is met.

SysVinit Command	systemd Command	Description
chkconfig foo off	systemctl disable foo.service	Disables a service when the service activation time arrives or a trigger condition for disabling the service is met.
chkconfig foo	systemctl is-enabled foo.service	Checks whether a service is enabled.
chkconfig –list	systemctl list-unit-files --type=service	Lists all services in each runlevel and checks whether they are enabled.
chkconfig foo –list	ls /etc/systemd/system/*.wants/foo.service	Lists the runlevels in which a service is enabled and those in which the service is disabled.
chkconfig foo –add	systemctl daemon-reload	Used when you need to create a service file or change settings.

Listing Services

To list all currently loaded services, run the following command:

```
systemctl list-units --type service
```

To list all services regardless of whether they are loaded, run the following command (with the all option):

```
systemctl list-units --type service --all
```

Example list of all currently loaded services:

```
$ systemctl list-units --type service
UNIT                           LOAD   ACTIVE SUB     DESCRIPTION
abrt-ccpp.service              loaded active exited  Install ABRT coredump hook
abrt-oops.service              loaded active running ABRT kernel log watcher
abrt-vmcore.service            loaded active exited  Harvest vmcores for ABRT
abrt-xorg.service              loaded active running ABRT Xorg log watcher
abrtd.service                  loaded active running ABRT Automated Bug Reporting
Tool
...systemd-vconsole-setup.service loaded active exited  Setup Virtual Consoletog-
pegasus.service            loaded active running OpenPegasus CIM Server

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

46 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'
```

Displaying Service Status

To display the status of a service, run the following command:

```
systemctl status name.service
```

Table 5-4 describes the parameters in the command output.

Table 5-4 Output parameters

Parameter	Description
Loaded	Information on whether the service has been loaded, the absolute path to the service file, and a note of whether the service is enabled.
Active	Information on whether the service is running and a time stamp.
Main PID	PID of the service.
Status	Additional information about the service.
Process	Additional information about related processes.
CGroup	Additional information about related control groups.

To verify whether a particular service is running, run the following command:

```
systemctl is-active name.service
```

Similarly, to determine whether a particular service is enabled, run the following command:

```
systemctl is-enabled name.service
```

For example, to display the status of gdm.service, run the systemctl status gdm.service command.

```
# systemctl status gdm.service
gdm.service - GNOME Display Manager   Loaded: loaded (/usr/lib/systemd/system/
gdm.service; enabled)   Active: active (running) since Thu 2013-10-17 17:31:23
CEST; 5min ago
 Main PID: 1029 (gdm)
   CGroup: /system.slice/gdm.service
           ├─1029 /usr/sbin/gdm
           ├─1037 /usr/libexec/gdm-simple-slave --display-id /org/
gno...           └─1047 /usr/bin/Xorg :0 -background none -verbose -auth /r...Oct
17 17:31:23 localhost systemd[1]: Started GNOME Display Manager.
```

Starting a Service

To start a service, run the following command as the root user:

```
systemctl start name.service
```

For example, to start the httpd service, run the following command:

```
# systemctl start httpd.service
```

Stopping a Service

To stop a service, run the following command as the root user:

```
systemctl stop name.service
```

For example, to stop the bluetooth service, run the following command:

```
# systemctl stop bluetooth.service
```

Restart a Service

To restart a service, run the following command as the root user:

```
systemctl restart name.service
```

This command stops the selected service in the current session and immediately starts it again. If the selected service is not running, this command starts it too.

For example, to restart the bluetooth service, run the following command:

```
# systemctl restart bluetooth.service
```

Enabling a Service

To configure a service to start automatically at system boot time, run the following command as the root user:

```
systemctl enable name.service
```

For example, to configure the httpd service to start automatically at system boot time, run the following command:

```
# systemctl enable httpd.serviceln -s '/usr/lib/systemd/system/httpd.service'
'/etc/systemd/system/multi-user.target.wants/httpd.service'
```

Disabling a Service

To prevent a service from starting automatically at system boot time, run the following command as the root user:

```
systemctl disable name.service
```

For example, to prevent the bluetooth service from starting automatically at system boot time, run the following command:

```
# systemctl disable bluetooth.servicerm '/etc/systemd/system/dbus-
org.bluez.service'rm '/etc/systemd/system/bluetooth.target.wants/
bluetooth.service'
```

5.4 Changing a Runlevel

Targets and Runlevels

In systemd, the concept of runlevels has been replaced with systemd targets to improve flexibility. For example, you can inherit an existing target and turn it into your own target by adding other services. The table below provides a complete list of runlevels and their corresponding systemd targets.

Table 5-5 Mapping between runlevels and targets

Runlevel	systemd Target	Description
0	runlevel0.target, poweroff.target	The operating system is powered off.

Runlevel	systemd Target	Description
1	runlevel1.target, rescue.target	The operating system is in single user mode.
2	runlevel2.target, multi-user.target	The operating system is in user-defined or domain-specific runlevel (by default, it is equivalent to runlevel 3).
3	runlevel3.target, multi-user.target	The operating system is in non-graphical multi-user mode, and can be accessed from multiple consoles or networks.
4	runlevel4.target, multi-user.target	The operating system is in user-defined or domain-specific runlevel (by default, it is equivalent to runlevel 3).
5	runlevel5.target, graphical.target	The operating system is in graphical multi-user mode. All the services running at level 3 can be accessed through graphical login.
6	runlevel6.target, reboot.target	The operating system is rebooted.

Viewing the Default Target

To determine which target is used by default, run the following command:

```
systemctl get-default
```

Viewing the Current Target

To list all currently loaded targets, run the following command:

```
systemctl list-units --type target
```

Changing the Default Target

To change the default target, run the following command as the root user:

```
systemctl set-default name.target
```

Changing the Current Target

To change the current target, run the following command as the root user:

```
ssystemctl isolate name.target
```

Changing to Rescue Mode

To change the operating system to rescue mode, run the following command as the root user:

```
systemctl rescue
```

This command is similar to systemctl isolate rescue.target, but it also sends an informative message to all login users. To prevent systemd from sending this message, run this command with the --no-wall option: The command is as follows:

```
systemctl --no-wall rescue
```

NOTE

You need to restart the system to enter the normal working mode from the rescue mode.

Changing to Emergency Mode

To change the operating system to emergency mode, run the following command as the root user:

```
systemctl emergency
```

This command is similar to systemctl isolate emergency.target, but it also sends an informative message to all login users. To prevent systemd from sending this message, run this command with the --no-wall option: The command is as follows:

```
systemctl --no-wall emergency
```

NOTE

You need to restart the system to enter the normal working mode from the emergency mode.

5.5 Shutting Down, Restarting, Suspending, and Hibernating the Operating System

systemctl Command

The systemd uses the systemctl command instead of old Linux system management commands to shut down, restart, suspend, and hibernate the operating system. Although old Linux system management commands are still available in systemd for compatibility reasons, but it is advised that you use systemctl when possible.

Table 5-6 Mapping between old Linux system management commands and systemctl

Old Command	systemctl Command	Description
halt	systemctl halt	Shuts down the operating system.
poweroff	systemctl poweroff	Powers off the operating system.
reboot	systemctl reboot	Reboots the operating system.

Old Command	systemctl Command	Description
pm-suspend	systemctl suspend	Suspends the operating system.
pm-hibernate	systemctl hibernate	Hibernates the operating system.
pm-suspend-hybrid	systemctl hybrid-sleep	Hibernates and suspends the operating system.

Shutting Down the Operating System

To shut down the system and power off the operating system, run the following command as the root user:

```
systemctl poweroff
```

To shut down the operating system without powering it off, run the following command as the root user:

```
systemctl halt
```

By default, running either of these commands causes systemd to send an informative message to all login users. To prevent systemd from sending this message, run this command with the **--no-wall** option. The command is as follows:

```
systemctl --no-wall poweroff
```

Restarting the Operating System

To restart the operating system, run the following command as the root user:

```
systemctl reboot
```

By default, running either of these commands causes systemd to send an informative message to all login users. To prevent systemd from sending this message, run this command with the **--no-wall** option. The command is as follows:

```
systemctl --no-wall reboot
```

Suspending the Operating System

To suspend the operating system, run the following command as the root user:

```
systemctl suspend
```

Hibernating the Operating System

To hibernate the operating system, run the following command as the root user:

```
systemctl hibernate
```

To suspend and hibernate the operating system, run the following command as the root user:

```
systemctl hybrid-sleep
```

6 Process Management

This topic explains how Linux kernel manages processes. It also provides examples to help you better understand common process control commands, at and cron services, as well as process query commands.

6.1 Managing System Processes

The operating system manages multiple user requests and tasks. In most cases, the operating system comes with only one CPU and one main memory, but it may have multiple tier-2 disks and input/output (I/O) devices. Therefore, users have to share resources, but it appears to users that they are exclusively occupying resources. The operating system places user tasks, OS tasks, emailing, print tasks, and other pending tasks in the queue and schedules the tasks according to predefined rules. In this topic, you will know how the operating system manages processes.

6.1.1 Scheduling a Process

The time-consuming and resource-demanding part of maintenance work is often performed at late night. You can arrange relevant processes to get started at the scheduled time instead of staying up all night. Here, we will explain the process scheduling commands.

6.1.1.1 Using the at Command to Run Processes at the Scheduled Time

Function

The at command is used to run a batch of processes (a series of commands) at the scheduled time or time+date.

Syntax of the at command:

```
at [-V] [-q queue] [-f filename] [-mldbv] time
at -cjob[job…]
```

Time Format

The scheduled time can be in any of the following formats:

- hh:mm today: If hh:mm is earlier than the current time, the selected commands will be run at hh:mm the next day.
- midnight, noon, teatime (typically at 16:00), or the like
- 12-hour format followed by am or pm
- Time + date (month day, mm/dd/yy, or dd.mm.yy) The scheduled time can also be relative time.

The scheduled time can also be relative time. For example, now+*N* minutes, hours, days, or weeks. **count** is time, which may be a few days or hours. Further, the scheduled time can be words like today, tomorrow, or the like. Here are some examples of the scheduled time.

Imagine the current time is 12:30 June 7 2015 and you want to run a command at 4:30 pm. The scheduled time in the at command can be any of the following:

```
at 4:30pm
at 16:30
at 16:30 today
at now+4 hours
at now+ 240 minutes
at 16:30 7.6.15
at 16:30 6/7/15
at 16:30 Jun 7
```

Although you can select any of the preceding examples according to your preference, absolute time in 24-hour format, such as at 16:30 6/7/15, is recommended.

Privileges

Only commands from standard input or from the file specified by the -f option can be scheduled by the at command to be executed. If the su command is executed to switch the operating system from user A to user B and then the at command is executed at the shell prompt of user B, the at command execution result is sent to user B. whereas emails (if any) are sent to user A.

For example, to run the slocate -u command at 10 am on 8 June 2015, perform the following steps:

```
# at  10:00  6/8/15
at> slocate -u
at>
[1]+   Stopped    at  10:00  6/8/15
```

When the at> prompt appears, type **slocate -u** and press Enter. Repeat substep 2 to add other commands that need to be run at 10 am on 8 June 2015. Then, press Ctrl+d to exit the at command.

The administrator is authorized to run the at command unconditionally. For other users, their privilege to run the at command is defined in /etc/at.allow and /etc/at.deny files.

6.1.1.2 Using the cron Service to Run Commands Periodically

The at command can run commands at the scheduled time but only once. It means that after the running command is specified, the system completes the task at the specified time. If you need to run commands repeatedly, the cron service is a good helper.

Cron Service

The **cron** service searches the **/var/spool/cron** directory every minute for **crontab** files and loads the search results into memory to execute the commands in the **crontab** files. Each user has a crontab file, with the file name being the same as the user name. For example, the **crontab** file of the **globus** user is **/var/spool/cron/globus**.

The **cron** service also reads the cron configuration file **/etc/crontab** every minute, which can be edited in various formats. If no crontab files are found, the **cron** service enters sleep mode and releases system resources. One minute later, the **cron** service is awaken to repeat the search work and command execution. Therefore, the background process occupies few resources and is wakened up every minute to check whether there are commands that need to be run.

Command execution results are then mailed to users specified by the environment variable MAILTO in the /etc/crontab file. The **cron** service, once started, does not require manual intervention except when you need to replace periodic commands with new ones.

crontab Command

The crontab command is used to install, edit, remove, list, and perform other operations on crontab files. Each user has its own crontab files and can add commands to be executed to the files.

Here are common crontab command options:

- crontab -u //Set the **cron** service of a user. This option is required only when the **crontab** command is run by the **root** user.
- crontab -l //List details of the **cron** service of a user.
- crontab -r //Remove the **cron** service of a user.
- crontab -e //Edit the **cron** service of a user.

For example, to list cron service settings of the root user, run the following command:

```
crontab -u root -l
```

crontab Files

Enter the commands to be executed and time in crontab files. Each line in the files contains six fields. The first five fields are the time when the specified command is executed, and the last field is the command to be executed. Fields are separated by spaces ( ) or tabs. The format is as follows:

```
minute hour day-of-month month-of-year day-of-week commands
```

Each field is described as follows:

Table 6-1 Field description

Field	Description
minute	The minute of the hour at which commands will be executed. Value range: 0 – 59.
hour	The hour of the day at which periodic commands will be executed. Value range: 0 – 23.
day-of-month	The day of month at which periodic commands will be executed. Value range: 1 – 31.

Field	Description
month-of-year	The month of year at which periodic commands will be executed. Value range: 1 – 12.
day-of-week	The day of week at which periodic commands will be executed. Value range: 0 – 6.
commands	Periodic commands.

The fields cannot be left unspecified. In addition to numerical values, the following special symbols are allowed: Asterisk (*): a wildcard value. Forward slash (/): followed by a numeral N to indicate that commands will be executed at a regular interval of N. Hyphen (-): used with a range.Comma (,): used to separate discrete numbers. A complete path to the commands shall be provided.

For example, to allow the operating system to add sleepy to the /tmp/test.txt file every two hours from 11 pm to 08 am, add the following line in a crontab file:

```
* 23-8/2 * * * echo"sleepy" >> /tmp/test.txt
```

Each time the cron service settings of a user are edited, the cron service generates in the /var/spool/cron directory a crontab file named after the user. The crontab file can be edited only using the crontab -e command. Alternatively, the user can create a file and run the crontab *filename* command to import its cron settings into the new file.

For example, to create a crontab file for the globus user, perform the following steps: The procedure is as follows:

1. Create a file using any text editor. Add the commands that need to be executed periodically and the command execution interval to the new file. In this example, the new file is ~/globus.cron.
2. Run the following command to install the new file as the crontab file of the globus user: run the following command:
   ```
   crontab  globus. -/globus.cron
   ```

After the new file is installed, you will find a file named globus in the /var/spool/cron directory. This file is the required crontab file.

NOTE

Do not restart the cron service after a crontab file is modified, because the cron service, once started, reads the crontab file every minute to check whether there are commands that need to be executed periodically. You do not need to restart the **cron** service after modifying the **crontab** file.

/etc/crontab File

The **cron** service reads all files in the **/var/spool/cron** directory and the **crontab** file in the **/etc/crontab** directory every minute. Therefore, you can use the **cron** service by configuring the **crontab** file. A crontab file contains user-specific commands, whereas the **/etc/crontab** file contains system-wide commands. Example /etc/crontab file

```
SHELL=/bin/sh
PATH=/usr/bin:/usr/sbin:/sbin:/bin:/usr/lib/news/bin
MAILTO=root //If an error occurs or data is output, the data is sent to the
account by email.
HOME=/
```

```
#  run-parts
01 * * * * root run-parts /etc/cron.hourly //Run scripts in the /etc/cron.hourly
directory once a hour.
02 4 * * *   root run-parts /etc/cron.daily    //Run scripts in the /etc/
cron.daily directory once a day.
22 4 * * 0  root run-parts /etc/cron.weekly     //Run scripts in the /etc/
cron.weekly directory once a week.
42 4 1  * *  root run-parts /etc/cron.monthly     //Run scripts in the /etc/
cron.monthly directory once a month.
```

NOTE

If the **run-parts** parameter is deleted, a script name instead of a directory name is executed.

6.1.2 Suspending/Resuming a Process

A process can be suspended or resumed by job control, and the process will continue to work from the suspended point after being resumed. To suspend a foreground process, press Ctrl+Z. After you press Ctrl+Z, the cat command is suspended together with the foreground process you wish to suspend. You can use the jobs command instead to display a list of shell jobs, including their job names, IDs, and status.

To resume a process in foreground or background, run the fg or bg command, respectively. The process then starts from where it paused previously.

6.2 Viewing Processes

Linux is a multi-task system and needs to get process information during process management. To manage processes, you first need to know the number of processes and their statuses. Multiple commands are available to view processes.

who Command

The who command is used to display system user information. For example, before running the talk command to establish instant communication with another user, you need to run the who command to determine whether the target user is online. As another example, the system administrator can run the who command to learn what each login user is doing at the current time. The who command is widely seen in system administration since it is easy to use and can return a comprehensive set of accurate user information.

The following is an example output of the who command, where system users and their status are displayed: The use of the **who** command is as follows:

```
# who
admin     tty1         Jul 28 15:55
admin     pts/0        Aug  5 15:46 (192.168.0.110)
admin     pts/2        Jul 29 19:52 (192.168.0.110)
root     pts/3        Jul 30 12:07 (192.168.0.110)
root     pts/4        Jul 31 10:29 (192.168.0.144)
root     pts/5        Jul 31 14:52 (192.168.0.11)
root     pts/6        Aug  6 10:12 (192.168.0.234)
root     pts/8        Aug  6 11:34 (192.168.0.234)
```

ps Command

The **ps** command is the most basic and powerful command to view process information. The ps command is used to display process information, including which processes are running, terminated, resource-hungry, or stay as zombies.

A common scenario is using the ps command to monitor background processes, which do not interact with your screen, keyboard, and other I/O devices. **Table 6-2** lists the common ps command options.

Table 6-2 Common ps command options

Option	Description
-e	Displays all processes.
-f	Full output format.
-h	Hides column headings in the listing of process information.
-l	Long output format.
-w	Wide output format.
-a	Lists all processes on a terminal, including those of other users.
-r	Lists only running processes.
-x	Lists all processes without controlling terminals.

For example, to list all processes on a terminal, run the following command:

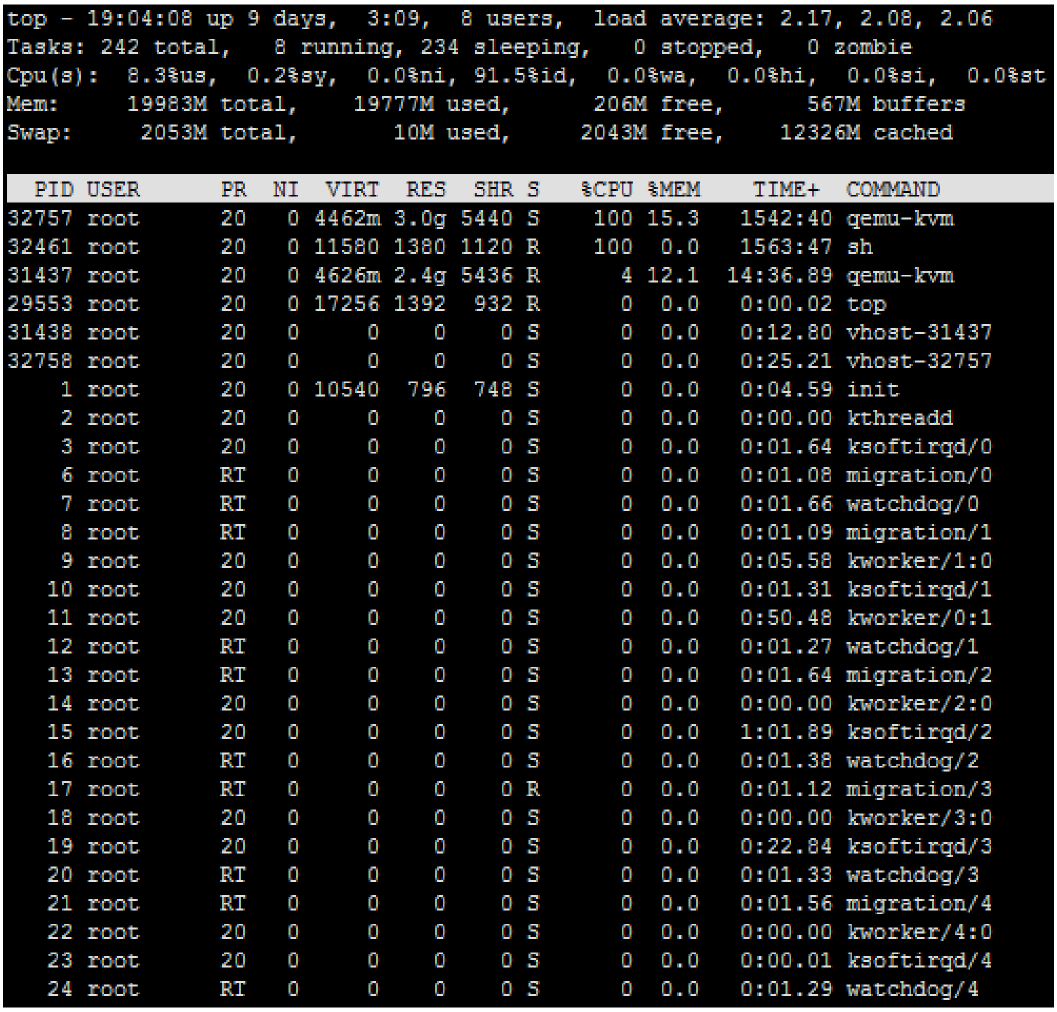
```
# ps -a
  PID TTY          TIME CMD
12175 pts/6    00:00:00 bash
24526 pts/0    00:00:00 vsftpd
29478 pts/5    00:00:00 ps
32461 pts/0    1-01:58:33 sh
```

top Command

Both the top and the ps commands can display a list of currently running processes, but the top command allows you to update the displayed list of processes repeatedly with the press of a button. If the top command is executed in foreground, it exclusively occupies foreground until it is terminated. The top command provides real-time visibility into system processor status. You can sort the list of CPU tasks by CPU usage, memory usage, or task execution time. Extensive customization of the display, such as choice of columns or sorting method, can be achieved using interactive commands or the customization file.

Figure 6-1 provides an example output of the top command.

Figure 6-1 Example command output



```
top - 19:04:08 up 9 days,  3:09,  8 users,  load average: 2.17, 2.08, 2.06
Tasks: 242 total,   8 running, 234 sleeping,   0 stopped,   0 zombie
Cpu(s):  8.3%us,  0.2%sy,  0.0%ni, 91.5%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:     19983M total,    19777M used,      206M free,      567M buffers
Swap:     2053M total,       10M used,     2043M free,    12326M cached

  PID USER      PR  NI  VIRT  RES  SHR S  %CPU %MEM    TIME+  COMMAND
32757 root      20   0 4462m 3.0g 5440 S   100 15.3   1542:40 qemu-kvm
32461 root      20   0 11580 1380 1120 R   100  0.0   1563:47 sh
31437 root      20   0 4626m 2.4g 5436 R     4 12.1  14:36.89 qemu-kvm
29553 root      20   0 17256 1392  932 R     0  0.0   0:00.02 top
31438 root      20   0     0    0    0 S     0  0.0   0:12.80 vhost-31437
32758 root      20   0     0    0    0 S     0  0.0   0:25.21 vhost-32757
    1 root      20   0 10540  796  748 S     0  0.0   0:04.59 init
    2 root      20   0     0    0    0 S     0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S     0  0.0   0:01.64 ksoftirqd/0
    6 root      RT   0     0    0    0 S     0  0.0   0:01.08 migration/0
    7 root      RT   0     0    0    0 S     0  0.0   0:01.66 watchdog/0
    8 root      RT   0     0    0    0 S     0  0.0   0:01.09 migration/1
    9 root      20   0     0    0    0 S     0  0.0   0:05.58 kworker/1:0
   10 root      20   0     0    0    0 S     0  0.0   0:01.31 ksoftirqd/1
   11 root      20   0     0    0    0 S     0  0.0   0:50.48 kworker/0:1
   12 root      RT   0     0    0    0 S     0  0.0   0:01.27 watchdog/1
   13 root      RT   0     0    0    0 S     0  0.0   0:01.64 migration/2
   14 root      20   0     0    0    0 S     0  0.0   0:00.00 kworker/2:0
   15 root      20   0     0    0    0 S     0  0.0   1:01.89 ksoftirqd/2
   16 root      RT   0     0    0    0 S     0  0.0   0:01.38 watchdog/2
   17 root      RT   0     0    0    0 R     0  0.0   0:01.12 migration/3
   18 root      20   0     0    0    0 S     0  0.0   0:00.00 kworker/3:0
   19 root      20   0     0    0    0 S     0  0.0   0:22.84 ksoftirqd/3
   20 root      RT   0     0    0    0 S     0  0.0   0:01.33 watchdog/3
   21 root      RT   0     0    0    0 S     0  0.0   0:01.56 migration/4
   22 root      20   0     0    0    0 S     0  0.0   0:00.00 kworker/4:0
   23 root      20   0     0    0    0 S     0  0.0   0:00.01 ksoftirqd/4
   24 root      RT   0     0    0    0 S     0  0.0   0:01.29 watchdog/4
```

kill Command

The **kill** command is used to terminate a process regardless of whether the process is running in foreground or background. It differs from the combo key **Ctrl+c**, which can terminate only foreground processes. The kill command is used to terminate a process regardless of whether the process is running in foreground or background. The reason for terminating a background process can be heavy use of CPU resources or deadlock.

The kill command sends a signal to terminate running processes. By default, the TERM signal is used. The TERM signal terminates all processes incapable of capturing the TERM signal. To terminate a process capable of capturing the TERM signal, use the KILL signal (signal ID: 9) instead.

Two types of syntax of the kill command:

```
kill [-s signal | -p] [-a] PID…
kill -l [signal]
```

The process ID is retrieved from the ps command. The **-s** option indicates the signal sent to terminate processes. The signal details can be viewed by running the **kill -l** command. The **-p** option indicates the ID of process that will be terminated.

For example, to terminate the process with ID 1409, run the following command:

```
# kill -9 1409
```

Example output of the kill command with the -l option

```
# kill -l
 1) SIGHUP? 2) SIGINT? 3) SIGQUIT? 4) SIGILL
 5) SIGTRAP? 6) SIGABRT? 7) SIGBUS? 8) SIGFPE
 9) SIGKILL?10) SIGUSR1?11) SIGSEGV?12) SIGUSR2
13) SIGPIPE?14) SIGALRM?15) SIGTERM?16) SIGSTKFLT
17) SIGCHLD?18) SIGCONT?19) SIGSTOP?20) SIGTSTP
21) SIGTTIN?22) SIGTTOU?23) SIGURG?24) SIGXCPU
25) SIGXFSZ?26) SIGVTALRM?27) SIGPROF?28) SIGWINCH
29) SIGIO?30) SIGPWR?31) SIGSYS?34) SIGRTMIN
35) SIGRTMIN+1?36) SIGRTMIN+2?37) SIGRTMIN+3?38) SIGRTMIN+4
39) SIGRTMIN+5?40) SIGRTMIN+6?41) SIGRTMIN+7?42) SIGRTMIN+8
43) SIGRTMIN+9?44) SIGRTMIN+10?45) SIGRTMIN+11?46) SIGRTMIN+12
47) SIGRTMIN+13?48) SIGRTMIN+14?49) SIGRTMIN+15?50) SIGRTMAX-14
51) SIGRTMAX-13?52) SIGRTMAX-12?53) SIGRTMAX-11?54) SIGRTMAX-10
55) SIGRTMAX-9?56) SIGRTMAX-8?57) SIGRTMAX-7?58) SIGRTMAX-6
59) SIGRTMAX-5?60) SIGRTMAX-4?61) SIGRTMAX-3?62) SIGRTMAX-2
```

7 Upgrade Mode of EulerOS-based Self-Developed Modules

This chapter describes how to upgrade the **ixgbevf-2.16.4-2** module.

Symptom

Ixgbevf is the kernel module and is stored in the **/lib/modules/<kernel version>/kernel/** directory of the system. The kernel version identified in the kernel module may differ from that in the current system (EulerOS KABI IS is supported and the driver can be properly installed). Therefore, the correct driver version cannot be loaded.

Solution

Create a link to the **/lib/modules/EulerOS** directory in the EulerOS kernel directory and change the path search priority of modprobe. Use modprobe to first search for **/lib/modules/<kernel version>/EulerOS** to ensure that the **ixgbevf** kernel module is properly loaded.

Procedure

If the **ixgbevf** kernel version is found to be incompatible with the OS of a kernel version earlier than 3.10.0-514.35.4.7.h35 during module loading, manually create the **/lib/modules/uname -r/EulerOS** soft link in the **/lib/modules/EulerOS** directory, and add the **/lib/modules/EulerOS/** directory to the search directory list in the **/etc/depmod.d/depmod.conf** file.

8 FAQs

8.1 How Can I Query EulerOS Version Identifier?

The configuration file of EulerOS V2.0SP5 identifier is **/etc/euleros-release**, which contains EulerOS version information. Run the following command to view such information:

```
[root@localhost ~]# cat /etc/euleros-release
EulerOS release 2.0 (SP5)             #EulerOS versions
```

Run the following command to query kernel version information:

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 3.10.0-327.44.58.19.x86_64 #1 SMP Wed Feb 22 17:25:10
UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
```

NOTE

The **/etc/euleros-lastest** file exists in the EulerOS V2.0SP5. This file can be used by developers. Users cannot be used to obtain information about the EulerOS version and kernel version.

8.2 What Are the Network Configuration Constraints?

NetworkManager service is used to perform network management of EulerOS. If NetworkManager service is enabled, only the **nmcli** command or configuration file can be used to configure the network, such as IP addresses and routes. Commands such as **ip**, **ifconfig**, and **route** cannot be used.

NOTE

When NetworkManager service is enabled, if you run commands such as **ip**, **ifconfig**, and **route**, the configurations will be overwritten by NetworkManager service later.

If you want to check whether NetworkManager is enabled, run the following command:

```
systemctl status NetworkManager
```

NOTE

For details about the use of the **nmcli** command, see the execution result of the **nmcli --help** or **man nmcli** command.

If you want to run commands such as ip, ifconfig, and route commands to manage network, run the following command to disable NetworkManager service:

```
systemctl stop NetworkManager
```

NOTE

If only NetworkManager is disabled, it will be re-enabled by other dependent services after the system restart. Therefore, when you want to completely disable NetworkManager in different service scenarios, you are advised to only uninstall the software package of this service by running the following command:

```
rpm -e --nodeps NetworkManager
```

8.3 How Can I Set glibc Memory Management Parameters?

After glibc is upgraded to glibc-2.17-196, the memory management algorithm uses PER_THREAD by default. By default, to weaken the lock contention when multi-thread memory is applied for, glibc creates an area called "arena" for each thread. This change causes virtual and physical memory occupied by some multi-thread applications to increase. The number of arenas can be limited by setting the following environment variables.

For example, to set the number of arenas to **1**, run the following command:

```
export MALLOC_ARENA_MAX=1
```

NOTE

If **MALLOC_ARENA_MAX** is not set, the maximum number of arenas created by glibc is 8 times the number of CPU cores by default.

8.4 How Can I Configure Firewalls to Enhance the Container Network Performance?

The firewall of EulerOS is built based on the firewalld of CentOS. The firewall policy is enabled by default, like most OSs, such as Windows, Red Hat, CentOS, and Ubuntu.

If firewalld is enabled, a series of iptables or ebtable kernel modules and default rules are loaded, decreasing the container network performance by about 20%, which varies with test environments. Under the same test conditions, the performance of EulerOS is slightly better than SuSE12 when the firewall is disabled, and is worse than SuSE12 when the firewall is enabled.

NOTE

If firewall is disabled, all iptables or ebtable rules and modules are uninstalled, and all default and user-defined protection rules are invalid.

Protection Rules and Functions

Protection rules and their functions supported by firewalld are as follows:

1. Zone

   Defines the security level. APIs, connections, and addresses of different security levels are placed in nine different zones, including drop, block, public, external, dmz, work, home, internal, and trusted. For example, home network can be placed into home or trusted zones, whereas common Wi-Fi is suitable for the public zone.

2. Service

   Although Service is used to protect ports and addresses, firewalld abstracts Service, allowing users to easily enable and disable protections on some services. For example:

```
firewall-cmd --permanent --zone=public --add-service=http
```

3. IPSet

   Configurations about the binding of IP and MAC.

4. ICMP Type

   Firewalld protects ICMP for its specialty in the IP network.

5. Direct Interface

   Direct Interface is used to define iptables and ebtable rules in firewalld.

Key Commands

1. Run following commands to check the firewall status:

```
[root@localhost firewalld]# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled)
   Active: active (running) since Wed 2017-03-22 06:17:32 EDT; 1h 25min ago
 Main PID: 28597 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─28597 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid
```

2. Run the following command to enable the firewall:

```
[root@localhost firewalld]# systemctl start firewalld
```

   Run following commands to check the firewall status:

```
[root@localhost firewalld]# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled)
   Active: active (running) since Wed 2017-03-22 10:36:38 EDT; 2s ago
 Main PID: 1500 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─1500 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid
```

3. Run the following command to disable the firewall:

```
[root@localhost firewalld]# systemctl stop firewalld
```

   Run following commands to check the firewall status:

```
[root@localhost firewalld]# systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled)
   Active: inactive (dead) since Wed 2017-03-22 07:43:05 EDT; 4min 15s ago
  Process: 28597 ExecStart=/usr/sbin/firewalld --nofork --nopid
$FIREWALLD_ARGS (code=exited, status=0/SUCCESS)
 Main PID: 28597 (code=exited, status=0/SUCC
```

8.5 How Can I Handle Conflicts Occurred During the Installation of rpm Packages?

Symptom

Conflict may be reported when some rpm packages are installed at the same time. Conflicts may occur in following scenarios:

- Some x86_64 packages have conflicts with i686 packages.
- There are packages providing similar functions.
- There are packages that are incompatible with the spec definition of rpm.

Solution

If conflicts are reported during rpm and yum installation mode, delete rpm packages as prompted. If only documents are conflicted, you can forcibly install rpm packages.

- Example 1:

```
Transaction check error:
file /usr/share/doc/dracut-033/dracut.html from install of
dracut-033-360.h1.i686 conflicts with file from package
dracut-033-360.h1.x86_64
```

  Select either x86_64 or i686 for installation or forcibly install.

```
rpm  -ivh  xxx.rpm --force
```

- Example 2:

```
Error: tog-pegasus-libs conflicts with libcmpiCppImpl0-2.0.3-5.x86_64
```

  Uninstall the conflicted package:

```
yum remove libcmpiCppImpl0
```

8.6 How Can I Delete the Kernel of an Earlier Version?

Symptom

```
rpm  -Uvh kernel
```

or

```
yum  update  kernel
```

If you run the preceding command to upgrade the kernel package, kernel of the earlier version is reserved for rollback in case of error of the new version. Other rpm packages are deleted.

Solution

After kernel is upgraded successfully, run the following command to delete kernel with the earlier version:

```
rpm -e kernel-xxx
```

or

```
yum remove kernel-xxx
```

xxx indicates the version ID of the kernel to be delete.

8.7 How Can I Configure a VM Serial Port Output File?

Background

To facilitate log analysis and shipping, VMs store the data output through VM serial ports in files.

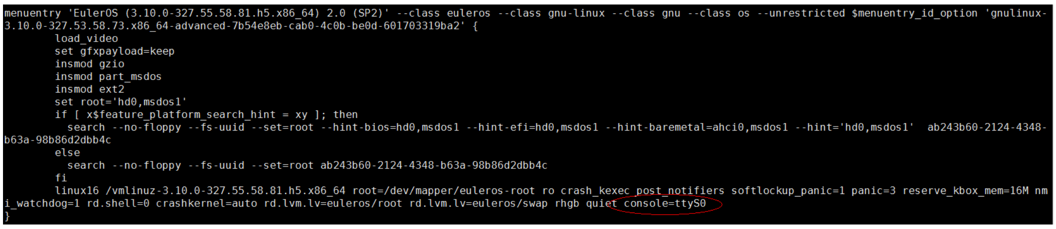
Solution

To implement the preceding function, perform the following steps:

1. Modify kernel startup parameters in the **grub.cfg** configuration file.

Find the kernel to be modified in the **/etc/grub2/grub.cfg** file, and add **console=ttyS0** after kernel startup parameters, as shown in **Figure 8-1**.

Figure 8-1 Adding kernel startup parameters



```
menuentry 'EulerOS (3.10.0-327.55.58.81.h5.x86_64) 2.0 (SP2)' --class euleros --class gnu-linux --class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-
3.10.0-327.53.58.73.x86_64-advanced-7b54e8eb-cab0-4c0b-be0d-601703319ba2' {
        load_video
        set gfxpayload=keep
        insmod gzio
        insmod part_msdos
        insmod ext2
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1'  ab243b60-2124-4348-
b63a-98b86d2dbb4c
        else
          search --no-floppy --fs-uuid --set=root ab243b60-2124-4348-b63a-98b86d2dbb4c
        fi
        linux16 /vmlinuz-3.10.0-327.55.58.81.h5.x86_64 root=/dev/mapper/euleros-root ro crash_kexec_post_notifiers softlockup_panic=1 panic=3 reserve_kbox_mem=16M nm
i_watchdog=1 rd.shell=0 crashkernel=auto rd.lvm.lv=euleros/root rd.lvm.lv=euleros/swap rhgb quiet console=ttyS0
}
```

2. Configure the host.

   Configure the serial port output file path on the host when defining a VM. The following is an example of configuring the source path based on site conditions:

```
<serial type='file'>
<source path='/sdb/test/Euler2.3_vm/serial.log'/>
<target port='0'/>
</serial>
<console type='file'>
<source path='/sdb/test/Euler2.3_vm/serial.log'/>
<target type='serial' port='0'/>
</console>
```

8.8 Precautions on the Default Certificate of OpenLDAP

During the installation of OpenLDAP, the default certificate and its database password file will be provided. You are advised to use your certificate when using OpenLDAP for the first time and properly control the permission because the security level of the preceding default certificate and its file is low.

8.9 Restraints on the Logrotate of rootsh

Logrotate of rootsh can be implemented using logs and determine whether to delete a log file to reserve space for logs using the log file timestamp. When the system time changes, the generated log of rootsh will be deleted.

NOTE

In this case, no action is required to avoid other risks.

8.10 How Can I Enable or Disable the Intel-Side Channel L1TF Vulnerability Patch?

A vulnerability similar to Spectre and Meltdown is found during the hardware implementation of modern microprocessors. This vulnerability affects Intel x86 series microprocessors. For details about the processor list, see https://www.intel.com/content/www/us/en/security-center/advisory/intel-sa-00161.html. Non-privileged attackers can access memory data by exploiting this vulnerability without memory security limitations. The CVE ID of this vulnerability is CVE-2018-3620. In virtualization, the CVE ID of this vulnerability is CVE-2018-3646. The vulnerability is called "L1TF (L1 Terminal Fault)" in the industry or called "Foreshadow" and "Foreshadow-NG" by security personnel.

This vulnerability has an impact on three parts. The first part is the Intel® Software Guard Extensions (Intel® SGX) security zone. This part can be repaired by updating the microcode independent of the operating system. The other two parts need to be repaired by the operating system and hypervisors. To fully repair potential attack risks from untrusted guest VMs in a virtualization environment, you need to perform specific operations required by the system administrator.

EulerOS has provided kernel updates to fix this vulnerability. These updates are as follows:

1. Page Table Inversion: a small change in a kernel. This function is enabled by default in the updated kernel.
2. Flushing the L1 Data Cache: This function is optional during inter-VM switchover. This function can be implemented by updating the kernel or the Intel microcode.
3. SMT Disable: This function is optional. Note that SMT is the abbreviation of simultaneous multi-threading. This function is implemented by modifying the kernel and a new control interface is added. In addition, this function can be implemented by disabling the Intel's hyper-threading on the BIOS. However, this way is not recommended because it is complex.

The preceding updates specially protect the attacked parts.

Description

- Page Table Inversion

  When attackers try to access physical addresses in invalid page table entries (PTEs) listed in the page table. If the physical addresses are cached in the L1 data cache, the storage device access is successful. Data leak occurs. If the physical addresses are not cached, no data leak occurs during the physical address access. EulerOS fixes this defect by inverting the page table in the kernel. Update a physical address in an invalid PTE by setting the high-order address bit to the physical address that is not in the memory or cannot be cached. In this way, the physical address cannot be cached in the L1 data cache. This prevents data leak caused by speculating invalid PTEs.

  After the EulerOS kernel is upgraded, run the following command to view the current status of this function displayed in the sysfs.

  ```
  # cat /sys/devices/system/cpu/vulnerabilities/l1tf
  Mitigation: PTE Inversion; VMX: SMT vulnerable, L1D conditional cache flushes
  #
  ```

  The value of **/sys/devices/system/cpu/vulnerabilities/l1tf** may be as follows:

  ```
  'Not affected'                The current CPUs are not affected by this
  vulnerability.
  'Mitigation: PTE Inversion'   Protection is enabled.
  ```

  If KVM or VMX is enabled and the current CPUs are affected by the vulnerability, the following information is displayed after the **Mitigation: PTE Inversion** value:

  SMT status information is as follows:

  ```
  'VMX: SMT vulnerable'         SMT is enabled.
  'VMX: SMT disabled'           SMT is disabled.
  ```

  L1D cache flushing modes are as follows:

  ```
  'L1D vulnerable'                     L1D cache flushing is disabled.
  'L1D conditional cache flushes'      L1D cache flushing is enabled (automatic
  flushing).
  'L1D cache flushes'                  L1D cache flushing is enabled
  (unconditionally forcible flushing).
  ```

- Flushing the L1 Data Cache

You can use either of the following two methods to flush the L1 data cache:

a. Update the Intel microcode.

If you have updated the Intel microcode to the latest version, the IA32_FLUSH_CMD MSR register is generated for Intel x86 CPUs. This register is used to flush the L1 data cache.

You can run the **cat /proc/cpuinfo** or **lscpu** command to view CPU flags and check whether the CPUs support this register.

```
# lscpu
Flags:    ... ssbd ibrs ibpb stibp spec_ctrl intel_stibp flush_l1d
#
```

If the CPU flags are not found, the Intel microcode is not updated in the current environment. You are advised to update the Intel microcode to the latest one and restart the operating system to enable CPUs to automatically flush the L1 data cache on hardware.

b. Update EulerOS.

If you cannot update the microcode, EulerOS can flush the L1 data cache at the software layer. However, such flushing is slower than that at the hardware layer.

EulerOS provides the startup parameter **l1tf=** to control whether to flush the L1 data cache. The value may be as follows:

Value	Function	Description	Remarks
full	Enables measures to fix all L1TF vulnerabilities, including flushing the L1 data cache and disabling SMT or hyper-threading after VM access operations.	When setting **l1tf** to **full**, you can dynamically control L1 data cache flushing (this function is provided by the KVM modules of the host instead of the guest host) and SMT enabling and disabling through the sysfs interface during the system operation.	The SMT control interface in the sysfs is as follows: /sys/devices/system/cpu/smt/control (For details about the SMT control files provided by the sysfs, see **Control SMT**.)
full,force	Enables measures to fix all L1TF vulnerabilities, including flushing the L1 data cache and disabling SMT or hyper-threading after VM access operations.	When setting **l1tf** to **full,force**, you cannot dynamically control L1 data cache flushing (this function is provided by the KVM modules of the host instead of the guest host) and SMT enabling and disabling through the sysfs interface during the system operation.	-

Value	Function	Description	Remarks
flush	Enables conditional (based on the kernel) L1 data cache flushing and SMT or hyper-threading.	When setting **l1tf** to **flush**, you can dynamically control L1 data cache flushing (this function is provided by the KVM modules of the host instead of the guest host) and SMT enabling and disabling through the sysfs interface during the system operation. In the KVM virtualization scenarios, if insecure configurations are used on VMs, for example, SMT is enabled or L1 data cache flushing is disabled, the KVM hypervisors generate alarms.	Set **l1tf** to **flush** by default.
flush, nosmt	Enables conditional (based on the kernel) L1 data cache flushing and SMT or hyper-threading.	When setting **l1tf** to **flush,nosmt**, you can dynamically control L1 data cache flushing (this function is provided by the KVM modules of the host instead of the guest host)and SMT enabling and disabling through the sysfs interface during the system operation. In the KVM virtualization scenarios, if insecure configurations are used on VMs, for example, SMT is enabled or L1 data cache flushing is disabled, the KVM hypervisors generate alarms.	-

Value	Function	Description	Remarks
flush, nowarn	Enables measures to fix all L1TF vulnerabilities, including flushing the L1 data cache and disabling SMT or hyper-threading after VM access operations. Disables KVM hypervisors to send alarms.	When setting **l1tf** to **full**, you can dynamically control L1 data cache flushing (this function is provided by the KVM modules of the host instead of the guest host) and SMT enabling and disabling through the sysfs interface during the system operation. In the KVM virtualization scenarios, if insecure configurations are used on VMs, for example, SMT is enabled or L1 data cache flushing is disabled, the KVM hypervisors do not send alarms.	-
off	Disables measures to repair the hypervisors. The KVM does not flush the L1 data cache.	-	-

The default value of the startup parameter **l1tf** is **flush**.

Note: The host KVM module provides a sysfs interface to dynamically enable or disable L1 data cache flushing. The sysfs interface is as follows:

```
/sys/module/kvm_intel/parameters/vmentry_l1d_flush
```

This interface can be set to the following three values:

```
always     Flush the L1 data cache when performing the VMENTER operation
on a VM.
cond       Conditionally flush the L1 data cache. The condition is as
follows: In the process of accessing a VM, flush the L1 data cache when
performing the VMENTER operation to avoid code data leakage from the
VMEXIT operation to the VMENTER operation.
never      Disable L1 data cache flushing.
```

For example, during the host operation, dynamically disable L1 data cache flushing by running the following command:

```
echo "never" > /sys/module/kvm_intel/parameters/vmentry_l1d_flush
```

- Control SMT

  When simultaneous multi-threading or hyper-threading technology is used, irrelevant threads in the shared processor cache resources can read data of other threads from the

L1 data cache based on the L1TF vulnerability. If the current environment is an untrusted shared environment, you need to disable SMT to prevent the vulnerability from being exploited.

Either of the following two methods can be used to enable or disable SMT:

a. Enable or disable SMT on the BIOS.

b. Enable or disable SMT using the startup parameter **nosmt** from EulerOS. The details are as follows:

Value	Function	Description	Remarks
nosmt	Disables SMT.	You can dynamically re-enable SMT through the sysfs interface during the system operation.	The sysfs interface information is as follows: /sys/devices/system/cpu/smt/active /sys/devices/system/cpu/smt/control (For details about both control files, see the following description.)
nosmt=force	Disables SMT and sets the value to **force**.	Set the value to **force**. You are not allowed to dynamically re-enable SMT through the sysfs API during the system operation.	The same as above.

EulerOS provides two sysfs interfaces to control SMT. If it is allowed, you can dynamically control SMT through both sysfs interfaces during the system operation. Two sysfs interfaces are as follows:

```
/sys/devices/system/cpu/smt/active
/sys/devices/system/cpu/smt/control
```

Note:

active is a read-only interface. You can run the **cat /sys/devices/system/cpu/smt/active** command to read the value. The value may be **0** or **1**.

```
If the value of active is 0, SMT is disabled. If the startup parameter nosmt
is specified when EulerOS is started, the value of active is 0.
If the value of active is 1, SMT is enabled and logical CPUs are all online.
```

control is an interface for data reading or writing. You can view the enabling status of SMT using this file.

Status	Writable	Description
on	Yes	SMT is enabled. If logical CPUs are offline, enable them.

Status	Writable	Description
off	Yes	SMT is disabled. If logical CPUs are online, disable them.
forceoff	Yes	SMT is forcibly disabled and cannot be dynamically adjusted during the system operation.
nosupport	No	The system CPUs do not support hyper-threading or SMT.

As described in the preceding table, you can enable or disable SMT by setting the **control** value to **on**, **off**, or **forceoff**. Note that the value of the startup parameter **l1tf=** has an impact on the current operation.

For example, during the guest host operation, dynamically disable SMT by running the following command:

```
echo "off" > /sys/devices/system/cpu/smt/control
```

8.11 A Large Number of Small Emails Are Generated in the /var/spool/postfix/maildrop Directory When cron Is Used to Perform Scheduled Tasks

Symptom

When cron is used to perform scheduled tasks, a large number of small emails are generated in the **/var/spool/postfix/maildrop** directory.

Cause Analysis

When EulerOS executes cron, it sends the output and warning information in the **cron** script to the cron scheduled task owner by email. However, the sendmail or postfix on the server does not work, all emails fail to be sent and are stored in the **/var/spool/postfix/maildrop** directory. In addition, the automatic cleanup and conversion mechanism is unavailable. Therefore, more emails are stored in the directory.

Solution

Add **MAILTO=""** to the first line of a crontab file. In this way, when you perform the scheduled tasks, no email is sent.

8.12 How Can I Enable or Disable the Spectre and Meltdown Vulnerability Patches?

For the Spectre and Meltdown vulnerabilities, EulerOS has provided kernel updates to fix these CVE vulnerabilities. For security reasons, these updates are enabled by default in the kernel. However, kernel and microcode updates must cause performance degradation because speculative execution is the performance optimization technology of processors and the security vulnerability fixing affects the original logic of speculative execution. Some users believe that their systems have been protected (for example, physically isolated) and will not be affected by the vulnerabilities. Therefore, they do not want to cause unnecessary performance loss due to vulnerability fixing. Based on the preceding reasons, EulerOS provides relevant enabling and disabling methods. If you believe that it is unnecessary to enable these patches, you can disable the patches by referring to this guide to reduce performance loss.

Description

1. Page Table Isolation (pti) isolates kernel page tables in user mode. This function addresses CVE-2017-5754 (variant 3 or Meltdown).
2. Indirect Branch Restricted Speculation (ibrs), Indirect Branch Prediction Barriers (ibpb), and microcode address CVE-2017-5715 (variant 2 or Spectre).

   NOTE

   In bare metal scenarios, ibrs and ibpb can be enabled only after the microcode is upgraded on a physical machine. If the microcode is not upgraded on a physical machine, both functions are disabled by default and cannot be enabled by changing kernel parameters manually. The corresponding patches do not take effect.

   In guest OS scenarios, ibrs and ibpb can be enabled by upgrading the host microcode and transparently transmitting both functions to the guest OS through host virtualization components.
3. Retpoline (retp) can replace ibrs in the CPU before the Skylake micro-architecture is used and combine ibpb and microcode to address CVE-2017-5715 (variant 2 or Spectre). The Skylake micro-architecture still uses ibrs and ibpb to address CVE-2017-5715 (variant 2 or Spectre).
4. The kernel patch for CVE-2017-5753 (variant 1 or Spectre) cannot be controlled using the preceding functions.

Default Architecture Setting

By default, the kernel automatically sets the enabling or disabling function of the CPU micro-architecture based on the detected CPU micro-architecture at the start stage. The following table lists the default settings of different Intel CPU micro-architectures.

Architecture	pti	ibrs	retp	ibpb	Description
Skylake	1	1	0	1	Fixes variants 1, 2, and 3.
pre-skylake(support mircrocode)	1	0	1	1	Fixes variants 1, 2, and 3.
older Intel CPU with no microcode update available	1	1	0	0	Fixes variants #1 and #3.

Disabling Functions

You can disable functions using either of the following two methods:

1. Permanent disabling (still takes effect after system restart)

   In this method, disable related patches and restart the operating system to make the modification take effect by adding the following parameters to the kernel startup parameters:

   ```
   noibrs noibpb nopti
   ```

   NOTE

   The preceding three parameters can be set separately or simultaneously with other parameters. Some patches are disabled to provide different security protections and avoid performance impacts.

2. Disabling during operation (to be invalid after system restart and reconfigured)

   In this method, disable related patches by running the following commands. The setting immediately takes effect without system restart.

   ```
   # echo 0 > /sys/kernel/debug/x86/pti_enabled
   # echo 0 > /sys/kernel/debug/x86/ibrs_enabled
   # echo 0 > /sys/kernel/debug/x86/retp_enabled
   ```

   NOTE

   - To run the preceding three commands, you need the permission of the **root** user and have loaded the **debugfs** file system. In EulerOS, the **debugfs** file system is loaded by default. To manually uninstall the file system, run the following command:

     ```
     mount -t debugfs nodev /sys/kernel/debug
     ```

   - **ibpb_enabled** is a read-only configuration item, which is set by the kernel.

Verifying the Modification

You can verify whether a function is disabled by running the following commands. If the obtained value is **0**, the function is disabled.

```
# cat /sys/kernel/debug/x86/pti_enabled
# cat /sys/kernel/debug/x86/ibpb_enabled
# cat /sys/kernel/debug/x86/ibrs_enabled
# cat /sys/kernel/debug/x86/retp_enabled
```

NOTE

Although all the functions have been disabled, some applications may have a performance loss of approximately 2%.

spectre_v2 Kernel Startup Parameters

The **spectre_v2** kernel startup parameters address the variant 2 vulnerability.

- on: Unconditionally enables the kernel to address the variant 2 vulnerability.
- off: Unconditionally disables the kernel to address the variant 2 vulnerability.
- auto: Enables the kernel to check whether CPU models have vulnerabilities.

When **spectre_v2** is set to **off**, **ibrs**, **retp**, or **ibpb**, functions are all disabled. You can also use the following methods to address the variant 2 vulnerability:

- retp: Replaces ibrs.
- ibrs: Enables the Intel kernel and patches to run in ibrs mode. This mode prevents the kernel space from being attacked.

- ibrs_always: Enables the Intel kernel, user mode, and patches to run in ibrs mode. This mode prevents the kernel space and user space from being attacked.

NOTE

If **spectre_v2** is not specified, it is set to **auto**.

Checking the System CPU Status

Check whether the Spectre and Meltdown vulnerabilities have impacts on the system CPUs by running the following commands:

```
# cat /sys/devices/system/cpu/vulnerabilities/meltdown
# cat /sys/devices/system/cpu/vulnerabilities/spectre_v1
# cat /sys/devices/system/cpu/vulnerabilities/spectre_v2
```

The file name matches the vulnerability name. The output of these files shows the current CPU status. The output values may be as follows:

```
"Not affected": The current CPUs are not affected by vulnerabilities.
"Vulnerable": CPUs are affected by vulnerabilities and do not take the
corresponding measures to address vulnerabilities.
"Mitigation: $M": CPUs are affected by vulnerabilities and use the $M method to
eliminate the impact of vulnerabilities.
```

8.13 How Can I Enable or Disable the Speculative Store Bypass Vulnerability Patch (for x86_64)?

NOTE

For the Speculative Store Bypass vulnerability, some products, such as unified storage and Dorado, do not provide patches. Therefore, the enabling and disabling configuration and verification methods described in this section are invalid for these products.

There is a vulnerability that exists in modern microprocessors, requiring updates to the Linux kernel, virtualization-related components, and a microcode update. An unprivileged attacker can use this flaw to bypass restrictions in order to gain read access to privileged memory that would otherwise be inaccessible. This issue has been assigned CVE-2018-3639 and is also referred to as "Variant 4" or "Speculative Store Bypass". This issue is known to affect CPUs of various micro-architectures, including AMD, ARM, IBM POWER8, POWER9, SystemZ, and Intel processors. EulerOS is also affected.

CVE-2018-3639 (also called "Speculative Store Bypass") opens a new avenue (like Branch Misprediction) which can be exploited via speculative execution and cache based side channel methods to bypass security measures and access privileged memory. This issue is similar to CVE-2017-5753 ("Spectre v1"), except it leverages Speculative Store Bypass memory optimization in place of Branch Misprediction used by Spectre v1.

For the Speculative Store Bypass vulnerability, EulerOS has provided kernel updates to fix the CVE vulnerability. For security reasons, the preceding updates are enabled by default in the kernel. However, to enable prediction execution is to disable memory disambiguation optimization because prediction execution is a performance optimization technology of processors. This may affect the system performance. The extent to which the performance is affected depends on the specific system load. In the default setting, EulerOS takes the basic principle of giving a priority to security and then performance. Some users believe that their systems have been protected (for example, physically isolated) and will not be affected by this vulnerability. Therefore, they do not want to cause unnecessary performance loss due to vulnerability fixing. Based on the preceding reasons, EulerOS provides relevant enabling and disabling methods. If you believe that it is unnecessary to enable these patches, you can disable the patches by referring to this guide to reduce performance loss.

Answer

Speculative Store Bypass vulnerability fixing only provides the setting method of permanent validation (still valid after system restart).

In this method, restart the operating system to make the modification take effect by adding parameters to the kernel startup parameters (for details, see **Table 8-1** and **Table 8-2**).

NOTE

> EulerOS recommends that users use the microcodes or firmware provided by CPU vendors to update the kernel and install the kernel updates as soon as the kernel updates are available. The software update can be performed without the hardware microcode. However, software can work only after the CPU firmware has been updated.

Table 8-1 spec_store_bypass_disable parameter

Value	Description
auto	Default value. If the value is used during the system startup, the kernel checks whether the processor supports the Speculative Store Bypass function and selects an appropriate mitigation solution. If **spec_store_bypass_disable** is not specified, it is set to **auto**. It is equivalent to set **spec_store_bypass_disable** to **prctl**.
on	Enables the Speculative Store Bypass vulnerability patch. Before all storage (write) addresses are resolved, processors will not predict and execute the loading (read) commands.
off	Disables the Speculative Store Bypass vulnerability patch. Processors use the memory disambiguation function to predict and execute the loading (read) commands before executing the early storage (write) commands.
prctl	Use the prctl(2) interface (for details, see **prctl Interface Description**) to enable the Speculative Store Bypass vulnerability patch based on each thread. For details about the impact of different startup parameters on calling the prctl interface, see **Table 8-4**.

Table 8-2 nospec_store_bypass_disable parameter description

Value	Description
nospec_store_bypass_disable	Sets **spec_store_bypass_disable** to **off**. All the Speculative Store Bypass vulnerability patches are disabled.

Enabling and Disabling Setting

If you want to disable the function permanently, add the following parameters to the kernel startup parameters:

```
Spec_store_bypass_disable=off or nospec_store_bypass_disable
```

Checking the System Vulnerability Fixing Status

Check whether the Speculative Store Bypass vulnerability has an impact on the system by running the following command:

```
cat /sys/devices/system/cpu/vulnerabilities/spec_store_bypass
```

The output value may be as follows:

```
"Not affected": The current CPUs are not affected by the vulnerability.
"Vulnerable": CPUs are affected by the vulnerability and do not take the
corresponding measures to address the vulnerability.
"Mitigation: Speculative Store Bypass disabled": The impact of the CPU
vulnerability has been eliminated using the Speculative Store Bypass
vulnerability patch.
"Mitigation: Speculative Store Bypass disabled via prctl": The impact of the CPU
vulnerability has been eliminated using the prctl Speculative Store Bypass
vulnerability patch.
```

prctl Interface Description

The kernel provides a control option for each process by updating the prctl(2) interface.

Applications can enable or disable Speculative Store Bypass using the control option of each process. Applications can use the prctl(2) interface in the following ways:

- Obtain the current Speculative Store Bypass status of a process. For details about the return value, see **Table 8-3**.

```
prctl(PR_GET_SPECULATION_CTRL, PR_SPEC_STORE_BYPASS, 0, 0, 0);
```

- Enable the Speculative Store Bypass function. No remedy is used.

```
prctl(PR_SET_SPECULATION_CTRL, PR_SPEC_STORE_BYPASS, PR_SPEC_ENABLE, 0, 0);
```

- Disable the Speculative Store Bypass function to make the remedy take effect.

```
prctl(PR_SET_SPECULATION_CTRL, PR_SPEC_STORE_BYPASS, PR_SPEC_DISABLE, 0, 0);
```

Table 8-3 PR_GET_SPECULATION_CTRL return value description

Return Value	Meaning	Description
bit 0	PR_SPEC_PRCTL	If the value is **1**, you can set this process through the prctl(PR_SET_SPECULATION_CTRL) interface to eliminate the impact of the Speculative Store Bypass vulnerability. If the value is **0**, the process cannot be set through the prctl(PR_SET_SPECULATION_CTRL) interface. If this interface is called, the system returns an error message.

Return Value	Meaning	Description
bit 1	PR_SPEC_ENABLE	If the value is **1**, the Speculative Store Bypass function is enabled. The impact of the vulnerability is not eliminated.
bit 2	PR_SPEC_DISABLE	If the value is **1**, the Speculative Store Bypass function is disabled. The Speculative Store Bypass vulnerability patch will be properly applied to eliminate the impact of the vulnerability.

Table 8-4 Impact of different startup parameters on calling the prctl interface

spec_store_bypass_disable	spec_store_bypass Status	get SSB status	set SSB to enable	set SSB to disable
Do not set or set to **auto** or **prctl**.	Mitigation: Speculative Store Bypass disabled via prctl	The default status is as follows: PRCTL: 1 ENABLE: 1 DISABLE: 0	The setting is successful and the new status is as follows: PRCTL: 1 ENABLE: 1 DISABLE: 0	The setting is successful and the new status is as follows: PRCTL: 1 ENABLE: 0 DISABLE: 1
Set to **off** or **nospec_store_bypass_disable**.	Vulnerable	The default status is as follows: PRCTL: 0 ENABLE: 1 DISABLE: 0	The setting fails and the status is still as follows: PRCTL: 0 ENABLE: 1 DISABLE: 0	The setting fails and the status is still as follows: PRCTL: 0 ENABLE: 1 DISABLE: 0

spec_store_bypass_disable	spec_store_bypass Status	get SSB status	set SSB to enable	set SSB to disable
Set to **on**.	Mitigation: Speculative Store Bypass disabled	The default status is as follows: PRCTL: 0 ENABLE: 0 DISABLE: 1	The setting fails and the status is still as follows: PRCTL: 0 ENABLE: 0 DISABLE: 1	The setting fails and the status is still as follows: PRCTL: 0 ENABLE: 0 DISABLE: 1

NOTE

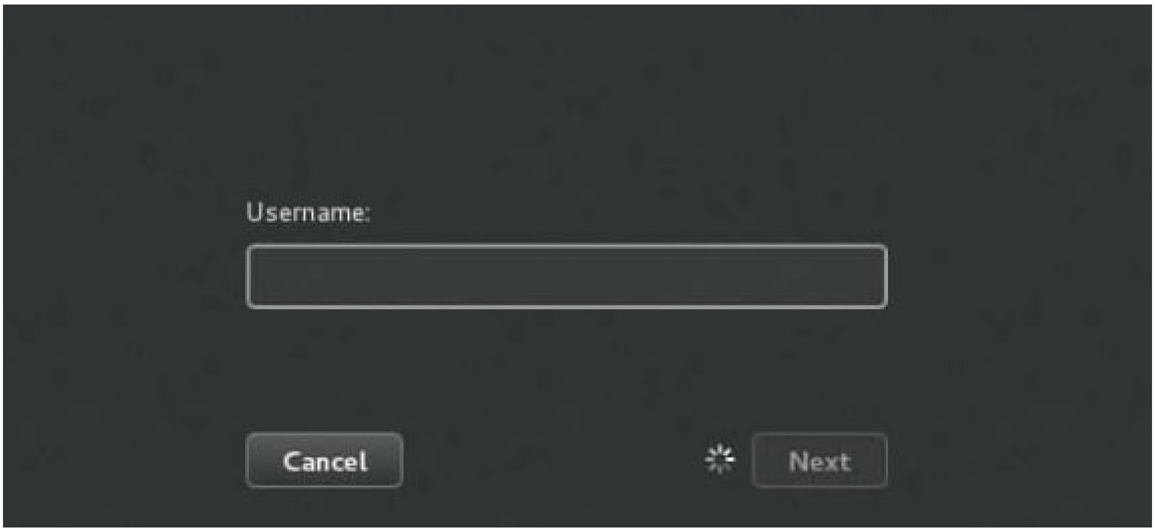
If all values (bits) returned through the prctl(PR_GET_SPECULATION_CTRL, PR_SPEC_STORE_BYPASS, 0, 0, 0) interface are 0, CPUs are not affected by the vulnerability.

8.14 GUI Is Suspended

Symptom

If you enter incorrect passwords for three times when logging in to the GUI, the GUI is locked for 300 seconds. After 300 seconds, you enter the user name and click **Next**, the GUI is suspended. You fail to log in to the GUI. If you stay on the GUI and do not perform any operation for a long period, this fault also occurs, as shown in **Figure 8-2**:

Figure 8-2 Login failure



Cause Analysis

The GNOME graphic login communication depends on the dbus connection between GNOME Shell and GDM. When a user stays on the GUI and does not perform any operation for a long period, the connection automatically breaks due to timeout. If the user continues to log in to the GUI, the GUI is suspended. Click **Cancel** to go back to the login page and rebuild the connection. This fault is rectified.

Solution

Click **Cancel** to go back to the login page, and enter the user name again. The login is successful.

8.15 File Specified by -S Cannot Be Used When a Mail Fails to Be Sent by Running the mail Command

Symptom

If an error message is displayed when the **mail** client runs the following command, the error message is not recorded in the specified file.

```
echo "hello world" | mail -S DEAD= "/root/aaa" -s "test" aaa@huawei.com
```

Cause Analysis

To avoid the risk that the **/root/dead.letter** file generated by default fully occupies the partition when mails fail to be sent, EulerOS does not record mails that fail to be sent in the file by default. Therefore, the **-S** parameter does not work when the **mail** command is executed.

Solution

If you need to record a mail that fails to be sent to a specified file, modify the mail configuration file by running the following command:

```
 #vi /etc/mail.rc
```

Set **set -S DEAD=""** to **set -S DEAD="~/$MY_HOPE_DIR"**.

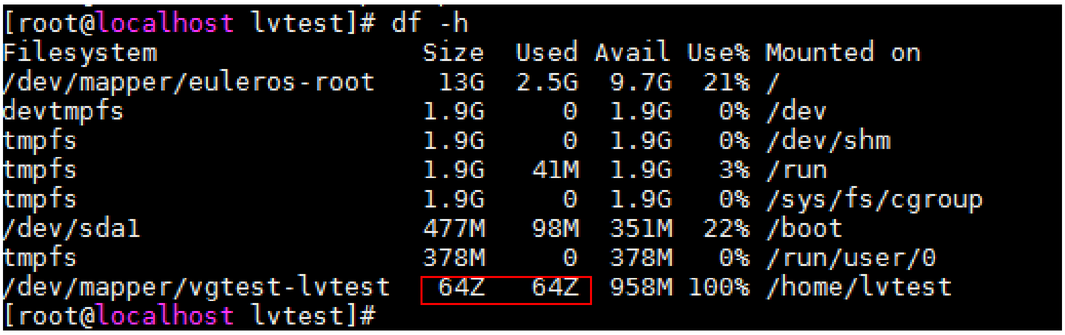
NOTE

$MY_HOPE_DIR is the storage path specified by the user.

8.16 Created Logic Volume Is Automatically Mounted and the Mount Point Is Unavailable

Symptom

After you create a logical volume by running the **lvcreate** command, run the **df -h** command to view mounting information.



```
[root@localhost lvtest]# df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/euleros-root     13G  2.5G  9.7G  21% /
devtmpfs                    1.9G     0  1.9G   0% /dev
tmpfs                       1.9G     0  1.9G   0% /dev/shm
tmpfs                       1.9G   41M  1.9G   3% /run
tmpfs                       1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1                   477M   98M  351M  22% /boot
tmpfs                       378M     0  378M   0% /run/user/0
/dev/mapper/vgtest-lvtest    64Z   64Z  958M 100% /home/lvtest
[root@localhost lvtest]#
```

The output shows that the newly created logical volume is automatically mounted. The size of the logical volume becomes infinite, and the mount point cannot be written.

Cause Analysis

This fault is caused by the following four points:

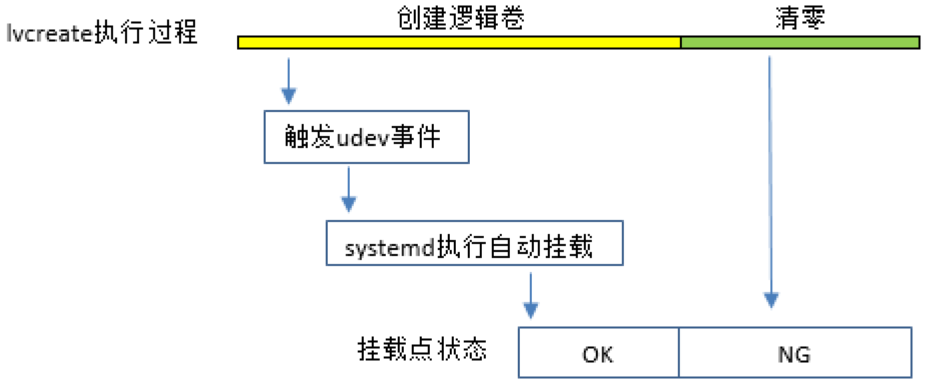
1. The automatic mounting item of the logical volume has been configured in the **/etc/fstab** file and has taken effect.
2. The logical volume has the EXT file system (it is verified that this fault does not exist in the FAT file system. Other file systems are not verified).
3. The following udev-related services have been enabled:

   systemd-udevd

   systemd-udevd-control.socket

   systemd-udevd-kernel.socket
4. The logical volume called xxx in the same volume group is created, deleted, and recreated (the size of the recreated logical volume cannot be smaller than that of the original logical volume).

After a logical volume is deleted, its data related to file systems remains on disks. When a logical volume with the same name and size is created at the same disk location, information about file systems will be reused by the new logical volume. The process of running the **lvcreate** command is as follows:



File systems are mounted after clearing is performed by running the **lvcreate** command, and the mounted file systems are damaged. Therefore, the **df -h** command output is abnormal, and the mount point is unavailable.

Solution

Before deleting a logical volume, clear its file system information. For example, to delete the **/dev/vgtest/lvtest** logical volume by running the following commands:

```
#dd if=/dev/zero of=/dev/vgtest/lvtest bs=1M count=32
#lvremove /dev/vgtest/lvtest
```

8.17 System Is Suspended and Cannot Be Logged In to Because the System Kernel Writes A Large Number of Logs

Symptom

When the kernel mode and user mode write a large number of logs, the system is suspended and cannot be logged in to through SSH.

Cause Analysis

Kernel-mode logs are printed first and then user-mode logs based on the log priority. When the kernel mode writes a large number of logs and the write rate of the kernel mode is greater than the processing rate of the journal, the journal cannot process user-mode logs. When the user mode continues to use the syslog interface to print logs, the **/dev/log** space of the journal socket buffer pool is full and logs cannot be read by the journal. Therefore, return values cannot be obtained for the logs to cause blocking. When each process in user mode is executed, log printing is suspended. Users cannot log in to the system or perform other operations.

Solution

The systemd performance has the fault. Modify the **rsyslog** configuration file to avoid this fault.

Modify the rsyslog configurations from the following two parts:

- Change **$OmitLocalLogging on** to **$OmitLocalLogging off** in the **/etc/rsyslog.conf** file.
- Add **$SystemLogSocketName /dev/log** to the **/etc/rsyslog.d/listen.conf** file.

NOTE

After the preceding modifications are complete, restart the rsyslog service to make the modifications take effect.

8.18 Impact of Inappropriate Firewall Rule Setting or Setting tcp_sack to 0 on Network Performance

Symptom

Improper firewall rule setting (for example, TCPOPTSTRIP tcp -- anywhere anywhere TCPOPTSTRIP options mss,sack-permitted,sack,timestamp,md5) or setting **tcp_sack** to **0** causes the heavy degradation of system network performance in severe packet loss scenarios.

Cause Analysis

In severe packet loss scenarios, when the TCP connection has packet loss, SACK can send the unacknowledged packet sequence numbers from the receiving end to the transmitting end. In this way, the transmitting end can avoid resending all the packets when retransmission is restored.

However, the firewall rule setting (for example, TCPOPTSTRIP tcp -- anywhere anywhere TCPOPTSTRIP options mss,sack-permitted,sack,timestamp,md5) strips information required by SACK. The retransmitted data is lost. Similar to set **tcp_sack** to **0**, the transmitting end retransmits all packets. The overall network performance degrades.

Solution

- In the system, **tcp_sack** is set to **1** by default. Keep the default setting.
- Forbid adding SACK information stripping rules for firewalls.

8.19 OpenLDAP Server Load Specifications

When a client is using SSSD, the client sends 10,000 query requests to the OpenLDAP server within a short period. Approximately 60% to 80% of CPU resources are required.

When less than 20% of CPU resources are available for OpenLDAP server processes, there is a high probability that request timeout occurs on the SSSD client.

Due to the uncertainty of the environment in which the OpenLDAP server is located, it is recommended to set the **ldap_search_timeout** to **10s** on the SSSD client to ensure that the SSSD can work normally (timeout may be avoided) under the conditions that the load of OS deployed on the OpenLDAP server is high and that service processes occupy less CPU resources.

In addition, multiple I/O read and write operations reduce the response speed of the server (the OpenLDAP server needs to read data in databases on disks). A large number of write logs generate multiple I/Os and greatly increase the CPU usage of the journal by occupying more I/Os and CPU resources. Therefore, it is recommended that the value of **olcLogLevel** on the server be not less than 32. The smaller the value, the larger the log volume.

8.20 Failed to Resolve the Host Due to the nslcd OOM in Extreme Cases

The memory usage continuously increases when nslcd has 1000 concurrent connections. When the system memory usage reaches a certain value, nslcd may cause OOM and stop processes (panic_on_oom = 0). Then, nslcd enters the failed state. After a period (one hour by default), host records in the nscd cache start to expire. When you run the command to parse the domain IP address, the cache has expired and nslcd has failed. Therefore, information cannot be obtained from the local cache and the OpenLDAP server. The message "unknown host" is displayed. You are advised to properly limit the number of concurrent connections based on the system memory size to prevent memory from being used up.

8.21 Failed to Resolve the Host Because nscd Writes Negative Entries Into Caches in High Concurrency Scenarios

In high concurrency scenarios, the nslcd client sends a large number of query requests to the OpenLDAP server. However, the OpenLDAP server cannot respond in time. Therefore, the nslcd client cannot query host records and write negative entries into caches to reduce the load on the server. Before negative entries expire (last for 20 seconds by default), the message "unknown host" is returned immediately after the **ping** command is executed and other resolution operations (last for 20 seconds). If you want to obtain positive entries again, wait until negative entries expire or manually clear the host cache of nscd to rebuild the cache. You are advised to properly set the number of concurrent connections on the client to ensure that the OpenLDAP server can stably respond to requests.

8.22 Operation Guidance to the Surprise Hot Plug of NVMe SSDs (Intel VMD)

Application Scenario

As a new hardware integrated with the latest Intel CPU, VMD can effectively manage the indicator status and hot plug of NVMe SSDs. Surprise hot plug: Users can directly insert an NVMe SSD with I/O loads into or remove it from a powered system while the OS is running without notification. This section describes how to properly conduct the surprise hot plug of an NVMe SSD that has been mounted to the system.

Procedure

Assume that the name of the device to be removed is **/dev/nvme0n1p1**, and the mount point is **/mnt/nvme0**,

1. Forcibly remove the NVMe SSD.
2. Run the **umount /dev/nvme0n1p1** command to uninstall the device from the system.

NOTE

If you have not uninstalled the **/dev/nvme0n1p1** device, you will find that the device is still mounted to the mount point **/mnt/nvme0** after running the **df** command. The device name will be used.

8.23 NetworkManager Memory Usage

Symptom

After a large number of created virtual NIC devices are deleted in batches by running the **ip** command, the memory usage of NetworkManager does not decrease. After the same number of virtual NIC devices are created, the memory usage of NetworkManager does not increase.

Cause Analysis

NetworkManager uses the data structures of the glib2 dynamic libraries, such as g_ptr_array, g_array, and ghash, to manage device information. When a large number of devices are concurrently created, deleted, or modified, memory fragments are generated. This is a NetworkManager defect instead of a memory leak.

After different virtual NICs (for example, 1000 for each type of virtual NICs) are added in batches and IP addresses, routes, and NIC information are configured, the memory usage is as follows:

Virtual NIC Type	Quantity	Memory Usage (MB)
veth	1000	192
vlan	1000	128
dummy	1000	192
macvlan	1000	128
bridge	1000	192
bond	1000	192

Solution

Monitor the memory usage of NetworkManager based on the system NIC information. If all virtual NICs are deleted and the NetworkManager memory usage does not decrease, restart the NetworkManager services to release memory.

The service restart command is as follows:

```
systemctl restart NetworkManager
```

8.24 Failed to Modify PAM Configurations by Running the authconfig Command

Symptom

When the **authconfig** command is executed to modify PAM configurations. The modification failed.

Cause Analysis

By default, PAM configurations are security hardened on EulerOS. To prevent the configurations from being overwritten, users fail to make the automatic configuration take effect by running the **authconfig** command.

Solution

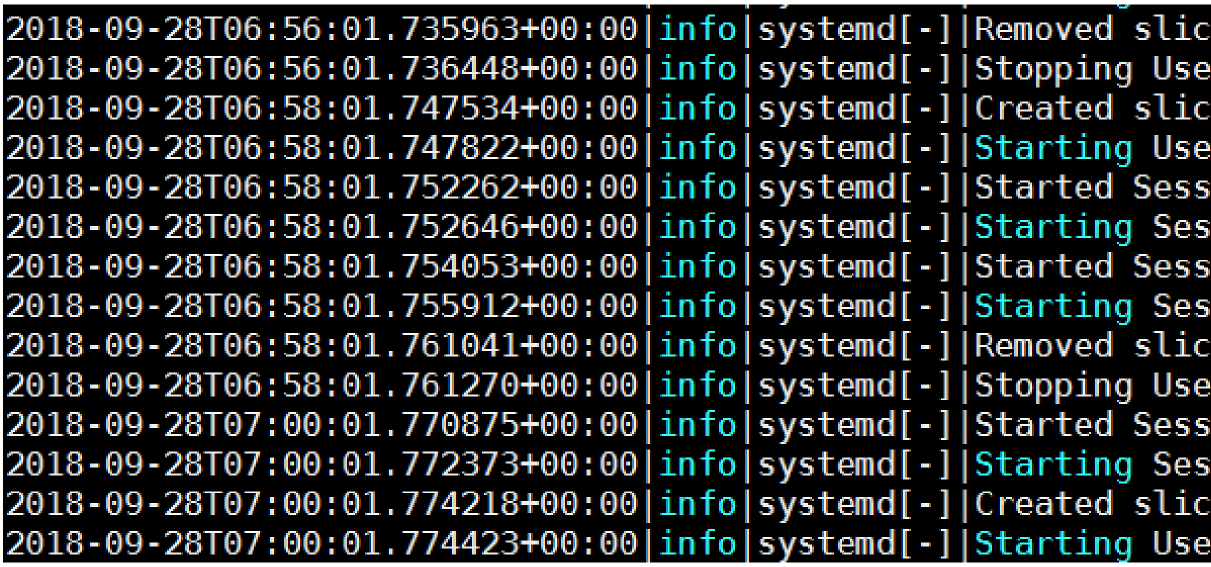
If you want to change the current PAM configurations, use either of the following two methods:

- Method 1: Run the **authconfig** command. After running the command, create a soft link between the **system-auth** file and the **system-auth-ac** file and between the **password-auth** file and the **password-auth-ac** file in the **/etc/pam.d/** directory.
- Method 2: Manually configure the **system-auth-local** and **password-auth-local** files in the **/etc/pam.d/** directory.

8.25 The systemd Frequently Prints the "Starting Session XX of user USERNAME" Log

Symptom

When no operation is performed, logs, such as "Starting Session XX of user USERNAME", are continuously printed in the **/var/log/messages** directory. These logs may be regarded as logs that have been written.



```
2018-09-28T06:56:01.735963+00:00|info|systemd[-]|Removed slic
2018-09-28T06:56:01.736448+00:00|info|systemd[-]|Stopping Use
2018-09-28T06:58:01.747534+00:00|info|systemd[-]|Created slic
2018-09-28T06:58:01.747822+00:00|info|systemd[-]|Starting Use
2018-09-28T06:58:01.752262+00:00|info|systemd[-]|Started Sess
2018-09-28T06:58:01.752646+00:00|info|systemd[-]|Starting Ses
2018-09-28T06:58:01.754053+00:00|info|systemd[-]|Started Sess
2018-09-28T06:58:01.755912+00:00|info|systemd[-]|Starting Ses
2018-09-28T06:58:01.761041+00:00|info|systemd[-]|Removed slic
2018-09-28T06:58:01.761270+00:00|info|systemd[-]|Stopping Use
2018-09-28T07:00:01.770875+00:00|info|systemd[-]|Started Sess
2018-09-28T07:00:01.772373+00:00|info|systemd[-]|Starting Ses
2018-09-28T07:00:01.774218+00:00|info|systemd[-]|Created slic
2018-09-28T07:00:01.774423+00:00|info|systemd[-]|Starting Use
```

Cause Analysis

When cron is running, the user specified for scheduled tasks starts a session through the systemd and executes the corresponding commands or scripts in the session. In this period, related session logs are printed. If the scheduled tasks are frequently executed, the systemd keeps printing session logs to the **messages** file.

```
Euler:~ # cat /etc/cron.d/sysstat
# Run system activity accounting tool every 10 minutes
*/10 * * * * root /usr/lib64/sa/sa1 1 1
```

Solution

The systemd usually prints less session logs. When cron is in heavy load, a large number of session logs are printed in the **messages** file. To filter such session logs, add configurations to the rsyslog by running the following command:

```
echo 'if $programname == "systemd" and ($msg contains "Starting Session" or $msg
contains "Started Session" or $msg contains "Created slice User Slice of"  or
$msg contains "Starting User Slice of" or $msg contains "Removed slice User Slice
of" or $msg contains "Stopping User Slice of") then stop' >/etc/rsyslog.d/ignore-
systemd-session-slice.conf
```

Run the following command to restart the rsyslog to make the configurations take effect:

```
systemctl restart rsyslog
```

8.26 Terms

- **account**

  A combination of name, directories, password, and shell of a login user.

- **alias**

  Alias. A mechanism that enables a replacement of a word in a shell command by another string. To list all defined aliases, type **alias** at the shell prompt.

- **ARP**

  Address Resolution Protocol (ARP), which is used to map IP addresses to MAC addresses of equipment in a local area network.

- **batch**

  Batch processing. A processing mode in which a processor executes a series of jobs without manual intervention until it is ready to receive another series of jobs.

- **boot**

  Boot. A process in which the operating system is loaded into memory after completion of power-on self-tests.

- **bootdisk**

  Boot disk. A removable digital data storage medium from which a computer can load and run (boot) an operating system or utility program.

- **BSD**

  Berkeley Software Distribution (BSD) is a Unix operating system derivative developed and distributed by the Computer Systems Research Group (CSRG) of the University of California, Berkeley.

- **buffer**

  Buffer. A memory region with fixed capacity to load region mode files, system partition tables, processes that are being executed, and so on. Buffers in memory can be contiguous.

- **buffer cache**

  Buffer cache. An import part of operating system kernel to keep all buffers in the up-to-date state and free up memory space by clearing unnecessary buffers.

- **CHAP**

Challenge-Handshake Authentication Protocol (CHAP), a communication protocol used by ISP to verify its clients. Unlike the Password Authentication Protocol (PAP), CHAP periodically initiates verification after the initial verification.

- **client**

  Client. A program or computer that connects temporarily to other programs or computers and issues instructive commands or information requests to the latter. It is a part of client/server system.

- **client/server system**

  Server or client system. A system architecture composed of a server and one or more clients.

- **compilation**

  Compilation. The act of transforming source code written in a human-readable programming language such as the C language into binary files that can be recognized by machines.

- **completion**

  Automatic completion. A feature of command line interpreters, in which shell automatically fills in partially typed commands.

- **compression**

  Compression. A method of encoding information using fewer characters or smaller files than the original representation. Common compressors include compress, zip, gzip, and bzip2.

- **console**

  Console. A machine used by users to operate a computer and often referred to as a terminal. A machine used by users to operate a super computer and often referred to as a terminal. The console to operate a PC is the keyboard and screen.

- **cookies**

  A small piece of data sent from a website and stored in the user's web browser while the user is browsing. Every time the user loads the website, the browser sends the cookie back to the server to notify the user's previous activity.

- **DHCP**

  Dynamic Host Configuration Protocol (DHCP) is a communication protocol used within a local area network to allocate IP addresses dynamically to DHCP clients.

- **DMA**

  Direct memory access (DMA), a feature of computer systems that allows interface devices to access main system memory independently of the central processing unit (CPU).

- **DNS**

  Domain Name System (DNS), a mechanism of mapping machine names to IP addresses recognizable for network devices, or conversely.

- **DPMS**

  Display Power Management System (DPMS), a standard for managing power supply of monitors. Display Power Management System (DPMS), a standard for managing power supply of monitors. Example usage includes shutting off the monitor after a period of idle time to save power.

- **editor**

Editor. A type of program used for editing plain text files. Popular GNU/Linux editors include Emacs and VIM.

- **email**

  Email. A method of exchanging digital messages between users in the same network. The addresses of the sender and recipients must be correct.

- **NVRAM**

  Non-volatile random-access memory (NVRAM) is random-access memory that retains its information when power is turned off (non-volatile).