РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Руководство пользователя

QSR-2830

www.qtech.ru

Оглавление

BASIC CONFIGURATION

1. CONFIGURING THE COMMAND LINE INTERFACE

This chapter describes how to configurethe command line interface(CLI) to manage network devices.

1.1. Command Mode

The management interface of Qtech devices has multiple modes and they determine the commands you can use.

To listusable commands in each mode, enter a question mark (?) at the command prompt.

After setting up a session connection to the network device management interface, you enteruser EXEC mode first. In the user EXEC mode, only a few commands are usable with limited functions, for example, the **show**command. The command results are also not saved.

To use all commands, enter privileged EXEC mode with the privileged password. Then you can use all privileged commands and enter global configuration mode.

Using commands in configuration (for example, global configuration or interface configuration) modewill influence the current configuration. If you have saved the configuration information, these commands will be saved and executed when the system restarts. To enter any of the configuration modes, enter global configuration mode in the first.

The following table describescommand modes, access methods, prompts, and exit methods.Suppose the device is named "Qtech" by default.

The following table summariesmain command modes.

Command mode	Access method	Prompt	Exit or enter the next mode	Remark
User EXEC	Log in.	Qtech>	To quit this mode, enter the **exit** command. To enter privileged EXEC mode, enter the**enable**command.	Performs basic tests and showssystem information.
Privileged EXEC	In the user EXEC mode, enter the**enable**command.	Qtech#	To return touser EXEC mode, enter the**disable**command. To enter global configuration mode, enter command **configure**.	Verifiessettings. This mode is password-protected.
Global configuration	Inprivileged EXEC mode, enter the**configure terminal**command.	Qtech(config)#	To return to privileged EXEC mode, enter command **end** or **exit** or press Ctrl+C. To access the interface configuration mode, enter command **interface**with an interface specified. To access VLAN configuration mode, enterthe **vlan***vlan_id*command.	Executes commands to configure global parameters influencing the entireswitch.
Interface configuration	Inglobal configuration mode, enter the **interface**command.	Qtech(config-if)#	To return to privileged EXEC mode, enter command **end** orpress Ctrl+C. To return to global configuration mode, enter the**exit**command. Moreover, you need specify an interface in the **interface** command.	Configures various interfaces of the device in this mode.
Config-vlan (Vlan Mode)	Inglobal configuration mode, enter the**vlan** vlan-idcommand.	Qtech(config-vlan)#	To return to privileged EXEC mode, enter command **end** or pressCtrl+C. To return to global configuration mode, enter the**exit**command.	Configures VLAN parameters in this mode.

1.2. Obtaining Help

Enter a question mark(?) at the command prompt to obtain a list of commands that are available for each command mode. You can also obtain a list of command keywords beginning with the same character or parameters of each command. See the following table.

Command	Description
Help	Obtainsthe brief description of the help system under any command mode.
abbreviated-command-entry?	Obtains a list of commands that begin with a particular character. string. For example: Qtech# **di?** dir disable
abbreviated-command-entry <Tab>	Completes a partial command name. For example: Qtech# **show conf<Tab>** Qtech# **show configuration**
?	Listskeywordsassociated witha command. For example: Qtech# **show ?**
command keyword ?	Listsarguments associated witha command. For example: Qtech(config)# **snmp-server community ?** WORD SNMP community string

1.3. Abbreviating Commands

To abbreviate a command, simply enter part of the command that can uniquely identify the command.

For example, **show configuration**can be abbreviated as follows:

```
Qtech# show conf
```

1.4. Using the No Form and the Default Form

Most commands have the **no**formthatdisables a feature or function, or performs a reversed action of acommand. For example, the **no shutdown**command turns on aninterface, which is the reversed action of the **shutdown** command. You can use the commands without the keyword**no**to enable the features that have been disabled or are disabled by default.

Most configuration commands have the **default**formthat restoresa command setting to its default. The **default** form is disabled for most commands by default. In this case, the **default** and **no**formsgenerally serve the same purpose. However, the default form is enabled for some commandsby default. In this case, the **default** and **no**formsserve different purposes, where the **default**formenables the command and restores the arguments to the default settings.

1.5. Understanding CLI Error Messages

The following table describes the error messages that may occur when you use the CLI to manage devices.

Error Message	Meaning	How to Obtain Help
% Ambiguous command: "show c"	The switch cannot identify the unique command because you have entered insufficient characters.	Re-entersthe command with the question mark (?)next tothe ambiguous word. The possible keywords will be listed.
% Incomplete command.	You have not entered the required keywords or arguments.	Re-entersthe command withthespace followed by thequestion mark (?). The possible keywords or arguments will be displayed.
% Invalid input detected at '^' marker.	The symbol (^)indicates the positions of wrong words when user entersa wrong command.	Entersthequestion mark (?) at the command prompt to show the allowed keywords of the command.

1.6. Using Historical Commands

The system records the commands you have entered, whichis very useful when you entera long and complex command again.

To re-execute the commands you have entered, perform the following operations.

Operation	Result
Ctrl-P or **Up**	Allows you to browse the preceding command in the historical command records.You can inquire earlier records by repeating this operationfrom the latest record,.

Operation	Result
Ctrl-N or **Down**	Allows you to return to a more recent command in the historical command records. You can inquire laterrecords by repeating this operation.

1.7. Using Editing Features

1.7.1. Editing Shortcut Keys

The following table describes the editing shortcut keys.

Function	Shortcut Key	Description
Move cursor in an editing line	Left direction key or Ctrl+B	Moves the cursor to left by one character.
	Right direction key or Ctrl+F	Moves the cursor to right by one character.
	Ctrl+A	Moves the cursor to the beginning of the command line.
	Ctrl+E	Moves the cursor to the end of the command line.
Delete the entered characters	Backspace	Deletes the character to the left of the cursor.
	Delete	Deletes the character to the right of the cursor.
Scroll up by one line or one page	Return	Scrolls up one line of the display contents anddisplaythe next line. This is used only before the end of the output.
	Space	Scrolls up one page of the displayed contents anddisplaythe next page appear. This is used only before the end of the output.

1.7.2. Sliding Window of Command Lines

You can use this function to edit acommand that exceeds the width of one line. When the editing cursor closes to the right border, the whole command line will move to the left by 20 characters. In this case, the cursor can still be moved back to the previous character or the beginning of the command line.

The following table describes shortcut keys used in this function:

Function	Shortcut key
Moves the cursor to the left by one character.	Left direction key or Ctrl+B
Moves the cursor to the head of a line.	Ctrl+A
Moves the cursor to the right by one character.	Right direction key or Ctrl+F
Moves the cursor to the end of a line.	Ctrl+E

For example, the contents of the **mac-address-table static** command may exceed the screen width. When the cursor approaches the line end for the first time, the whole line move left by 20 characters, and the hidden beginning part is replaced by the symbol ($) on the screen. The line moves left by 20 characters when the cursor reaches the right border.

access-list 199 permit ip host 192.168.180.220 host

$ost 192.168.180.220 host 202.101.99.12

$0.220 host 202.101.99.12 time-range tr

Now you can press **Ctrl+A** to return to the beginning of the command line. In this case, the hidden ending part is replaced by the symbol ($).

access-list 199 permit ip host 192.168.180.220 host 202.101.99.$

⚠ Caution The default line width on the terminal is 80 characters.

Combined with historical commands,the sliding window enables you to invoke complicated commands repeatedly. For details about shortcut keys, see the description about the section"Editing Shortcut Keys".

1.8. Filtering and Searching CLI Output Information

1.8.1. Filtering and Searching the Output Information of the show Command

Use the following command to search the specified content in theoutput informationof the **show** command.

Command	Description
Qtech# **show** any-command |**begin**regular-expression	Searches the specified content in the outputinformation of the **show** command andexportsthe first line that contains the specified content and allinformation after the line.

Caution

- You can execute show command in any mode.
- The information to be searched is case sensitive, and the feature is also effective in the following.

Use the following commands to filter the specified content in the output information ofthe **show** command:

Command	Description
Qtech# **show** any-command **|exclude**regular-expression	Filters the content inthe output information ofthe **show** command andexportsother information excluding the line that includes the specified content.
Qtech#**show**any-command **|include**regular-expression	Filters the content inthe output information ofthe **show** command and exports the line that includes the specified content.Other information will be filtered.

Caution

To search and filter the contents exportedby the **show** command, you must enter the pipeline sign, which is the vertical bar (|) followed by searchand filterrules and contents (characters or strings). The contents are case sensitive.

1.9. Using Command Aliases

The system provides the command alias function.You can specify any word as the alias of a command. For example, you can define the word “mygateway” as the alias of the **ip route 0.0.0.0 0.0.0.0 192.1.1.1** command.The effect of enteringthis word is equal to that of enteringthe entire command.

You can use one word to replace one command by configuring an alias for the command. For example, you can define an alias to represent the firstpart of acommand, and then continue to enter the rest parts.

The command that an alias represents must run under the mode you have defined in the current system. In global configuration mode, you can enter **alias?** to list all command modes that can configure aliases.

```
Qtech(config)#alias ?
  aaa-gs                 AAA server group mode
acl                     acl configure mode
  bgp                     Configure bgp Protocol
config                 globle configure mode
......
```

An alias supports help information. An alias appears withtheasterisk (*) before it in the following format:

*command-alias=original-command

For example, in EXEC mode, the alias “s” indicates the **show** command by default. You can enter “s?” to obtain the help information on the command and the aliases beginning with ‘s’.

```
Qtech#s?
*s=show  show  start-chat  start-terminal-service
```

If the command that an alias represents containsmore than one word, the command will be included by the quotation marks. As shown in the following example, you can configure the alias “sv” to replace the **show version**command in EXEC mode.

```
Qtech#s?
*s=show  *sv=”show version”  show  start-chat
start-terminal-service
```

An alias must begin with the first character of the command line entered without any spacebefore it. As shown in the preceding example, the alias is invalid if you have entereda spacebefore the command.

```
Qtech# s?
show  start-chat  start-terminal-service
```

An alias can also be used to obtain the help information of command parameters. For example, the alias “ia" represents the **ip address** commandin interface configuration mode.

```
Qtech(config-if)#ia ?
  A.B.C.D  IP address
  dhcp     IP Address via DHCP
Qtech(config-if)#ip address
```

The preceding informationlists the parameter information after the command **ip address**, and replaces the alias with the actual command.

A completealias must be entered for use. Otherwise, it can not be identified.

To view the setting of aliases in the system, use the **show aliases** command.

1.10. Accessing the CLI

Before using the CLI, you need to use a terminal or PC to connect toanetwork device. Power on the network device.After initializing the hardware and software, you can use the CLI. If the network device is used for the first time, you can only connect to the network deviceoverthe serial port (Console), which is referred to as out-band management. In addition, you can connect and manage the network device through the virtual terminal of Telnet. In either case, you can access the CLI.

2. CONFIGURING LINE MODE

2.1. Configuring LINE Mode

2.1.1. Entering the LINE mode

After entering the specific LINE mode, you can configure the specified line. Use the following command to enter the specified LINE mode:

Command	Function
Qtech(config)# **line [aux | console | tty | vty]** *first-line* [*last-line*]	Enters the specified LINE mode.

2.1.2. Increasing/Decreasing LINE VTY

The number of line vty is 5 by default. Use the following commands to increase or decrease line vty. 36 line VTYs are supported at most.

Command	Function
Qtech(config)# **line vty** *line-number*	Increases the number of LINE VTY to the specified value.
Qtech(config)# **no line vty** *line-number*	Decreases the number of LINE VTY to the specified value.

2.1.3. Configuring the Protocols to Communicate on the Line

Use this command to limit the communication protocol type supported on the line. By default, VTY supports communication of all protocols while TTY does not support the communication of any protocol.

Command	Description
Qtech# **configure terminal**	Enters the configuration mode.
Qtech(config)# **line vty *line number***	Enters the line configuration mode.
Qtech(config-line)# **transport input {all | ssh | telnet | none**}	Configures the protocol to communicate on the line.
Qtech(config-line)# **no transport input**	Disables the communication of any protocol on the line.
Qtech(config-line)# **default transport input**	Restores the default value.

2.1.4. Configuring the Access Control List on the Line

Use this command to configure the access control list on the line. No access control list is configured on the line by default.That is, all incoming and outgoing connections are permitted.

Command	Description
Qtech# **configure terminal**	Enters the configuration mode.
Qtech(config)# **line vty** *line number*	Enters the line configuration mode.
Qtech(config-line)# **access-class** *access-list-number* **{in | out}**	Configures the access control list on the line.
Qtech(config-line)# **no access-class** *access-list-number* **{in | out}**	Removes the configuration.

3. SYSTEM UPGRADE AND MAINTENANCE

3.1. Overview

Upgrade and maintenance refers to upgrade the main program or CTRL program or upload and download files on the CLI . There are two ways to upgrade programs: use TFTP through a network interface or use Xmodem protocol through a serial interface.

3.2. Upgrade and Maintenance Method

3.2.1. Transferring Files by TFTP

There are two ways to transfer files by TFTP: download files from the host to the equipment, or upload files from the equipment to the host.

In the CLI command mode, download the files by performing the following steps:

Before download, first run the TFTP server software on the local host. Then, select the directory of the file to download. Finally, log in to the equipment. In the privileged mode, download the files by using the following commands. If no location is specified, you need to separately input the IP address of the TFTP server.

Command	Function
Qtech# **copy tftp:** *//location/ filename* **flash:** *filename* [**vrf** *vrfname*]	Download the specified file from the URL on the host to the equipment.

In the CLI command mode, upload the files by performing the following steps:
Before upload, first run the TFTP server software on the local host. Then, select the destination directory for the file to upload at the host. Finally, upload the files by using the following commands in the privileged mode.

Command	Function
Qtech# **copy flash:** *filename* **tftp:** *//location/filename* [**vrf** *vrfname*]	Upload the specified file from the equipment to the directory specified by the URL on the host. You can also rename the file.

Note

It is necessary to put the tftp link in quotes if the filename of the source file has space. For example:

Note

copy tftp:”//localtion/filename” **flash:**filename [**vrf** vrfname]

Note

It is necessary to put the filename in quotes if the filename of the destination file has space. For example:

Note

copy tftp://localtion/filename **flash:**”filename” [**vrf** vrfname]

3.2.2. Transferring Files by TFTP IPv6

There are two ways to transfer files by TFTP: download files from the host to the equipment, or upload files from the equipment to the host.

In the CLI command mode, download the files by performing the following steps:

Before download, first run the TFTP server software on the local host. Finally, log in to the equipment. In the privileged mode, download the files by using the following commands.

Command	Function
Qtech# **copy tftp:** *//location/filename* **flash:** *filename*	Download the specified file from the URL on the host to the equipment.

In the CLI command mode, upload the files by performing the following steps:

Before upload, first run the TFTP server software on the local host. Then, select the destination directory for the file to upload at the host. Finally, upload the files by using the following commands in the privileged mode.

Command	Function
Qtech# **copy flash:** *filename* **tftp:** *//location/filename*	Upload the specified file from the equipment to the directory specified by the URL on the host. You can also rename the file.

Caution

If location is the local link address, use the following command to specify the egress:

```
Qtech#copy tftp: flash:
Address of remote host []?fe80::5efe:192.168.195.90
Output Interface: loopback 0
Source filename []?rgos.bin
Extended commands [n]:
Destination filename [rgos.bin]?
```

3.2.3. Transferring Files by XMODEM

There are two ways to transfer files by Xmodem: download files from the host to the equipment, or upload files from the equipment to the host.

In the CLI command mode, download the files by performing the following steps:

Prior to download, first log in to the out-band management interface of the device by using the Windows

HyperTerminal. Then, download the files by using the following command in the privileged mode. Finally, select the "Send File" from the "Transfer" menu on the Windows HyperTerminal on the local host.

In the pop-up dialog box, select the file to download from the File Name field and Xmodem from the Protocol field. Click "Send", and the Windows HyperTerminal will show the transmission process and packets.

Command	Function
Qtech# **copy xmodem flash**:*filename*	Download the file from the host to the equipment and name it *filename*.

In the CLI command mode, upload the files by performing the following steps:

Prior to upload, first log in to the out-band management interface of the switch by using the Windows HyperTerminal. Then, upload the files by using the following command in the privileged mode. Finally, select the "Receive File" from the "Transfer" menu on the Windows HyperTerminal on the local host.

In the pop-up dialog box, select the storage location for the file to upload and select the "Xmodem" as the reception protocol. Click "Receive", and the Windows HyperTerminal will further prompt the name of the locally stored file. Click "OK" to start reception.

Command	Function
Qtech# **copy flash**:*filename* **xmodem**	Upload the file from the equipment to the host.

Caution It is necessary to put the filename with space in quotes. For example:

```
copy xmodem flash:"filename" OR copy flash:"filename" xmodem
```

3.2.4. Upgrading System

You can transfer the upgrading file to a device through TFTP or Xmodem, no matter the device is box-mount or chassis-mount. After transmission, restart the device. The upgrading file will automatically check and upgrade the system without manual interference.

The upgrade procedure on the box-mount equipment is slightly different from that on the chassis-mount equipment:

On the box-mount equipment, the upgrading file upgrades only its single supervisor engine. After upgrading, the system automatically resets. The equipment works normally after restart.

The chassis-mount equipment includes supervisor engines, line cards and multi-service cards. To upgrade the whole system with a upgrading file, first upgrade the supervisor engine. The system resets. When the equipment restarts, the automatic version synchronization function runs to upgrade line cards and multi-service cards.

Automatic Upgrade: a function running on the supervisor engine that verifies the version consistency for the slave supervisor engine, line cards and multi-service cards. When it is found that the version is not consistent with the one in the master supervisor engine, the function sends the upgrading files to those blades for upgrading so as to keep the version consistence in the whole system.

Caution Whenever you upgrade the master supervisor engine, the slave one (if any) is upgraded at the same time to keep the version consistent. The upgrade of a line card will upgrade all the line cards inserted into the device. Do not power off the device before the upgrade is complete. Otherwise, the upgrade program may be lost.

Caution

Before the chassis-mount device is upgraded, you can check whether the software version of all line cards and supervisor engines are consistent with the upgraded object version by the **show version** command. However, you cannot carry out master-slave switch (such as **redundancy force-switchover**). Otherwise, it will cause the upgrade failure and return to the original version.

Upgrade the chassis-mounted device by the upgrade file:

Confirm the filename of the upgrade file to be loaded is rgos.bin.

Download the file to the device by using the copy command.

If there is a slave supervisor engine on the device, you need to first upgrade the main programs of the master and slave supervisor engines successfully. After upgrading the main program successfully, the system prompts:

```
Upgrade Slave CM MAIN successful!!
Upgrade CM MAIN successful!!
```

Reset the equipment.

After reset, the upgrade file will run automatically. The system prompts:

```
Installing is in process ......
Do not restart your machine before finish !!!!!!
......
```

After the upgrade operation is completed, the system prompts:

```
Installing process finished ......
Restart machine operation is permitted now !!!!!!
```

After the operation of the upgrade file is completed, the system resets automatically and prompts:

```
System restarting, for reason 'Upgrade product !'.
```

After reset, the upgrading operation of the supervisor engines is completed. The system will load and operate the upgrade pack of boards. Moreover, it prompts information in Steps 5 to 6. Instead of the information in Step 7, it prompts:

```
System load main program from install package ......
```

Load the main program of the supervisor engine to operate from the upgrade file directly.

After the main program operates normally, the automatic upgrade function starts. If there is the slave supervisor engine or other modules in the chassis-mount device, the system prompts:

```
A new card is found in slot [1].
System is doing version synchronization checking ......
Current software version in slot [1] is synchronous.
System needn't to do version synchronization for this card ......
```

Or, the system prompts:

```
System is doing version synchronization checking ......
Card in slot [3] need to do version synchronization ......
```

Other Printing Information

```
Version synchronization begain ......
Keep power on, don't draw out the card and don't restart your machine before
finished !!!!!!
```

Other Printing Information

```
Transmission is OK, now, card in slot [3] need restart ...
Software installation of card in slot [3] is in process ......
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
Software installation of card in slot [3] has finished successfully ......
The version synchronization of card in slot [3] get finished successfully.
```

The former indicates the version of the line card is synchronous and it is not necessary to upgrade again. The latter indicates the version of the line card, and it is necessary to upgrade the line card.

The system will carry out above operation for the slave supervisor engine and each module in turn.

After checking the version consistency on all modules and upgrading, the system will work normally

Caution

During the upgrade or automatic upgrade, the system may prompt that the reboot is not allowed. In this case, neither power off or reset the system nor plug or unplug other modules

Note

Automatic upgrading and checking also applies to the system with hot-plugging modules.

Upgrade the box-mount device by the upgrade file

To upgrade the box-mount device, do Steps 1 to 7, and then the system resets. After that, the equipment runs well.

4. CONFIGURING BASIC MANAGEMENT FEATURES

4.1. Overview

Note

For more information about the CLI commands mentioned in this chapter, see the Device Management Command Reference.

4.1.1. Access Control through Command Authorization

4.1.1.1. Overview

A terminal's network access can be simply managed by using passwords and assigning privileged levels. Passwords restrict access to a netwdydyork or network device. Privileged levels define the commands you can use after logging in to a network device.

For security sake, passwords are stored in a configuration file. Passwords must be kept secure when the configuration file is transmitted, for example, over TFTP or across a network. Passwords are encrypted before they are saved into the configuration file. Plain text passwords becomes cipher text passwords. The **enable secret** command builds on a private encryption algorithm.

4.1.1.2. Configuring Default Passwords and Privileged Levels

No password at any level is available by default. The default privileged level is 15.

4.1.1.3. Configuring or Changing the Passwords at Different Levels

Our products provide the following commands for you to configure or change passwords at different levels.

Command	Function
Qtech(config)# **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}	Sets a static password. You can only set a level-15 password when no level-15 security password is configured. If you have set a non-level-15 password, the system will show a message and automatically convert it into a security password. If you have set the same level-15 static password as the level 15 security password, the system will show a warning message. If the encryption type is set to 0, the following password is configured in plain text. If the encryption type is set to 7, the following password is configured in cipher text.
Qtech(config)# **enable secret** [**level** *level*] {*encryption-type encrypted-password*}	Sets the security password, which provides the same function but a better encryption algorithm than a static password. For security sake, it is recommended you use a security password.
Qtech(config)#**service password-encryption**	Determines whether to encrypt the related password.
Qtech# **enable** [*level*], and Qtech# **disable** [*level*]	Switches between user levels. To move from a lower to a higher level, input the password for the higher level.

When you set a password, the keyword "level" is used to define the password for a specified privileged level. After setting, it only works for the users at that level.

4.1.1.4. Configuring Multiple Privileged Levels

By default, the system provides only two password-protected levels: normal user (level 1) and privileged user (level 15). You can configure up to 16 hierarchical command levels for each mode. By configuring different passwords at different levels, you can use different sets of commands for different levels.

When no password is set for the privileged user level, you can enter the privileged mode without password authentication. For security, it is recommended you set the password for the privileged user level.

Configuring Command Authorization

To expand the application scope of a command, you can assign it to users at lower levels. On the contrary, to narrow the scope, you can assign it to users at higher levels.

You can use the following commands to authorize users to use a command:

Command	Function
Qtech# **configure terminal**	Enters global configuration mode.
Qtech(config)# **service display command privilege**	Enables command-level display. After this function is enabled, you can enter ? to view the level of a command key.
Qtech(config)# **privilege** *mode* [**all**] {**level** *level* | **reset**} *command-string*	Sets a privileged level for a command. *mode* – The CLI command mode in which you authorize the command. For example, config indicates global configuration mode, exec indicates privileged command mode, and interface indicates interface configuration mode. **all** – Changes the privileges of all the sub-commands of a specified command to the same level. **level** *level* – Authorization level in the range from 0 to 15. **reset**：Restores the command privilege to the default level. *command-string:* Indicates the command you want to authorize.

To restore the configuration for a specified command, use the **no privilege** *mode* [**all**] **level** *level* command in global configuration mode.

Example of Command Authorization Configuration

The following example shows the configuration process that sets the **reload** command and all its sub-commands to level 1, and activates level 1 (by setting the command as “**test**”):

```
Qtech# configure terminal
Qtech(config)# service display command privielge
Qtech(config)# privilege exec all level 1 reload
Qtech(config)# enable secret level 1 0 test
Qtech(config)# end
```

Enter level 1, and you can see the command and its subcommands:

```
Qtech# disable 1
Qtech> reload ?
  at                       reload at a specific time/date (privilege: 14)
  cancel                   cancel pending reload scheme (privilege: 14)
  in                       reload after a time interval (privilege: 14)
  <cr>
```

The following example shows the configuration process that restores the privilege settings of the reload command and all its sub-commands to the default value:

```
Qtech# configure terminal
Qtech(config)# privilege exec all reset reload
Qtech(config)# end
```

Enter the level 1, the privilege setting for the command is removed.

```
Qtech# disable 1
Qtech> reload ?
% Unrecognized command.
```

4.1.1.5. *Configuring Line Password Protection*

Our products offer password authentication for remote logins (such as Telnet). A password is required for protection. Execute the following command in line configuration mode:

Command	Function
Qtech(config-line)# **password** [**0** | **7**] *line*	Specifies a line password. 0: The password is configured in plaintext. 7: The password is encrypted by a Qtech device. Line: the character string of the password to be configured.
Qtech(config-line)# **secret** { [**0**] *password* | **5** *encrypted-secret* }	Specifies the line’s password encrypted by irreversible MD5 **0**: (Optional) specifies the plaintext password text and encrypts it with irreversible MD5 after configuration. *Password*: The password plaintext.. ***5*** *encrypted-secret*: Specifies the password text encrypted by irreversible MD5 and saves it as the encrypted password after configuration.
Qtech(config-line)# **login**	Enables line password protection.

Note

If no login authentication is configured, password authentication on the line layer will be ignored even when a line password is configured. Login authentication will be discussed in the next section.

Note

If no login authentication is configured, password authentication on the line layer will be ignored even when a line password is configured. Login authentication will be discussed in the next section.

4.1.1.6. *Supporting Session Locking*

Our products allow you to lock the session terminal temporarily using the lock command, so as to prevent unauthorized access. To do so, enable the terminal locking function in the line configuration mode, and lock the terminal using the lock command in terminal EXEC mode: The system prompts you for a password for unlocking when you enter any character on a locked terminal. The terminal is locked when your password is authenticated.

Command	Function
Qtech(config-line)# **lockable**	Enables the function of locking the line terminal.
Qtech# lock	Locks the current line terminal.

4.1.2. Login Authentication Control

4.1.2.1. Overview

The previous section discusses how to control access to network devices by configuring a locally stored password. In addition to line password protection and local authentication, in AAA mode, we can authenticate users' management privilege based on usernames and passwords on some servers when they log in to the switch. Take an RADIUS server for example.

With an RADIUS server, the network device sends encrypted user information to the RADIUS server for authentication instead of authenticating them with locally stored credentials. The RADIUS server configures user information consistently like user name, password, shared key, and access policy to facilitate user access management and control and enhance the security of user information.

4.1.2.2. Configuring Local Users

Our products support identity authentication system based on a local database for local authentication of the method list in AAA mode and local authentication of line login management in non-AAA mode.

To enable username identity authentication, run the following commands in global configuration mode:

Command	Function
Qtech(config)# **username** *name* [**password** *password* | *encryption-type encrypted password*]	Enables username identity authentication with encrypted password. Encryption type 0 defines a password in plaintext. Encryption type 7 defines an encrypted password.
Qtech(config)# **username** *name* **secret** { [**0**] *password* | **5** *encrypted-secret* }	Sets the password encrypted by irreversible MD5 for the local user.
Qtech(config)# **username** *name* [**privilege** *level*]	Sets the privilege level for the user (optional).

4.1.2.3. Confining the Simultaneously Online Amount of a Local Username

Qtech products support local usernames confining the simultaneously online amount. By default, local usernames does not limit the simultaneously online amount.

Run the following commands to limit the simultaneously online amount of a local username:

Command	Function
Qtech(config)# **username** *name* **online amount** *numbers*	Confines the simultaneously online amount of a local username.
Qtech(config)# **no username** *name* **online amount**	Cancels the limit on the simultaneously online amount of a local username.

After the simultaneously online amount of a local username is set, the number of clients logging in with the username must be within the specified range. When the number exceeds the limit, the username is not allowed to be used for login.

When the simultaneously online amount of a local username is set to 0, no login is allowed with the username by any client, including console login and remote login through this user.

4.1.2.4. Confining Username Login Mode

Qtech products support configuration of local username login mode. Login mode can be one type or several types among aux, console ssh and telnet. By default, when there is no restriction on local username login mode, the local username will not confine user login mode.

Run the following demands in global configuration mode to confine local username login mode:

Command	Function
Qtech(config)#**username** *name* **login mode** { **aux | console | ssh | telnet** }	Confines local username login mode.
Qtech(config)#**no username** *name* **login mode** { **aux | console | ssh | telnet** }	Cancels restriction on local username login mode.

This command is used to set local username login mode to one type or several types among aux, ssh and telnet. Only the configured login mode is allowed while the other modes are prevented.

4.1.2.5. Configuring Line Login Authentication

To enable line login identity authentication, run the following commands in line configuration mode:

Command	Function
Qtech(config-line)# login local	Sets local authentication for line login in non-AAA mode.
Qtech(config-line)# login authentication {default | list-name}	Sets AAA authentication for line login in AAA mode. The authentication methods in the AAA method list will be used for authentication, including Radius authentication, local authentication and no authentication.

Note

For more information on how to set AAA mode, configure Radius service, and configure the method list, see the sections relating to AAA configuration.

4.1.3. System Time Configuration

4.1.3.1. Overview

Every switch has its clock, which indicates date (year, month, day) and time (hour, minute, second) and week. When using a switch for the first time, you must configure the clock manually. Of course, you can adjust the clock when necessary. The clock is used for system login that requires you to record the time of an event.

4.1.3.2. Setting System Time and Date

You can configure the system time on the network device manually. Once configured, the clock will be running continuously even if the network device is powered off. Therefore, unless you need to modify the time, it is not necessary to reconfigure the time.

However, the configuration does not apply to network devices without hardware clock, as the manual time setting actually configures software clock. When the network devices are powered off, you cannot set the time manually.

Command	Function
Qtech# **clock set** hh:mm:ss month day year	Sets system date and time.

For example, change the system time to10:10:12, 2003-6-20:

```
Qtech# clock set 10:10:12 6 20 2003                    //Set system time and date.
Qtech# show clock                       //Confirm the modification takes effect.
clock: 2003-6-20 10:10:54
```

4.1.3.3. Showing System Time and Date

You can show system time and date by using the **show clock** command in privileged mode. The following example shows the format:

```
Qtech# sh clock     //Show the current system time and date.
clock: 2003-5-20 11:11:34
```

4.1.3.4. Updating Hardware Clock

Some platforms use hardware clock (calendar) to double as software clock. Since battery enables hardware clock to run continuously, hardware clock still runs even though the device is turned off or restarted.

If hardware clock and software clock are out of sync, software clock prevails. Execute the clock update-calendar command to copy date and time from software clock to hardware clock.

In privileged mode, execute the **clock update-calendar** command for software clock to overwrite hardware clock.

Command	Function
Qtech# **clock update-calendar**	Updates hardware clock through software clock.

Execute the following command to copy current date and time from software clock to hardware clock.

```
Qtech# clock update-calendar
```

4.1.4. Scheduled Restart

4.1.4.1. Overview

This section describes how to use the **reload** [*modifiers*] command to schedule system restarts at specified time. This feature facilitates your operation in some scenarios (for testing, for example). Modifiers is a set of options provided by the **reload** command to increase the command flexibility. The optional modifiers includes **in**, **at** and **cancel**. See the following examples for details:

- **reload in** *mmm* | *hhh:mm* [*string*]

This command sets up the system to restart at regular intervals in the format of mmm or hhh:mm. string is a help prompt. You can give the scheme a name using the string to indicate its purpose. string is a prompt. For example, to reload the system at intervals of 10 minutes for testing, type **reload in** *10 test*.

- **reload at** hh:mm month day year [string]

This command sets up the system to restart at a specified future time. The parameter year is optional. The year recorded in the system clock is shown by default if no year is specified. As the time span is limited to 30 days, the current system date generally ranges between January 1 and November 30. Therefore, you do not need to specify the year. However, the restart time you have specified can be sometime next January if the system currently shows December. In this situation, you must specify a year to instruct the system to restart next January rather than this January. The system may fail as the restart time is considered to fall in this January by default. string is used in a similar way. For example, input **reload at** *08:30 11 1 newday* if the current system time is 14:31 on January 10, 2015 and you want the system to reload tomorrow. If the current system time is 14:31 on December 10, 2015 and you want the system to reload at 12:00 a.m. on January 1, 2016, input **reload at** *12:00 1 1 2016 newyear*.

- **reload cancel**

This command deletes a user-defined restart scheme. As mentioned earlier, you have specified the system to reload at 8:30 a.m. tomorrow, the setting will be canceled after you input **reload cancel**.

Note

If the system supports clock function, you can use option at. Before the use, it is recommended you configure the system clock as required. If a restart scheme has been set before, subsequent settings will overwrite previous settings. If you have set a restart scheme and you restart the system before the scheme takes effect, the scheme will be lost.

Note

The span between the time indicated in the restart scheme to the current time must be within the range of 200 days and must be later than the current system time. Besides, after you have set reload, you should not set system clock. Otherwise, your setting may fail if the system time is later than the reload time.

4.1.4.2. Specifying the System to Restart at the Specified Time

In privileged mode, you can configure system reload at the specified time using the following commands:

Command	Function
Qtech# **reload at** *hh:mm month day* [*year*] [*reload-reason*]	Reloads at hh:mm,month day,year. reload-reason (if any); indicates the reason that the system reloads.

The following example shows an example of system reload at 12:00 a.m. January 11, 2015 (suppose the current system clock is 8:30 a.m. January 11, 2015):

```
Qtech# reload at 12:00 1 11 2015 midday//Set the reload time and date.
Qtech# show reload //Confirm the modification takes effect.
Reload scheduled for 2015-01-11 12:00  (in 3 hours 29 minutes)16581 seconds.
At 2015-01-11 12:00
Reload reason: midday
```

4.1.4.3. Specifying the System to Restart after a Period of Time

In privileged mode, you can configure the system reload at the specified time using the following commands:

Command	Function
Qtech# **reload in** *mmm* [*reload-reason*]	Configures the system reload in mmm minutes, where the reload reason is described in reload-reason (if entered)
Qtech# **reload in** *hhh:mm* [*reload-reason*]	Configure the system reload in hhh hours and mm minutes, where the reload reason is described in reload-reason (if entered)

The following example shows how to reload the system in 125 minutes (assume that the current system time is 12:00 a.m. January 10, 2015):

```
Qtech# reload in 125 test //Set the system reload time
Or
Qtech# reload in 2:5 test        //Set the system reload time
Qtech# show reload  //Confirm whether the restart time change takes effect
System will reload in 7485 seconds.
```

4.1.4.4. Immediate Restart

The **reload** command without any parameter will restart the device immediately. In privileged mode, you can restart the system immediately by using the **reload** command.

4.1.4.5. Deleting the Configured Restart Scheme

In privileged mode, use the following command to delete the configured restart scheme:

Command	Function
Qtech# **reload cancel**	Deletes the configured restart scheme.

If no reload scheme is used, an error message appears.

4.1.5. Configuring a System Name and Prompt

4.1.5.1. Overview

For easier management, you can configure a system name for the switch to identify it. If you configure a system name that contains more than 32 characters, the first 32 characters are used as the system prompt. The prompt varies with the system name. The system is named Qtech by default.

4.1.5.2. Configuring a System Name

Our products provide the following commands to configure a system name in global configuration mode:

Command	Function
Qtech(Config)# **hostname** *name*	Sets the system name. The name must contain up to 63 printable characters.

To restore the name to the default value, use the **no hostname** command in global configuration mode. The following example changes the device name to RGOS:

```
Qtech# configure terminal        //Enter global configuration mode.
Qtech(config)# hostname RGOS             //Set the equipment name to RGOS
RGOS(config)#                                    //The name has been modified
successfully.
```

4.1.5.3. Configuring a Command Prompt

The system name appears as the default if you have not configured any command prompt. (If the system name exceeds 32 characters, the first 32 characters will be blocked.) The prompt varies with the system name. You can use the **prompt** command to configure a command prompt in global configuration mode. The command prompt only applies in EXEC mode.

Command	Function
Qtech# **prompt** *string*	Sets the command prompt with printable characters. If the name exceeds 32 characters, the first 32 characters are blocked.

To restore the prompt to the default value, use the **no prompt** command in global configuration mode.

4.1.6. Banner Configuration

4.1.6.1. Overview

When a user logs in to the switch, you may need to give the user useful information through a banner. There are two kinds of banners: message-of-the-day (MOTD) and login banner. The MOTD is unique to users who connect with switches. When users log in, the notification message will appear on the terminal. MOTD allows you to send urgent messages (for example, the system is shutting down) to network users. The login banner also appears on all connected terminals. It provides some common login messages. By default, no MOTD and login banners are configured.

4.1.6.2. Configuring a Message-of-the-Day

You can create a notification of single or multi-line messages that appears when a user logs in the switch. To configure the message of the day, execute the following commands in global configuration mode:

Command	Function
Qtech(Config)# **banner motd** *c* *message c*	Specifies the message of the day, with c being the delimiter, for example, a pound sign (&). After entering the delimiter, press Enter. Now, you can type text. You need to input the delimiter and then press Enter. Note that if you type additional characters after the ending delimiter, these characters will be discarded by the system. Also note that you cannot use the delimiter in the message and the message can contain a maximum of 255 bytes.

To delete the MOTD, use the no banner motd command in global configuration mode. The following example describes how to configure a MOTD. The # symbol is used as the delimiter, and the text is “Notice: system will shutdown on July 6th.”

```
Qtech(config)# banner motd #                                //Start delimiter.
Enter TEXT message.  End with the character '#'.
Notice: system will shutdown on July 6th.#               //End delimiter.
Qtech(config)#
```

4.1.6.3. Configuring a Login Banner

To configure a login banner, execute the following commands in global configuration mode:

Command	Function
Qtech(Config)# **banner login** *c* *message c*	Specifies the text of the login banner, with c being the delimiter, for example, a pound sign (&). After entering the delimiter, press Enter. Now, you can start to type text. You need to input the delimiter and then press Enter. Note that if you type additional characters after the end delimiter, these characters will be discarded by the system. Also note that you cannot use the delimiter in the text of the login banner and the text can contain a maximum of 255 bytes.

To delete the login banner, use the **no banner login** command in global configuration mode.

The following example shows how to configure a login banner. The pound sign (#) is used as the starting and end delimiters and the text of the login banner is "Access for authorized users only. Please enter your password."

```
Qtech(config)# banner login #    //Start delimiter
Enter TEXT message. End with the character '#'.
Access for authorized users only. Please enter your password.
#      //End delimiter
Qtech(config)#
```

4.1.6.4. Displaying a Banner

A banner is displayed when you log in the network device. See the following example:

```
C:\>telnet 192.168.65.236
 Notice: system will shutdown on July 6th.
 Access for authorized users only. Please enter your password.
 User Access Verification
Password:
```

“Notice: system will shutdown on July 6th." is a MOTD banner and "Access for authorized users only. Please enter your password." is a login banner.

4.1.7. Viewing System Information

4.1.7.1. Overview

You can check some system information using the show command on the command-line interface, such as version and device information.

4.1.7.2. Viewing System Information and Version

System information includes description, power-on time, hardware version, software version, BOOT-layer software version, and CTRL-layer software version.This information helps you know the system better. You can show system information using the following commands in privileged mode.

Command	Function
Qtech# **show version**	Shows system information.

Note

For a sequence number, run the **show version** command on the main program interface to view SYSTEMUPTIME in the form of DD:HH:MM:SS.

Note

During upgrading, the running software version may differ from the version in the file system. In this case, the main program version shown by running the show version command is the one running in the memory, but the Boot/Ctrl version is the one stored in the flash memory.

4.1.7.3. Viewing Hardware Entity Information

Hardware information relate to physical devices, slots and modules assembled in a device. The information on a device includes description, number of slots,slot information, slot number, description of the module on the slot (empty description if no module is plugged in the slot), the number of physical ports of the module in the slot, and the maximum number of ports supported in the slot (the number of ports on the plugged module). You may use the following commands to show the information about the device and slots in privileged mode:

Command	Function
Qtech# **show version devices**	Shows the current device information.
Qtech# **show version slots**	Shows current information about slots and modules.

4.1.8. Setting Console Rate

4.1.8.1. Overview

The device provides a console interface for management. When using the switch for the first time, you need to perform configuration through the console interface. You can change the console rate on the device if necessary. Note that the rate of the terminal used to manage the switch must be the same as that of the console interface on the switch.

4.1.8.2. Setting Console Rate

In line configuration mode, execute the following command to set the console rate:

Command	Function
Qtech(config-line)# **speed** *speed*	Sets transmission rate in bps on the console interface. For a serial interface, you can only set the transmission rate to any one of the following values: 9600, 19200, 38400, 57600 and 115200 bps, with 9600 bps by default.

This example shows how to configure the baud rate of the serial interface to 57600 bps:

```
Qtech# configure terminal //Enter global configuration mode.
Qtech(config)# line console 0    //Enter the console line configuration mode
Qtech(config-line)# speed 57600  //Set the console rate to 57600bps
Qtech(config-line)# end          //Return to the privileged mode
Qtech# show line console 0       //View the console configuration
CON     Type     speed   Overruns
* 0     CON      57600   0
Line 0, Location: "", Type: "vt100"
Length: 25 lines, Width: 80 columns
Special Chars: Escape  Disconnect  Activation
               ^^x     none        ^M
Timeouts:    Idle EXEC    Idle Session
               never          never
History is enabled, history size is 10.
Total input: 22 bytes
Total output: 115 bytes
Data overflow: 0 bytes
stop rx interrupt: 0 times
Modem: READY
```

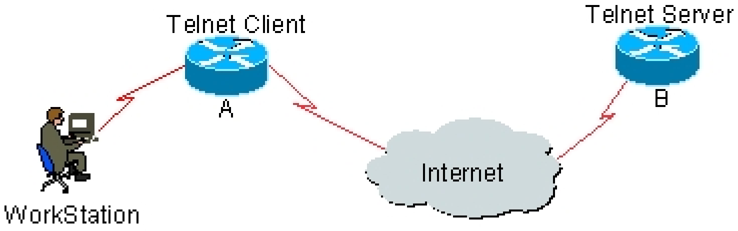
4.1.9. Configuring Telnet

Overview

Telnet, as an application layer protocol in the TCP/IP protocol suite, provides the specifications for remote login and virtual terminal communication. The Telnet Client service is used by a local or remote user who has logged onto the local network device to work with the Telnet Client program to access other remote system resources on the network. As shown below, after setting up a connection with Switch A through the terminal emulation program or Telnet, you can log in the Switch B for management and configuration using the telnet command.

Qtech's telnet program supports IPV4 and IPV6 addresses. The telnet server can receive IPV4 and IPV6 telnet connection requests. The telnet client can send connection requests to an IPV4 or IPV6 host.

Figure 1



4.1.9.1. *Using Telnet Client*

You can log in to a remote device by using the **telnet** command on the switch.

Command	Function
Qtech# **telnet** *host* [*port*] [/**source** {**ip** *A.B.C.D* **ipv6** *X:X:X::X* | **interface** *interface-name*}] [/**vrf** *vrf-name*]	Logs in to a remote device via Telnet. host may be an IPv4 or IPv6 host name or an IPv4 or IPv6 address. For optional parameters, refer to relevant Telnet command section in Basic Configuration Management Command.

The following example shows how to establish a Telnet session and manage the remote device with the IP address 192.168.65.119:

```
Qtech# telnet 192.168.65.119     //Establish the telnet session to the remote device
Trying 192.168.65.119 ... Open
User Access Verification         //Enter into the login interface of the remote device
Password:
```

The following example shows how to establish a Telnet session and manage the remote device with the IPv6 address 2AAA:BBBB::CCCC:

```
Qtech# telnet 2AAA:BBBB::CCCC    //Establish the telnet session to the remote device
```

```
Trying 2AAA:BBBB::CCCC ... Open
User Access Verification        //Enter into the login interface of the remote device
Password:
```

4.1.9.2. *Using Telnet Server*

Use the following command to enable the Telnet server service for network devices:

Command	Function
Qtech(config)# **enable service telnet-server**	Enables the Telnet server service. This command enables IPv4 and IPv6 services concurrently.

4.1.10. Setting Connection Timeout

4.1.10.1. *Overview*

You can control the connections of a device (including the accepted connections and sessions between the device and a remote terminal) by configuring the connection timeout for the device. When the idle time exceeds the set value and no input or output is found, this connection will be released.

4.1.10.2. *Connection Timeout*

When there is no information running through an accepted connection within a specified time, the server will release this connection.

Our products provide commands for you to configure the connection timeout in line configuration mode.

Command	Function
Qtech(Config-line)#**exec-timeout** *minutes* [*seconds*]	Configures the timeout for the accepted connection. When the configured time is due and there is no input, this connection will be released. *minutes*: timeout in minutes; *seconds*: timeout in seconds.

You can cancel the connection timeout by using the **no exec-timeout** command in line configuration mode.

```
Qtech# configure terminal //Enter global configuration mode.
Qtech# line vty 0         //Enter the line configuration mode
Qtech(config-line)#exec-timeout 20      //Set the timeout to 20min
```

4.1.10.3. *Session Timeout*

When there is no input for the session established with a remote terminal over the current line within the specified time, the session will be released and the remote terminal becomes idle.

RGOS provides commands in line configuration mode to configure the timeout for sessions with a remote terminal.

Command	Function
Qtech(Config-line)#**session-timeout** *minutes* [**output**]	Configures the timeout for the session set up with the remote terminal over the line. If there is no input within the specified time, this session will be released. *minutes*: timeout in minutes; **output**: Determines whether the session has expired using output data.

You can remove the timeout setting for the session set up with the remote terminal by using the **no exec-timeout** command in the line configuration mode.

```
Qtech# configure terminal //Enter global configuration mode.
Qtech(config)# line vty 0              //Enter the line configuration mode
Qtech(config-line)# session-timeout 20  //Set the session timeout to 20min
```

4.1.11. Executing the Commands for Executable Batch Files

During the process of system management, it is sometimes necessary to enter multiple configuration commands to manage a function. It takes a long time to enter all the commands on CLI, causing errors. To solve this problem, you can include all the commands into a batch file by taking configuration steps. Then, you can execute the batch file for configuration when necessary.

Command	Function
Qtech# **execute** {[**flash**:] *filename*}	Executes a batch file.

For example, the batch file line_rcms_script.text enables the reversed Telnet function on all the asynchronous interfaces as shown below:

```
configure terminal
line tty 1 16
transport input all
no exec
end
```

Result:

```
Qtech# execute flash:line_rcms_script.text
executing script file line_rcms_script.text ......
executing done
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# line vty 1 16
Qtech(config-line)# transport input all
Qtech(config-line)# no exec
Qtech(config-line)# end
```

Note

The file name and contents of a batch file can be specified. You can send an edited batch file to the flash memory of the network device in TFTP mode. The contents of the batch file will synchronize the input completely. Hence, it is necessary to edit the contents of the batch file in the sequence that CIL commands are configured. For some interactive commands, it is necessary to write corresponding response information into the batch file to ensure that the commands can be executed normally.

Caution

Files exceeding 128 KB may cause batch processing to fail. For batch processing, split a large file into a number of files, each of which is smaller than 128 KB.

4.1.12. Setting a Service Switch

During operations, you can adjust services dynamically to enable or disable specified services (SNMP Server/SSH Server/Telnet Server/Web Server).

Command	Function
Qtech(Config)# **enable service snmp-agent**	Enables SNMP Server.
Qtech(Config)# **enable service ssh-sesrver**	Enables SSH Server.
Qtech(Config)# **enable service telnet-server**	Enables Telnet Server.
Qtech(Config)# **enable service web-server**	Enables http and https servers.

In configuration mode, you can use the **no enable service** command to disable corresponding services.

```
Qtech# configure terminal                     //Enter global configuration mode.
Qtech(config)# enable service ssh-server      //Enable SSH Server
```

To enable http service only, use the following command:

```
Qtech(config)# enable service web-server http
```

To enable https service only, use the following command:

```
Qtech(config)# enable service web-server https
```

Note

The **enable service web-server** command can be followed by three optional keywords:
enable service web-server [*http* | *https* | *all*]
If the command is followed by no keyword or by all, the command enables http and https services. Followed by http, the command enables http service only. Followed by https, the command enables https service only.
This command and related HTTP service commands discussed later do not necessarily enable you to access the web management page through the browser. These commands only enable the HTTP service and provide an HTTP access channel. To access the web management page, upload a compressed web management work package in upd format to the flash memory of your device. These commands are not designed only for web management. Instead, they support HTTP detection, redirection, and flash file download on the device.

Caution

For the HTTP server to work after the HTTP service is enabled, store the server certificate and private key in the root directory of the file system on the device. Name the server certificate httpd_cert.crt and private key file httpd_key.pem, and upload these files to the root directory through a TFTP server.

4.1.13.Setting HTTP Parameters

When using the integrated Web for management, you can adjust HTTP parameters, and specify service ports or login authentication methods.

Command	Function
Qtech(Config)# **ip http port** *number*	Specifies HTTP service port, 80 by default.
Qtech(Config)# **ip http authentication** {**enable** | **local**}	Sets Web login authentication method, which is enable by default. **enable**: Uses the password set by the enable password or enable secret command for authentication, where the password must be 15 levels. **local**: Uses the username and password set by the username command for authentication, where the user must be bound with 15-level access rights.

In configuration mode, you can use the **no** form of the command to restore the setting to the default value. The following example enables the HTTP Server, sets the service port to 8080, and uses the local username for login authentication.

```
Qtech# configure terminal                     //Enter global configuration mode.
Qtech(config)# enable service web-server http        //Enable http Server
Qtech(config)# username name password pass //Set local user
Qtech(config)# username name privilege 15     //Bind user right
Qtech(config)# ip http port 8080                 //Set service port
Qtech(config)# ip http authentication local //Set authentication method
```

Use the following command to configure an HTTPS service port.

Command	Function
Qtech(Config)# **ip http secure-port** *number*	Specifies the HTTP service port. (default:443)

In configuration mode, you can use the **no** form of the command to restore the setting to the default value. The following example enables the HTTP Server and sets the service port to 4443.

```
Qtech# configure terminal                     //Enter global configuration mode.
Qtech(config)# enable service web-server https//Enable https Server
Qtech(config)# ip http secure-port 4443
```

Use the following command to verify the status of WEB server.

```
Qtech# show web-server status
http server status : enabled
http server port : 8080
https server status:  enabled
https server port: 4443
```

⚠ Caution

Avoid configuring http and https service ports to the same value. If https service is enabled after http service, and the port is configured to the same port as http service, you can only access https service through this port, and http service will be blocked temporarily until https service port is changed or the service is disabled.

4.1.14.Setting Multi-boot Function

4.1.14.1. Overview

By default, the device searches for the main program file and boots it in the built-in flash memory. If the main program file is damaged due to upgrade failure, formatted flash memory, for example, the device may fail to boot the system.

Some Qtech products support the multi-boot function, which enables you to boot the device using main program files from local flash memory, a removable disk (USB drive or SD card) or a remote TFTP server. When the device starts, the system boots the main programs by boot pirority in descending order until it boots successfully or all programs are filed. Multi-boot function is mandatory for some environments with higher demands on reliability and availability.

Product support	Only the QSR-2830 series of routers currently support the multi-boot function. Unless otherwise stated, this section applies to the above products.

This following examples describe how to use multi-boot for redundant backup of the main program.

4.1.14.2. Configuring the Main Boot Program

You can use the following command to configure the main boot program and specify the boot priority. The system will boot the main program based on priority in descending order with 1 being the highest and 10 being the lowest priority.

Command	Function
Qtech(Config)#**boot system** *priority* prefix:/ [directory/] filename	Sets the main boot program and specify its priority. The boot priority is in the range of 1 to 10, with 1 being the highest.

⚠ Caution

Using URL prefix to locate a file is only supported in 10.4(2) and higher versions. For details, refer to the File System Configuration Guide. Path is used to locate a file in a version lower than 10.4(2), for example, usb0:/backup/rgos.bin represents rgos.bin in the backup directory on the first USB device. flash:/rgos.bin indicates the rgos.bin file under the Flash root directory.

⚠ Caution

Supported URL prefixes vary with platforms. To show the currently supported URL prefixes, run the following command:

```
Qtech (config) # boot system 2 ?
  flash:   Boot from flash: file system
  tftp:    Boot from tftp server
  usb0:   Boot from usb0: file system
  usb1:   Boot from usb1: file system
```

Caution Multi-boot is not allowed during ISSU operations (see “Upgrading ISSU”).

By default, the bootable main program is flash:/rgos.bin with the priority of 5.

Caution Since the system uses this command in the early stage of booting, the configuration is saved in the Boot ROM rather than in the configuration file.

4.1.14.3. Specifying a File in Local Flash Memory

The following example sets the file on the local flash memory as the main program.

```
Qtech(config)# boot system 5 flash:/rgos.bin
```

Note When you specify a local file through prefix, the path following “:” must be an absolute path.

When you configure the boot system command, the system will check the validity of the main program in the local flash memory. You can configure the command successfully only when the main program meets the following requirements.

- The main program must exist.
- The main program is a legal RGOS main program.
- The main program is complete and passes CRC check.

If any requirement is unmet, the systme will dispay an error message, for example:

```
Qtech(config)# boot system 5 flash:/foo.bin
Set boot system file error:[ flash:/foo.bin] does not exist!
```

In addition, a priority can be set for more than one main program. Otherwise, the system will display an error message and print the current main program list for your selection. For instance:

```
Qtech(config)# boot system 5 flash:/rgos.bin
Qtech(config)# boot system 5 flash:/rgos_bak.bin
Set boot system file error: priority 5 has been assigned to file [ flash:/rgos.bin].
Boot system config:
=====================================================
Prio      Size                Modified  Name
---- --------- ------------------- ------------------
  1
  2
  3
  4
  5        3205120 2008-08-26 05:22:46 flash:/rgos.bin
  6
  7
  8
  9
 10
======================================================
Qtech(config)# boot system 6 flash:/rgos_bak.bin
```

4.1.14.4. Specifying a File on a Removable Storage Device

The same procedure applies for saving the main program as a file on a removable storage device and in the local flash memory. The only difference is that the system does not check the file for existence or validity when you save it on a removable storage device.

⚠ Caution

The file is not checked so that you can configure the device remotely without having to insert a USB drive containing a valid main program into the device. However, you must enter a correct filename.

Do as follows to set up the device to boot from a USB drive:

```
Qtech(config)# boot system 1 usb1:/rgos.bin
```

Note

Currently, the device cannot start a RGOS installation package earlier than 10.4 (3) from a USB drive.

4.1.14.5. Specifying a File on a Remote TFTP Server

Do as follows to set up the device to boot from a TFTP server:

```
Qtech(config)# boot system 2 tftp://192.168.7.24/rgos.bin
```

Note

Currently, the device cannot start a RGOS installation package earlier than 10.4 (3) from a TFTP server.

For the device to download the main program through the TFTP protocol during the boot process, use the boot ip command to configure a correct local IP address used for TFTP address:

Command	**Function**
Qtech(config)# **boot ip** *local-ip*	Configures a local IP address for TFTP transfer during the boot process.
Qtech(config)# **no boot ip**	Clear the boot ip configuration.

⚠ Caution

This configuration is stored in the Boot ROM rather than in a configuration file, as the system must use the configuration early in the boot process.

⚠ Caution

Ensure that the built-in flash memory contains sufficient free space for the boot file when booting from a TFTP server. During the boot process, the file is hidden in the flash memory. Clear it before the next boot.

Do as follows to configure the IP address for the device to boot up:

```
Qtech(config)# boot ip 192.168.7.11
```

If no boot ip address is specified, the device cannot load main program files from the TFTP server during the boot process due to communication failure. The following message appears on the screen:

```
Load program file: [tftp://192.168.7.24/rgos.bin]
[Failed] (Boot IP was not assigned)
Load program file: [/rgos.bin]
[OK]
Executing program, launch at: 0x00010000
```

```
......
```

4.1.14.6. Modifying the Boot Priority of Main Program

The **boot system** command can also be used to modify the boot priority of the main program. Aussme that the configured boot main program list is shown as follows:

```
Qtech# show boot system
Boot system config:
=====================================================
Prio      Size               Modified   Name
---- --------- ------------------- ------------------
  1
  2
  3
  4
  5        3205120 2008-08-26 05:22:46 flash:/rgos.bin
  6
  7
  8        3205120 2008-08-26 05:25:09 flash:/rgos_bak.bin
  9
 10
======================================================
```

To set the boot priority of flash:/rgos_bak.bin to 1, run the following command:

```
Qtech(config)# boot system 1 flash:/rgos_bak.bin
File [flash:/rgos_bak.bin] has been configured with priority 8,
Change the priority to [1]? [yes] yes
```

The result is as follows:

```
Qtech# show boot system
Boot system config:
=====================================================
Prio      Size               Modified   Name
---- --------- ------------------- ------------------
  1        3205120 2008-08-26 05:25:09 flash:/rgos_bak.bin
  2
  3
  4
  5        3205120 2008-08-26 05:22:46 flash:/rgos.bin
  6
  7
  8
  9
 10
======================================================
```

4.1.14.7. Deleting the Main Boot Program

You can use the following command to delete the main boot program.

Command	Function
Qtech(Config)# **no boot system** [*priority*]	Deletes the main boot program. The boot priority is in the range of 1 to 10. If no priority is specified, all the main boot programs will be reset.

Use the following command to delete the main program with the prioirty of 8. During the process of deletion, the system displays the corresponding main program name and prompts you for confirmation.

```
Qtech(config)# no boot system 8
Delete boot system config: [Priority: 8; File Name: flash:/rgos_bak.bin]? [no] yes
```

Use the following command to clear all the main boot programs.

```
Qtech(config)# no boot system
```

```
Clear ALL boot system config? [no] yes
```

Caution

If you have not configured the main boot program after using the **no boot system** command to clear all main boot programs, the system will automatically restore to the default setting during the next booting process (the bootable main program is flash:/rgos.bin with the priority of 5).

4.1.14.8. Showing the Configuration of Multi-boot

You can use the following command to show the configuration of multi-boot.

Command	Function
Qtech# **show boot system**	Show the configuration of the main boot program.
Qtech# **show boot ip**	Shows the local IP address used by the device during the boot process.

The local IP address for booting up the device is shown as follows:

```
Qtech# show boot ip
System boot ip: [192.168.7.11]
```

Use the following command to show the main program and its boot priority.

```
Qtech# show boot system
Boot system config:
=====================================================
Prio      Size               Modified  Name
---- --------- ------------------- ------------------
  1
  2
  3
  4
  5        3205120 2008-08-26 05:22:46 flash:/rgos.bin
  6
  7
  8        3205120 2008-08-26 05:25:09 flash:/rgos_bak.bin
  9
 10
======================================================
```

Note

The size and modification time are not shown for files on a remote TFTP server. The size and modification time are shown as N/A for such files.

Note

If the related main program does not exist when you run the **show boot system** command, the size and modification time of the file is also shown as N/A.

4.1.14.9. Configuration Example

The following example shows how the device boots up:

- The device boots from the USB drive in USB port 1 that contains a legal main program file in its root directory;
- The device boots from rgos.bin in the root directory of the built-in flash memory if no USB drive is available;
- The device will boot from the backup main program file rgos_bak.bin if rgos.bin is damaged or lost;
- The device will download and boot from a main program file from the remote TFTP server if the backup main

program file fails possibly because the built-in flash memory is formatted.

Step 1: Configure the default main program.

```
Qtech(config)# boot system 5 flash:/rgos.bin
```

Since the device is configure with the main program flash;/rgos.bin with priority of 5 during intialization, this step can be skipped.

Generally, it is recommended that you set the priority of an active main program to medium so that you can configure other main programs with a higher or lower priroity.

Step 2: configure the backup main program.

The backup main program should have a priority slightly lower than the active main program.

```
Qtech(config)# boot system 8 flash:/rgos_bak.bin
```

Step 3: Configure the name of the main program for booting from a remote TFTP server and the boot IP address.

Normally, the device is set up to boot from a remote TFTP server only when its built-in flash memory is damaged. Therefore, boot from TFTP is set to the lowest priority:

```
Qtech(config)# boot ip 192.168.7.11
Qtech(config)# boot system 10 tftp://192.168.7.24/rgos.bin
```

Step 4: Configure the name of the main program for USB boot.

Boot from a USB drive applies when a temporary version must be quickly deployed for trial run.

For a device to boot first from a USB drive, insert the USB drive that contains only the temporary software version and restart the device. To clear the temporary version, remove the USB drive and restart the device. The device will automatically boot from the main program in the built-in flash memory. Booting from a USB drive simplifies the deployment of a temporary version and shortens the downtime due to version upgrade.

Do as follows to configure the name of the main program and set up the device to boot first from a USB drive:

```
Qtech(config)# boot system 1 usb1:/rgos.bin
```

Step 5: Check the main program name and boot priority.

You can run the **show boot** system command to view confiugration.

```
Qtech# show boot system
Boot system config:
=====================================================
Prio      Size               Modified  Name
---- --------- ------------------- ------------------
  1
  2
  3
  4
  5        3205120 2008-08-26 05:22:46 flash:/rgos.bin
  6
  7
  8        3205120 2008-08-26 05:25:09 flash:/rgos_bak.bin
  9
 10              N/A                    N/A tftp://192.168.7.24/
                                         rgos.bin
======================================================
```

4.1.15. Setting Startup Configuration File

Some Qtech products can speficy a startup configuration file, which is stored in the flash memory, on a removable storage device (for example, USB drive, SD card) or remote TFTP server.

Once configured, a device can obtain a file from a specified location as the startup configuration file.

Product support	Only the QSR-2830 series of routers currently support this function. Unless otherwise stated, this section applies to the above products.

This following examples describe how to specify a startup configuration file.

4.1.15.1. Configuring the Startup Configuration File

You can use the following command to configure the startup configuration file.

Command	Function
Qtech(Config)#**boot config** *prefix:/ [directory/] filename*	Sets the startup configuration file.
Qtech(Config)#**no boot config**	Clears the startup configuration file.

Note

You can view the configuration file using the command line help, for example:

```
Qtech(config)#boot config ?
  flash:   Startup-config filename
  usb0:   Startup-config filename
  usb1:   Startup-config filename
```

The system loads a configuration file as follows:

- If the **service config** command is absent, configuration files are loaded in the following order: the startup configuration file specified by the **boot config** command, /config.text, the network startup configuration file configured by the **boot network** command, and the default factory configuration (null configuration).
- If the **service config** command is present, configuration files are loaded in the following order: the network startup configuration file configured by the **boot network** command, the startup configuration file specified by the **boot config** command, /config.text, and the default factory configuration (null configuration).
- While loading configuration files in order, the system will not load another configuration files until one configuration file is loaded successfully.

Caution

For the service config and boot config commands, refer to the following examples.
Because the system needs to use the configuration of this command in the early stage of booting, this configuration is stored in Boot ROM rather than in the configuration file.

When using the **write [memory]** command to store the startup configuration file, the system will save it as follows:

- If the **boot config** command is not used, the system stores the configuration in the flash:/config.text file in the built-in flash memory by default.
- If the **boot config** command is used to configure a startup configuration file and the file exists, the system stores the configuration in the startup configuration file.
- If the **boot config** command is used to configure a startup configuration file but the configuration file does not exist, then:
- If the device where the configuration file is located exists, the system will automatically create the specified configuration file and store it into the system configuration.
- If the device where the configuration file is located does not exist (for instance, the start cofniguration file is stored on a removable storage device such as a USB drive or SD card, but the device is not loaded when the system runs the **write [memory]** command), the system will prompt you whether to save the configuration into the default startup configuration file flash:/config.text and act as required.

The following example sets a file on the USB drive as the startup configuration file and demonstrates how to run the write command before and after removing the USB drive.

Set the file on the USB drive as the startup configuration file.

```
Qtech(config)# boot config usb1:/config.text
```

Run the **write** command before removing the USB drive to save the current configuation into the file specified by the boot config command.

```
Qtech# write
Building configuration...
Write to boot config file: [usb1:/config.text]
[OK]
```

Run the **write** command after removing the USB drive. The system will prompt you whether to save the current configuration into the default start configuraiton file /config.

```
Qtech# usb remove 1
0:1:1:38 Qtech: USB-5-USB_DISK_REMOVED: USB Device <USB Mass Storage Device> Removed!
Qtech# write
Building configuration...
Write to boot config file: [usb1:/config.text]
[Failed]
The device [usb1] does not exist, write to the default config file
[flash:/config.text]? [no] yes
Write to the default config file: [flash:/config.text]
[OK]
```

4.1.15.2. Configuring the Network Start Configuration File

You can use the following command to configure the network startup configuration file.

Command	Function
Qtech(Config)# **boot network tftp** :// *location / filename*	Configures the network startup configuration file.
Qtech(Config)#**no boot network**	Clears the network startup configuration file.

When the device starts, the system loads the configuration file as follows;

- If the service config command is absent, configuration files are loaded in the following order: the startup configuration file specified by the boot config command, /config.text, the network startup configuration file cofnigured by the boot network command, and the default factory configuration (null configuration).
- If the service config command is present, configuration files are loaded in the following order: the network startup configuration file configured by the boot network command, the startup configuration file specified by the boot config command, /config.text, and the default factory configuration (null configuration).
- While loading configuration files in order, the system will not load another configuration files until one configuration file is loaded successfully.

Caution

The system can obtain remote files through TFTP only after you run the bootip command to configure the local IP address of the device used for initiation. Otherwise, TFTP transmission may fail during initiation.

Because the system needs to use the configuration of this command in the early stage of booting, this configuration is stored in Boot ROM rather than the configuration file.

The following figure sets the boot IP address of the device and designates the network startup configuration file.

```
Qtech(config)# boot ip 192.168.7.11
Qtech(config)# boot network tftp://192.168.7.24/config.text
```

4.1.15.3. Configuring Preferrably Using the Network Start Configuration File

By default, the device loads the local startup configuration file specified by the **boot config** command. In some cases, if the device needs to use the network startup configuration file, run the **service config** command.

Command	Function
Qtech(Config)# **service config**	Enables the device to preferably load the startup configuration file from the remote network server.
Qtech(Config)#**no service config**	Disables the device to preferably load the startup configuration file from the remote network server.

This command should use in conjunction with the **boot config** and **boot network** commands.

When the device starts, the system loads the configuration file as follows;

- If the **service config** command is absent, configuration files are loaded in the following order: the startup configuration file specified by the boot config command, /config.text, the network startup configuration file configured by the boot network command, and the default factory configuration (null configuration).
- If the **service config** command is present, configuration files are loaded in the following order: the network startup configuration file configured by the boot network command, the startup configuration file specified by the boot config command, /config.text, and the default factory configuration (null configuration).
- While loading configuration files in order, the system will not load another configuration files until one configuration file is loaded successfully.

Caution

Because the system needs to use the configuration of this command in the early stage of booting, this configuration is stored in Boot ROM rather than the configuration file.

The following example loads the configuration file from a remote network server and configures the network startup configuration name.

```
Qtech(config)# service config
Qtech(config)# boot network tftp://192.168.7.24/config.text
```

4.1.15.4. Showing the Configuration of Start Configuration File

You can use the following command to show the configuration of a startup configuration file.

Command	Function
Qtech# **show boot config**	Shows the configuration of a startup configuration file.
Qtech# **show boot network**	Shows the configuration of a network startup configuration file.

The following example shows the configuration of a startup configuration file.

```
Qtech# show boot config
Boot config file: [flash:/config_main.text]
Service config: [Disabled]
```

The following example shows the configuration of a network startup configuration file.

```
Qtech# show boot network
Network config file: [tftp://192.168.7.24/config.text]
Service config: [Enabled]
```

4.1.15.5. Configuration Example

The following example sets up the device to first obtain the configuration file from a remote TFTP server and to use the backup configuration file in built-in flash memory when loading fails.

Step 1: Configure the device to first load the configuration file from a network server and configure the boot IP address.

```
Qtech(config)# service config
Qtech(config)# boot ip 192.168.7.11
```

Step 2: Configure the network startup configuration file.

```
Qtech(config)# boot network tftp://192.168.7.24/router_1.text
```

Step 3: Configure the local startup configuration file.

```
Qtech(config)# boot config flash:/router_1.text
```

Step 4: Show the configuration.

```
Qtech# show boot network
Network config file: [tftp://192.168.7.24/router_1.text]
Service config: [Enabled]
Qtech# show boot config
Boot config file: [flash:/router_1.text]
Service config: [Enabled]
```

5. CONFIGURING NETWORK COMMUNICATION DETECTION TOOLS

5.1. Ping Connectivity Test

To test the connectivity of a network, many network devices support the **Echo** protocol. The protocol sends a special packet to a specified network address and waits for a response. This allows you to evaluate the connectivity, delay and reliability of a network. The ping tool provided by RGOS can effectively help users diagnose and locate the connectivity problems in a network.

The **Ping** command runs in ordinary user mode and privileged user mode. In ordinary user mode, only basic ping functions are available. However, in privileged user mode, extended ping functions are available.

Command	Function
Qtech# **ping** [vrf *vrf-name*] [ip] [*address* [**length** *length*] [**ntimes** *times*] [data *data*][**source** *source*] [**timeout** *seconds*] [df-bit] [validate]]	Tests the network connectivity.

The basic ping function can be performed in either ordinary user mode or privileged user mode. By default, this command sends five 100-byte packets to the specified IP address. If the system receives a response within the specified time (2 seconds by default), it shows "!" . Otherwise, it shows ".". Finally, the system shows statistics. The following example shows an ordinary **ping**:

```
Qtech# ping 192.168.5.1
Sending 5, 100-byte ICMP Echoes to 192.168.5.1, timeout is 2 seconds:
< press Ctrl+C to break >
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/10 ms
```

The extended ping function can be performed in privileged user mode only. This function allows you specify the number of packets, packet length, and timeout time. As with the basic ping function, the extended ping also shows statistics. The following example shows an extended **ping**:

```
Qtech ping 192.168.5.197 length 1500 ntimes 100 data ffff source 192.168.4.190 timeout
3
Sending 100, 1000-byte ICMP Echoes to 192.168.5.197, timeout is 3 seconds:
  < press Ctrl+C to break >
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max = 2/2/3 ms
```

5.2. Ping IPv6 Connectivity Test

To test the connectivity of a network, many network devices support the **Echo** protocol. The protocol sends a special packet to a specified network address and waits for a response. This allows you to evaluate the connectivity, delay and reliability of a network. The ping tool provided by RGOS can effectively help users diagnose and locate the connectivity problems in a network.

The **Ping ipv6** command runs in ordinary user mode and privileged user mode. In ordinary user mode, only basic ping IPv6 functions are available. However, in privileged user mode, extended ping IPv6 functions are available.

Command	Function
Qtech# **ping ipv6** [*address* **[length** *length*] [**ntimes** *times*] [**data** *data*] [**source** *source*] [**timeout** *seconds*]]	Tests the network connectivity.

The basic ping function can be performed in either ordinary user mode or privileged user mode. By default, this command sends five 100-byte packets to the specified IP address. If the system receives a response within the specified time (2 seconds by default), it shows "!" . Otherwise, it shows ".". If the response does not match the request, the system shows “C” and outputs statistics. The following example shows an ordinary **ping**:

```
Qtech# ping ipv6 2000::1
Sending 5, 100-byte ICMP Echoes to 2000::1, timeout is 2 seconds:
< press Ctrl+C to break >
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/10 ms
```

The extended ping function can be performed in privileged user mode only. This function allows you specify the number of packets, packet length, and timeout time. As with the basic ping function, the extended ping also shows statistics. The following example shows an extended **ping**:

```
Qtech# ping ipv6 2000::1  length 1500 ntimes 100 data ffff source 2000::2 timeout 3
Sending 100, 1000-byte ICMP Echoes to 2000::1, timeout is 3 seconds:
  < press Ctrl+C to break >
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!
Success rate is 100 percent (100/100), round-trip min/avg/max = 2/2/3 ms
```

5.3. Traceroute Connectivity Test

The **Traceroute** command is mainly used to check the network connectivity. It show all the gateways that a packet passes through from the source address to the destination address and exactly locates the fault when the network fails.

One of the network transmission rules is that the number in the TTL field in the packet will decrease by 1 every time when a packet passes through a gateway. When the number in the TTL field is 0, the gateway will discard this packet and send an address unreachable error message back to the source. According to this rule, the execution of the **traceroute** command is as follows: At first, the source sends a packet whose TTL is 1 to the destination address. The first gateway sends an ICMP error message back, indicating that this packet cannot be forwarded for TTL timeout. Then, the first gateway re-sends the packet after the TTL domain adds 1. Likewise, the second gateway returns a TTL timeout error and the process lasts until the packet reaches the destination address. By recording every address returning the ICMP TTL timeout message, you can draw the entire path passed by the IP packet from the source address to the destination address.

The **traceroute** command can run in ordinary user mode and privileged user mode. The command format is as follows:

Command	Function
Qtech# **traceroute** [**vrf** *vrf-name* | **ip**] [*address* [**probe** *probe*] [**ttl** *minimum maximum*] [source *source*] [**timeout** *seconds*]]	Traces the path that a packet passes through.

The following are two examples that apply **traceroute**. In one example, network connectivity is good. In another example, some gateways in a network are not connected.

traceroute example where network connectivity is good:

```
Qtech# traceroute 192.168.0.10
  < press Ctrl+C to break >
Tracing the route to 61.154.22.36
1      192.168.12.1        0 msec        0 msec        0 msec
2      192.168.9.2         4 msec        4 msec   4 msec
3      192.168.9.1         8 msec        8 msec        4 msec
4      192.168.0.10        4 msec        28 msec   12 msec
```

As you can see, to access the host with an IP address of 192.168.0.10, the network packet passes throuth gateways 1 to 4 from the source address. Meanwhile, you can know the time that the network packet spennds to reach a gateway. This is very useful for network analysis.

a. **traceroute** example where some gateways in a network are not connected:

```
Qtech# traceroute 192.168.110.28
< press Ctrl+C to break >
Tracing the route to 202.108.37.42
1      192.168.12.1        0 msec        0 msec        0 msec
2      192.168.9.2         0 msec        4 msec        4 msec
3      192.168.110.1       16 msec       12 msec       16 msec
4      *  *  *
5      192.168.110.28            12 msec       28 msec       12 msec
```

As you can see, to access the host with an IP address of 192.168.110.28, the network packet passes through gateways 1 to 5 from the source address and there is failure in gateway 4.

5.4. Traceroute IPv6 Connectivity Test

The **Traceroute ipv6** command is mainly used to check the network connectivity. It shows all the gateways that a packet passes through from the source address to the destination address and exactly locates the fault when the network fails.

For network transmission rules, refer to the previous section.

The **traceroute ipv6** command can run in ordinary user mode and privileged user mode. The command format is as follows:

Command	Function
Qtech# **traceroute ipv6** [*address* [**probe** *probe*] [**ttl** *minimum maximum*] [**source** source] [**timeout** *seconds*]]	Traces the path that a packet passes through.

The following are two examples that apply **traceroute ipv6**. In one example, network connectivity is good. In another example, some gateways in a network are not connected.

traceroute ipv6 example where network connectivity is good:

```
Qtech# traceroute ipv6 3004::1
  < press Ctrl+C to break >
Tracing the route to 3004::1
1       3000::1         0 msec  0 msec  0 msec
2       3001::1         4 msec  4 msec 4 msec
3       3002::1         8 msec  8 msec  4 msec
4       3004::1         4 msec  28 msec  12 msec
```

As you can see, to access the host with an IP address of 3004::1, the network packet passes throuth gateways 1 to 4 from the source address. Meanwhile, you can know the time that the network packet spennds to reach a gateway. This is very useful for network analysis.

traceroute ipv6 example where some gateways in a network are not connected:

```
Qtech# traceroute ipv6 3004::1
  < press Ctrl+C to break >
Tracing the route to 3004::1
1       3000::1         0 msec  0 msec  0 msec
2       3001::1         4 msec  4 msec 4 msec
3       3002::1         8 msec  8 msec  4 msec
4       * * *
5       3004::1         4 msec  28 msec  12 msec
```

As you can see, to access the host with an IP address of 3004::1, the network packet passes through gateways 1 to 5 from the source address and there is failure in gateway 4.Configuring File System

6. CONFIGURING FILE SYSTEM

6.1. Understanding File System

6.1.1. Overview

The chapter describes the file system management on RGOS. The RGOS file management offers an unified cross-platform management function, providing the unified file management interface for different kinds of devices, storages and file transmission protocols.

Locally, there are many kinds of storage medias, for instance, Universal Serial BUS (USB) and FLASH, which can be distributed on different boards like primary and secondary management boards. Users can exchange files with remote devices through xModem and TFTP protocols. These functions can be realized using the same command.

Not all types of devices and file systems support all file system commands described here, because they support different types of file operations. The Help command shows the storage media and protocols supported by the file operation commands.

6.1.2. Basic Features

The file system management on a RGOS device offers an unified command interface for operations of various files on the device.

6.1.2.1. Using URL to Locate a File

The file system of RGOS uses Uniform Resource Locators (URLs) to uniformly locate files and directories in storage media on the local device or remote device. For example, you can copy a file from one place to another using the **copy** *source-url destination-url* command. The destination can be on the local device or a remote server.

URL representation varies by commands. The following sections describe URL usage:

- Locate a file on the server.
- Locate a local file.
- Description of URL prefixes

Locate a file on the server

To locate a file on the server, use the following command:

- **tftp:**[[//location]/directory]/filename

location: IP address or host name. *Path (directory and file name):* position for file transmission. For instance, the file transission directory specified by the TFTP server is C:\download, the file path specified by the device is the one under C:\download. tftp://192.168.0.1/binary/rgos.bin refers to the c:\download\binary\rgos.bin file on the TFTP server with the IP address of 192.168.0.1.

⚠ Caution

TFTP can only transmit files smaller than 32M. To transmit files larger than 32M, use the FTP protocol. Set a device as the FTP server to upload files to or download files from the server.

6.1.2.2. Locate a Local File

Use the [*prefix*]:[*directory*/]*filename* syntax to locate a local file on FLASH, USB and the FLASH of the management board of the device.

For example:

flash:/config.text: the configuration file on the local FLASH

usb0:/backup/rgos.bin.bak: the file on the first USB

slave:/rgos.bin: files under the root directory of the secondary management board

Note

Without prefix, the syntax refers to the file system type on the current path, for instance, if the current path is under the root directory of usb0, the syntax indicates usb0 if the file system does not specify prefix.

⚠ Caution

When you use a prefix to specify a local file, the path after “:” must be an absolute path.

6.1.2.3. Description of URL Prefixes

URL prefixes are used to specify file systems. Different devices and file operation commands can run different file systems. You can show the file systems supported on the device by the **show file system** command.

The following table shows the URL prefixes:

Prefix	Description
flash:	FLASH storage media, which can be used on all devices. The startup program is generally stored in the FLASH of a device when the device is delivered.
Tftp:	TFTP network server
xmodem:	Receives and sends files through xModem.
slave:	FLASH on the secondary management board of the chassis-based device
Usb0:	The first USB device
Usb1:	The second USB device
sd0:	The first SD card
sw1-m1-disk0:	Management board on the M1 slot of the chassis with switch id 1 in the VSU mode.

Prefix	Description
sw1-m2-disk0:	Management board on the M2 slot of the chassis with switch id 1 in the VSU mode.
sw2-m1-disk0:	Management board on the M1 slot of the chassis with switch id 2 in the VSU mode.
sw2-m2-disk0:	Management board on the M2 slot of the chassis with switch id 2 in the VSU mode.

Note

Different file system commands and different platforms support different types of file systems. For details, use the help information in the command line, for example:

```
Qtechr#copy ?
WORD            Copy from current file system
  flash:            Copy from flash: file system
  running-config    Copy from current system configuration
  slave:           Copy from slave: file system
  startup-config    Copy from startup configuration
  tftp:             Copy from tftp: file system
  usb0:           Copy from usb0: file system
  usb1:           Copy from usb1: file system
sd0:              Copy from sd0: file system
  xmodem:        Copy from xmodem: file system
```

Note

Given the limitation of xModem, files transmitted through xModem will be slightly larger than the real one.

Note

For chassis-based devices, the **slave:** prefix is supported and the **sw1-m1-disk0:** series prefixes are not supported in non-VSU modes, while in VSU mode, the **sw1-m1-disk0:** series prefixes are supported and the **slave:** prefix is not. In VSU mode, **copy flash:/file1 sw1-m1-disk0:/file2** and **copy sw1-m1-disk0:/file1 flash:/file2** are supported (the copy between the FLASH on the primary and secondary management boards of the VSU system), and **copy sw1-m1-disk0:/file1 sw1-m1-disk0:/file2** is also supported (the copy on the same management board: the primary or secondary management boards of the VSU system), but neither **copy sw1-m1-disk0:/file1 sw2-m2-disk0:/file2** nor **copy usb0:/file1 sw1-m1-disk0:/file2** is supported (only the combination with the **flash:** prefix is supported).

In VSU mode, you can operate file systems on the primary and secondary management boards of the VSU system using commands such as **dir sw1-m1-disk0:/**. The secondary management board on VSU master and slave chassis can only be used to increase bandwidth. Currently, you cannot operate file systems on the secondary management boards of VSU master and slave chassis on the primary management board of the VSU system.

6.1.2.4. *Showing the File System Information*

This command shows all the file systems supported on the device and their available spaces.

In privileged EXEC mode, use the following command:

```
Qtech#show file systems

File Systems:
         Size(b)         Free(b)          Type       Flags       Prefixes
    ------------    ------------     ---------     -------     ----------
*       33488896        16191488         flash          rw         flash:
               -               -         flash          rw          usb0:
               -               -         flash          rw          usb1:
               -               -         flash          rw           sd0:
               -               -         flash          rw         slave:
               -               -       network          rw          tftp:
               -               -       network          rw        xmodem:
--------------------------------------------------------------------
```

In this informatin, "*" means the active file system, **Size** means the space of the file system and **Free** means the available space.

Caution

Free means the idle status of the file system, not the size of files that can be stored. Since the file system has its own management overhead, the size of files that the system finally can store is slightly smaller than the free space.

6.1.2.5. Managing Local Files

Local files refer to ones stored in various storage media on the device, for instance, FLASH, and USB. System files such as main program, configuration files, logs and web files are stored generallly in FLASH. Some devices come with USB interfaces. Management of files on the flash disk is also local file management. For a chassis-based device with two management boards, you can manage files in the FLASH of the secondary management board with the **slave** prefix of URL.

For local files, you can:

- Copy files
- Move files
- Delete files
- Create directories
- Delete directories
- Show directories
- Show the current working path
- Modify the working path

These operations apply to slave-, USB-, or FLASH-type file systems and can copy files between these file systems.

Caution

File name is case sensitive on the FLASH- and slave- file systems. For example, abc.txt and Abc.txt are different. The file name must be entered correctly to locate the corresponding file. On USB-type file system, however, file name is not case sensitive, namely abc.txt and Abc.txt are considered the same.

Caution

Number and size of files will affect the startup and operation speed of files considerably. Storing too many or large files in FLASH will slow down startup of the device and update of the system. When the device starts for the first time, using the **dir** command will result in longer waiting time. Generally, we recommend a file system space of less than 128M. When necessary, we recommend storing a large number of files on a flash disk. We recommend clearing some old and useless files manually on a regular basis.

For chassis-based devices, a timeout failure may occur when the file system on the secondary management board (or the secondary management board of the VSU system in the VSU mode) is operated if too many files are stored on the secondary management board, the device is just started or for the first time. In such case, wait for a while according to the prompt and try again later.

Caution

Some files are vital for the system to work properly. Deleting these files will cause malfunction. These system files include:

- RCMS configuration file (/rcms_config.ini)
- Web management package (/web_management_pack.upd)
- Main program files (for multi-boot-supported devices, the main program includes all the files in the boot system configuration)

The system will automatically recognize these files and alarm you before you delete them. If you need to delete system files, the system will print WARN-level logs as below:

Qtech# **delete** *rgos.bin*
File [rgos.bin] is a system file. System may not work properly without it.
Are you sure you want to delete it? [no] yes

0:1:1:38 Qtech: FS-4-SYSTEM_FILE_DELETED: System file [rgos.bin] deleted!

Note

The file name with a path should be no more than 4096 bytes. Wildcard is not supported for file name and path.

6.1.2.6. Transmitting Files through Communication Protocols

- Transmitting files through TFTP:

You can upload and download files to and from the TFTP server.

In CLI privileged EXEC mode, use the following command to download files:

```
Qtech# copy tftp:[[//location]/directory]/filename destnation-url
```

In CLI privileged EXEC mode, use the following command to upload files:

```
Qtech# copy source-url  tftp:[[//location]/directory]/filename
```

- Transmitting files through xModem:

In CLI privileged EXEC mode, use the following command to download files:

```
Qtech# copy xmodem:  destnation-url
```

In CLI privileged EXEC mode, use the following command to upload files:

```
Qtech# copy source-url xmodem:
```

6.2. Typical Configuration Example

6.2.1. Downloading Files from TFTP Server

The following example shows how to download a.dat from directory c:\download\ of the TFTP server to the local device:

1) Run the TFTP Server on the host and select C:\download where the file to be downloaded is located.

b. Use the **ping** command to test the connection between the device and the TFTP server.

c. Log on the device, enter privileged EXEC mode and run the following command:

```
Qtech#copy tftp://192.168.201.54/a.dat flash:
Destination filename [a.dat]?
Accessing tftp://192.168.201.54/a.dat
!!!!!!!!!!!!!!!!!!!!!!!
Transmission finished, file length 343040
```

d. Run the **dir** command to show the files on the device.

```
Qtech#dir
Directory of flash:/
    Mode Link       Size                MTime Name
-------- ---- ---------- ------------------- -------------------
            1     343040 2009-01-01 02:02:59 a.dat
            1   10838016 2009-01-01 00:08:38 rgos.bin
            1        399 2009-01-01 00:01:37 config.text
---------------------------------------------------------------
3 Files (Total size 11181455 Bytes), 9 Directories.
Total 33030144 bytes (31MB) in this device, 20492288 bytes (19MB) available.
```

6.2.2. Uploading Files to TFTP Server

The following example shows how to upload a.dat to the c:\download\ of TFTP server:

1) Run the TFTP Server on the host and select directory C:\download where the file to be uploaded is located.

e. Use the **ping** command to test the connection between the device and the TFTP server.

f. Log on the device, enter privileged EXEC mode and run the following command:

```
Qtech#copy flash:/a.dat tftp://192.168.201.54/a.dat
Accessing flash:a.dat...
!!!!!!!!!!!!!!!!!!!!!!!
 Transmission finished, file length 343040
```

g. Check whether a.dat exists under C:\download on TFTP server.

6.2.3. Downloading Files through xModem

The following example shows how to download config.txt from PC to the local device through xModem:

1) Use a serial cable to connect the seiral interface of PC to the serial interface of the device.

h. Run hyperterminal of Windows to connect to the console of the device.

i. In privileged EXEC mode, use the following command to download the file:

```
Qtech# copy xmodem: flash:/config.text
```

j. In the Windows hyperterminal of local deivce, select Transmit files of Transmit menu:

k. In the pop-up dialog box, select the file to download and xModem and click Transmit. The Windows hyperterminal shows the transmission progress and packets.

l. Run the **dir** command to show the files on the device.

```
Directory of flash:/
    Mode Link      Size               MTime Name
-------- ---- --------- ------------------- ------------------
            1    343040 2009-01-01 02:02:59 a.dat
            1  10838016 2009-01-01 00:08:38 rgos.bin
            1       399 2009-01-01 00:01:37 config.text
--------------------------------------------------------------
3 Files (Total size 11181455 Bytes), 0 Directories.
Total 33030144 bytes (31MB) in this device, 20492288 bytes (19MB) available.
```

6.2.4. Uploading Files through xModem

The following example shows how to upload config.txt from the local device to C:\Documents and Settings\ju of PC through xModem:

1) Use a serial cable to connect the serial interface of PC and the serial interface of the device.

m. Run the hyperterminal of Windows to connect to the console of the device.

n. In privileged EXEC mode, use the following command to upload file:

```
Qtech# copy flash:/config.text xmodem
```

o. In the Windows hyperterminal of local deivce, select Receive files of Transmit menu:

p. In the pop-up dialog box, select the directory to save the uploaded file and xModem. Click Receive. The Windows hyperterminal prompts to set the name used to store the file. Click OK.

q. Check whether config.text exists under C:\Documents and Settings\ju on PC.

6.2.5. Moving Files from FLASH to USB Device

The following example shows how to move config.txt from FLASH to flash disk on USB0 and save it in the **backup** directory of flash disk:

```
Directory of flash:/
    Mode Link      Size               MTime      Name
-------- ---- --------- ------------------- ------------------
            1    343040 2009-01-01 02:02:59    a.dat
            1  10838016 2009-01-01 00:08:38    rgos.bin
            1       399  2009-01-01 00:01:37     config.text
--------------------------------------------------------------
3 Files (Total size 11181455 Bytes), 0 Directories.
Total 33030144 bytes (31MB) in this device, 20492288 bytes (19MB) available.
```

Enter the root directory of flash disk:

```
Qtech#cd usb0:/
```

Confirm the current path:

```
Qtech#pwd usb0:/
```

Create **backup** directory on flash disk:

```
Qtech#mkdir backup
```

Copy the file to the flash disk:

```
Qtech#copy flash:/config.text config.text
```

Check the result.

```
Qtech#dir backup
Directory of usb0:/backup
    Mode Link      Size                  MTime       Name
-------- ---- --------- ------------------- -------------------
            1       399    2009-01-01 00:01:37  config.text
--------------------------------------------------------------
Total 33030144 bytes (31MB) in this device, 20488192 bytes (19MB) available.
```

6.2.6. Moving Files from FLASH to SD Card

The following example shows how to move config.txt from FLASH to SD card and save it in the **backup** directory of SD card:

```
Directory of flash:/
    Mode Link      Size                  MTime       Name
-------- ---- --------- ------------------- -------------------
            1    343040 2009-01-01 02:02:59    a.dat
            1  10838016 2009-01-01 00:08:38    rgos.bin
            1       399  2009-01-01 00:01:37     config.text
--------------------------------------------------------------
3 Files (Total size 11181455 Bytes), 0 Directories.
Total 33030144 bytes (31MB) in this device, 20492288 bytes (19MB) available.
```

Enter the root directory of SD card:

```
Qtech#cd sd0:/
```

Confirm the current path:

```
Qtech#pwd sd0:/
```

Create **backup** directory on the SD card:

```
Qtech#mkdir backup
```

Make sure that the directory is created:

```
Qtech#dir
Directory of sd0:/
    Mode Link      Size                  MTime       Name
-------- ---- --------- ------------------- -------------------
    <DIR>   1    343040 2009-01-01 02:02:59    backup
--------------------------------------------------------------
3 Files (Total size 11181455 Bytes), 0 Directories.
Total 33030144 bytes (31MB) in this device, 20492288 bytes (19MB) available.
```

Copy the file to the SD card:

```
Qtech# copy flash:/config.text backup/config.text
```

Check the result:

```
Qtech#dir backup
Directory of sd0:/backup
    Mode Link      Size                  MTime       Name
-------- ---- --------- ------------------- -------------------
            1       399    2009-01-01 00:01:37  config.text
```

```
--------------------------------------------------------------
Total 33030144 bytes (31MB) in this device, 20488192 bytes (19MB) available.
```

6.2.7. Copying Files between USB and SD Card

The following example shows how to copy rgos_10_4.bin from flash disk to SD card:

Check the available space on the SD card:

```
Qtech#dir sd0:/
Directory of sd0:/
    Mode Link      Size                  MTime Name
-------- ---- --------- ------------------- -------------------
   <DIR>    2         0 2035-02-11 23:24:34 backup/
            1   7650112 2035-02-11 23:42:25 rgos.bin
--------------------------------------------------------------
1 Files (Total size 7650112 Bytes), 1 Directories.
Total 528482304 bytes (504MB) in this device, 475058176 bytes (453MB) available.
```

Copy the file from flash disk to SD card:

```
Qtech#copy usb0:/rgos_10_4.bin sd0:/rgos_10_4.bin
[OK 7,650,112 bytes]
```

Check the result:

```
Qtech#dir sd0:/
Directory of sd0:/
    Mode Link      Size                  MTime Name
-------- ---- --------- ------------------- -------------------
   <DIR>    2         0 2035-02-11 23:24:34 backup/
            1   7650112 2035-02-11 23:42:25 rgos.bin
            1   7650112 2035-02-11 23:47:36 rgos_10_4.bin
--------------------------------------------------------------
2 Files (Total size 15300224 Bytes), 1 Directories.
Total 528482304 bytes (504MB) in this device, 459571200 bytes (438MB) available.
```

Copy the file from SD card to flash disk:

```
Qtech#copy sd0:/rgos_10_4.bin usb0:/new_rgos.bin
[OK 7,650,112 bytes]
```

Check the result:

```
Qtech#dir usb0:/
Directory of usb0:/
    Mode Link      Size                  MTime Name
-------- ---- --------- ------------------- -------------------
            1   7650112 2035-02-11 23:49:21 new_rgos.bin
            1   7650112 2035-02-11 23:45:42 rgos_10_4.bin
--------------------------------------------------------------
2 Files (Total size 15300224 Bytes), 0 Directories.
Total 528482304 bytes (504MB) in this device, 451784704 bytes (430MB) available.
```

6.2.8. Copying Files from Primary Management Board to Secondary Management Board

The following example shows how to copy rgos_10_4.bin from primary management board to secondary management board:

Check the FLASH space on secondary management board:

```
Qtech#dir slave:/
Directory of slave:/
    Mode Link      Size                  MTime      Name
-------- ---- --------- -------------------- ------------------
             1    11014633   2016-01-01 08:00:46  rgos.bin
             1     399        2016-01-01 08:01:37  config.text
--------------------------------------------------------------
```

```
2 Files (Total size 11181455 Bytes), 0 Directories.
Total 33030144 bytes (31MB) in this device, 20488192 bytes (19MB) available.
```

Copy rgos_10_4.bin from primary management board to the secondary one:

```
Qtech#copy rgos_10.4.bin slave:/
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! [ok 10,234,345 bytes]
```

Check the result:

```
Qtech#dir slave:/
Directory of slave:/
    Mode Link      Size                   MTime      Name
-------- ---- --------- ------------------- -------------------
            1      11014633   2016-01-01 08:00:46  rgos.bin
            1      11025788   2008-01-01 08:00:46  rgos_10.4.bin
            1      399         2016-01-01 08:01:37  config.text
--------------------------------------------------------------
3 Files (Total size 11181455 Bytes), 0 Directories.
Total 33030144 bytes (31MB) in this device, 9563693 bytes (9MB) available
```

6.2.9. Deleting Directories

The following example shows how to delete a non-empty aaa directory:

Show the current directory status:

```
Qtech#dir
Directory of flash:/
    Mode Link      Size                   MTime      Name
-------- ---- --------- ------------------- -------------------
            1       11014633 2016-01-01 08:00:46  rgos.bin
 <dir>     1       0         2016-01-01 08:00:00 aaa/
            1       399       2016-01-01 08:01:37 config.text
--------------------------------------------------------------
2Files (Total size 11015032 Bytes), 1 Directories
Total 33030144 bytes (31MB) in this device, 9563693 bytes (9MB) available
```

Check whether there is any file under the directory:

```
Qtech#dir aaa
Directory of flash:/aaa
    Mode Link      Size                   MTime      Name
-------- ---- --------- ------------------- -------------------
           1           149  2016-01-01 08:01:37 backup.txt
--------------------------------------------------------------
1Files (Total size 149 Bytes), 0 Directories
Total 33030144 bytes (31MB) in this device, 9563693 bytes (9MB) available
```

Delete a non-empty directory:

```
Qtech# delete recursive aaa
```

Delete an empty directory:

```
Qtech# rmdir aaa
```

Check the result:

```
Qtech#dir
Directory of flash:/
    Mode Link      Size                   MTime      Name
-------- ---- --------- ------------------- -------------------
           1      11014633 2016-01-01 08:00:46  rgos.bin
           1      399      2016-01-01 08:01:37 config.text
--------------------------------------------------------------
```

```
2Files (Total size 11015032 Bytes), 0 Directories
Total 33030144 bytes (31MB) in this device, 9563693 bytes (9MB) available
```

7. CONFIGURING SYSLOG

7.1. Overview

During the operation of a device, there are various state changes, such as the link status up/down, and various events occurring, such as receiving abnormal messages and handling exceptions. Our product provides a mechanism to generate messages of fixed format (log message) in case of status change or event occurring. These messages can be displayed in related windows (console, VTY, etc.) or recorded in related media (memory buffer, FLASH), or sent to a group of log servers in the network for the administrators to analyze and locate problems. Meanwhile, in order to make it easy for administrators to read and manage log messages, these log messages can be labeled time stamps and serial numbers, and is graded according to the priority of log information.

7.1.1. Log Message Format

The format of the our log message is as follows:

```
<priority> seq no: timestamp sysname: %severity
%ModuleName-severity-MNEMONIC: description
```

Their meanings are as follows:

Command	Meaning
<priority>	Priority, priority value = Device value x 8 + Severity
seq no	System serial number, a six-digit integer. You can disable this information output by commands.
timestamp	Timestamp, local time by default. In the format of Mmm dd hh:mm:ss, Mmm indicates the English abbreviations of the 12 months. From January to December, they are abbreviated as: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, and Dec.
sysname	System name, you can disable this information output by commands.
ModuleName	Abbreviation of functional module name.
severity	Log severity level.
MNEMONIC	Information abbreviation
description	Information description

For example:

```
<189> 226:Mar  5 02:09:10 Qtech %SYS-5-CONFIG_I: Configured from console by console
```

⚠ Caution

The priority field is not attached to the log messages that are printed in the user window. It only appears in the log messages that are sent to the syslog server.

7.2. Log Configuration

7.2.1. Log Switch

The log switch is turned on by default. If it is turned off, the device will not print log information in the user window, or send log information to the syslog server, or record the log information in the related media (memory buffer, flash).

To turn on or off the log switch, run the following command in global configuration mode:

Command	Function
Qtech(config)# **logging on**	Turns on the log switch
Qtech(config)# **no on**	Turns off the log switch

⚠ Caution

Do not turn off the log switch in general case. If it prints too much information, you can reduce it by setting different displaying levels for device log information.

7.2.2. Configuring the Device for Displaying the Log Information

When the log switch is turned on, the log information will be displayed on the console and also sent to different displaying devices. To configure different displaying devices to receive logs, run the following commands in global configuration mode or privileged user level:

Command	Function
Qtech(config)# **buffered** [*buffer-size* | *level*]	Records log in memory buffer
Qtech# **termninal monitor**	Allows log to be displayed on VTY window
Qtech(config)# **logging server** *host*	Sends log information to the syslog sever on the network
Qtech(config)# **logging file flash:***filename* [*max-file-size*] [*level*]	Records log on extended FLASH. This command creates a file based on the specified file name on the FLASH to store logs. The file size increases with the log size, but its upper limit is subject to the configured max-file-size.

Buffered will record log information in the memory buffer. The memory buffer for log is used in recycled manner. That is, when it is full, the oldest information will be overwritten. To show the log information in the memory buffer, run the **show logging** command at privileged user level. To clear the log information in the memory buffer, run the **clear logging** command at privileged user level.

Terminal monitor allows log information to be displayed on the current VTY (such as the telnet window).

Logging server host specifies the address of the syslog server that will receive the log information. Our product allows the configuration of at most 5 syslog servers. The log information will be sent to all the syslog servers at the same time. The configuration of the **logging host** command has the same results.

Caution

To send the log information to the syslog server, it is required to turn on the timestamp switch or serial number switch of the log information. Otherwise, log information will not be sent to the syslog server.

Logging file flash: Records log information in FLASH. The filename for log shall not have any extension to indicate the file type. The extension of the log file is fixed as txt. Any configuration of extension for the filename will be refused.

More flash: The **filename** command shows the contents of the log file in the flash.

Caution

Some devices support extended FLASH. If the device has extended FLASH, the log information will be recorded there. If the device has no extended FLASH, the log information will be recorded in the serial FLASH.

7.2.3. Enabling the Log Timestamp Switch of Log Information

To add or delete timestamp in log information, run the following command in global configuration mode:

Command	Function
Qtech(config)# **service timestamps** [*message-type* [**uptime** | **datetime** [**msec**] [**year**]]]	Enables the timestamp in the log information
Qtech(config)# **no service timestamps** [*message-type*]	Disables the timestamp in the log information

The timestamp are available in two formats: device uptime and device datetime. Select the type of timestamp as required.

Message type: log or debug. The "log" type means the log information with severity levels 0-6. The "debug" type means that with severity level 7.

Caution

If the current device has no RTC, the configured time is invalid, and the device automatically uses the startup time as the timestamp for the log information. If the device has an RTC, the device time is used as the timestamp for the log information by default.

7.2.4. Enabling System Name Switches in Log System

By default, the system name is not included in the log information. To add or remove the system name in the log information, run the following commands in global configuration mode.

Command	Function
Qtech(config)# **no service sysname**	Removes the system name in the log message.
Qtech(config)# **service sysname**	Adds a system name to the log message.

7.2.5. Enabling Log Statistics

By default, the log statistics function is disabled. To enable or disable the log statistics function, run the following commands in global configuration mode.

Command	Function
Qtech(config)# **no logging count**	Disables the log statistics function and delete the statistics about the log information.
Qtech(config)# **logging count**	Enables the log statistics function.

Qtech# show logging count

```
Module Name    Message Name      Sev Occur   Last Time
================================================================
LINEPROTO      UPDOWN              5    2        Aug 20 01:41:19
-------------------------------------------------------------
LINEPROTO TOTAL                            2
LINK           CHANGED              5    1        Aug 20 01:41:19
-------------------------------------------------------------
LINK TOTAL                                  1
SYS            CONFIG_I             5    1        Aug 20 01:40:55
-------------------------------------------------------------
SYS TOTAL                                   1
```

Qtech #**config**

```
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech (config)#no logging count
Qtech (config)#end
Qtech #show logging count
Module Name    Message Name             Sev Occur   Last Time
===========================================================
```

7.2.6. Enabling the Serial Number Switch of Log Information

By default, the log information has no serial number. To add or delete the serial number in log information, run the following command in global configuration mode:

Command	Function
Qtech(config)# **no service sequence-numbers**	Deletes the serial number in the log messages.
Qtech(config)# **service sequence-numbers**	Adds a serial number to the log messages.

Note

The log serial number is a long integer, which increases in ascending order when a log is added. However, since only five digits of the serial number is displayed, when it increases from 1 to 100000 or reaches 2^32, a turnover occurs, that is the serial number is displayed from 00000 again.

7.2.7. Configuring Synchronization Between User Input and Log Output

By default, user input is asynchronous with log output. User input is interrupted if the log is output when the user is keying in characters. As following shows, the status of FastEthernet 0/12 changes and a log is printed after the user entered **vlan**, so that the user forgot which character he was entering previously, affecting the coherence of command entering.

```
Qtech(config)#vlan Aug 20 16:46:49 %LINK-5-CHANGED: Interface FastEthernet 0/12,
changed state to down
Aug 20 16:46:49 %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet 0/12,
changed state to DOWN
% Incomplete command.
```

While after the synchronization function is configured, the contents that the user entered previously will be displayed even though a log is printed when the user is entering a command, ensuring integrity and coherence. As following shows, the status of FastEthernet 0/1 changes and a log is printed after the user entered **vlan**, but the log module automatically prints **vlan** after the log is printed for the user to continue.

```
Qtech(config)#vlan
*Aug 20 10:05:19: %LINK-5-CHANGED: Interface GigabitEthernet 0/1, changed state to up
*Aug 20 10:05:19: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet 0/1,
changed state to up
Qtech(config)#vlan
To configure synchronization between user input and log output, run the following
commands in line configuration mode:
```

Command	Function
Qtech(config-line)# **logging synchronous**	Sets synchronization between user input and log output.
Qtech(config)# **no logging synchronous**	Removes synchronization between user input and log output.

7.2.8. Configuring Log Rate Limit

By default, log rate is not limited. However, when there are massive logs, no log rate limit will cause burden on the system. To configure log rate limit, run the following commands in global configuration mode:

Command	Function
Qtech(config)# **logging rate-limit** *number*	Sets log rate limit on the log information.
Qtech(config)# **no logging rate-limit**	Removes the setting of log rate limit.

7.2.9. Configuring the Log Information Displaying Level

Users can set the severity level of log information that is allowed to be displayed to view the log information of a specific severity level.

To configure the log information displaying level, run the following commands in global configuration mode:

Command	Function
Qtech(config)# **logging console** [*level*]	Sets the level of log information that is allowed to be displayed on the console.
Qtech(config)# **logging monitor** [*level*]	Sets the level of log information that is allowed to be displayed on the VTY window (such as telnet window).
Qtech(config)# **logging buffered** [*buffer-size*] [*level*]	Sets the level of log information that is allowed to be recorded in memory buffer.
Qtech(config)# **logging file flash:***filename* [*max-file-size*] [*level*]	Sets the level of log information that is allowed to be recorded in extended flash.
Qtech(config)# **logging trap** [*level*]	Sets the level of log information that is allowed to be sent to syslog server.

The log information of Qtechs products is classified into the following 8 levels:

Level Keyword	Level	Description
Emergencies	0	Emergency case, system cannot run normally
Alerts	1	Problems that need immediate remedy
Critical	2	Critical conditions
Errors	3	Error message
Warnings	4	Alarm information
Notifications	5	Information that is normal but needs attention
Informational	6	Descriptive information
Debugging	7	Debugging messages

Lower value indicates higher level. That is, level 0 indicates the information of the highest level.

When the level of log information that can be displayed is set for the specified device, the log information that is at or below the set level will be displayed. For example, after the **logging console *6*** command is executed, all log information at or below level 6 will be displayed on the console.

The log information that is allowed to be displayed on the console is at level 7 by default.

The log information that is allowed to be displayed on the VTY window is at level 7 by default.

The log information that is allowed to be sent to the syslog server is at level 6 by default.

The log information that is allowed to be recorded in the memory buffer is at level 7 by default.

The log information that is allowed to be recorded in the extended flash is at level 6 by default.

You can use the **show logging** command in privileged mode to show the level of log information allowed to be displayed on different devices.

7.2.10. Configuring the Device Value of the Log Information

The device value is one of the parts that form the priority field in the messages sent to the syslog server, indicating the type of device that generates the information.

To configure the log information device value, run the following command in global configuration mode:

Command	Function
Qtech(config)# **logging facility** *facility-type*	Configures the device value of the log information.
Qtech(config)# **no logging facility** *facility-type*	Restores the device value of the log information to the default value.

The meanings of various device values are described as below:

```
Numerical Code        Facility
           0             kernel messages
           1             user-level messages
           2             mail system
           3             system daemons
           4             security/authorization messages
           5             messages generated internally by syslogd
           6             line printer subsystem
           7             network news subsystem
           8             UUCP subsystem
           9             clock daemon
          10             security/authorization messages
          11             FTP daemon
          12             NTP subsystem
          13             log audit
          14             log alert
          15             clock daemon
          16             local use 0  (local0)
          17             local use 1  (local1)
          18             local use 2  (local2)
          19             local use 3  (local3)
          20             local use 4  (local4)
          21             local use 5  (local5)
          22             local use 6  (local6)
          23             local use 7  (local7)
```

The default device value of Qtech products is 23.

7.2.11. Configuring the Source Address of Log Messages

By default, the source address of the log messages sent to the syslog server is the address of the port that sends the messages. You can fix the source address for all log messages through commands.

It is possible to directly set the source IP address of the log messages or the source port of the log messages.

To configure the source address of the log messages, run the following command in global configuration mode:

Command	Function
Qtech(config)# **logging source interface** *interface-type*	Configures the source port of log information.

interface-number	
Qtech(config)# **logging source** {**ip** *ip-address* | **ipv6** *ipv6-address*}	Configures the source IP address of log messages.

Note

If the configured source IP address of the log message is not configured on any interface of the device, the source IP address of the log message is this inexistent address. However, it is not recommended to perform such configuration in actual use.

7.2.12. Setting and Sending User Log

Command	Function
Qtech(config)# **logging userinfo**	Sets user login/logoff log.
Qtech(config)# **logging userinfo command-log**	Send a log when a configuration command is executed.

7.3. Log Monitoring

To monitor log information, run the following commands in privileged user mode:

Command	Function
Qtech# **show logging**	Views the log messages in memory buffer as well as the log statistics.
Qtech# **show logging count**	Views the statistics of logs in every module.
Qtech# **clear logging**	Clears the log messages in the memory buffer.
Qtech# **more flash:***filename*	Views the log files in the extended flash.

Caution

The format of the timestamp in the output result of the **show logging count** command is the format in the latest log output.

7.3.1. Examples of Log Configurations

Here is a typical example to enable the logging function. Connect the device to the log server, whose IP address is 192.168.200.2. Perform the following configuration to make all logs carry timestamps and allow logs of all levels to be sent to the log server:

```
Qtech(config)# service timestamps debug datetime   //Enable debug information
timestamp, in date format
Qtech(config)# service timestamps log datetime      //Enable log information
timestamp, in date format
Qtech(config)# logging 192.168.200.2          //Specify the syslog server address
Qtech(config)# logging trap debugging   //The log information of all levels will be
sent to the syslog server
Qtech(config)# end
```

8. CONFIGURING DEVICE FAULT MANAGEMENT

8.1. Overview of Device Fault Management Module

8.1.1. Purpose

The device fault management module manages device faults, which generates alarms to alert router faults, protects devices against exceptions and adds some methods of preventing faults, such as displaying working status of various basic hardware devices, to enhance safe router running level.

8.1.2. Requirements and Notes

Hardware Requirements

Because fault alarms are closely related to hardware, many functions of the device fault management module are directly related to hardware. For example, hardware must support the detection of power voltage.

Caution

The device fault management module version 1.0 (DFM1.0) cannot display power voltage, fan performance and operating inlet temperature, and fails to allow MIB to search for this information.

System Support

Not all devices support alarm generation by interruption in the case of a device fault. As a result, scheduled detection is required. Here, the detection results obtained last time are used as displayed information and detection interval is set to five seconds that cannot be modified by users.

Running Mode

At present, all configuration of the device fault management module is carried out in privilege mode. As a result, to run the commands mentioned in this document, enter privilege mode first.

8.2. Checking Status Information

The command tree of status displaying of the total fault management:

Command	Function
show environment [alarms | all | fans | hardware | powers]	Displays environment of the managed faulty device.

8.2.1. Displaying Exception Alarm

Command	Function
show environment alarms	Displays information about alarm processing. For example, fans should be checked in the case of excessively high temperature

The following information is displayed for this command:

```
Qtech# show environment alarms
Warning!!!
Power supplies have been changed since the router start, please check them
Warning!!!
Fans have been changed since the router start, please check them.
Warning!!!
Temperature is high, please check powers and fans.
Qtech#
```

8.2.2. Displaying Operating Temperature

Command	Function
show temperature	Displays temperature of the current operating environment, namely temperature inside the chassis. Currently, inlet temperature cannot be detected.

The following information is displayed for this command:

```
Qtech#show temperature
Device     Temperature(C)
------     -----------
 CM            43
```

8.2.3. Displaying Fan Information

Command	Function
show environment fans	Displays running situations and status information on one or multiple fans, including whether fans run normally and the number of fans. Currently, performance detection is not supported.

The following information is displayed for this command:

```
Qtech# show environment fans
```

```
Environmental status update at 11:31:37 Jan 9, 1944.
Data is 13 second old, refresh in 20 second(s).
Fans working status:
Fan 0 is on.
Fan 1 is on.
Fan 2 is on.
Fan 3 is on.
Fan 4 is on.
Fan 5 is on.
Fan 6 is on.
Fan 7 is on.
```

8.2.4. Displaying Power Supply Information

Command	Function
show environment powers	Displays the current status information on the power supply, including rated operating voltage, the number of power supplies, and whether each power supply works normally. At present, the current operating voltage and threshold cannot be detected.

The following information is displayed for this command:

```
Qtech# show environment powers
Environmental status update at 11:28:50 Jan 9, 1944.
Data is 10 second old, refresh in 20 second(s).
Power Supplies:
Power supply 1 is present. Unit is on.
Power supply 2 is present. Unit is on.
Power supply 3 is present. Unit is on.
```

8.2.5. Displaying Information Related to Hardware

Command	Function
show environment hardware	Displays the current status information on the hardware, including CPU name and speed.

The following information is displayed for this command:

```
Qtech#show environment hardware
  Environmental status update at 16:25:26 2011-01-20.
  Data is 13 second old, refresh in 20 second(s).
  Hardware:
        CPU name: BCM1250.
        CPU Speed : 800M
```

8.2.6. Displaying All Information on Fault Management

Command	Function
show environment all	Displays device status information in the current device fault management.

The following information is displayed for this command:

```
Qtech#show environment all
  Environmental status update at 16:26:46 2011-01-20.
  Data is 18 second old, refresh in 20 second(s).
  Power Supplies:
        Power supply 1 is not present. Unit is off.
        Power supply 2 is present. Unit is on.
        Power supply 3 is not present. Unit is off.
  Fans working status:
        Fan 1 is on.
        Fan 2 is on.
        Fan 3 is on.
        Fan 4 is on.
        Fan 5 is on.
        Fan 6 is on.
Hardware:
        CPU name: BCM1250.
        CPU Speed : 800M
```

9. CONFIGURING SNMP

9.1. SNMP Overview

9.1.1. Introduction

As the abbreviation of Simple Network Management Protocol, SNMP has been a network management standard (RFC1157) since the August, 1988. So far, the SNMP becomes the actual network management standard for the support from many manufacturers. It is applicable to the situation of interconnecting multiple systems from different manufacturers. Administrators can use the SNMP protocol to query information, configure network, locate failure and plan capacity for the nodes on the network. Network supervision and administration are the basic function of the SNMP protocol.

As a protocol in the application layer, the SNMP protocol works in the client/server mode, including three parts as follows:

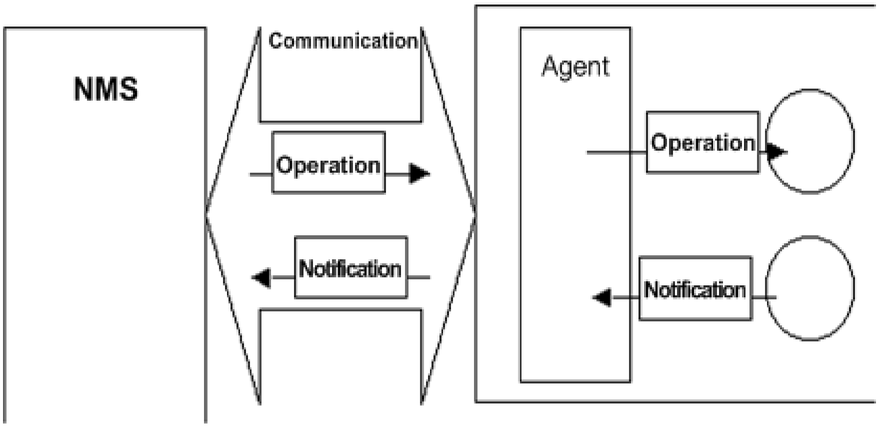
- SNMP network manager
- SNMP agent
- MIB (management information base)

The SNMP network manager, also referred to as NMS (Network Management System), is a system to control and monitor the network using the SNMP protocol.

The SNMP Agent is the software running on the managed devices. It receives, processes and responds the monitoring and controlling messages from the NMS, and also sends some messages to the NMS.

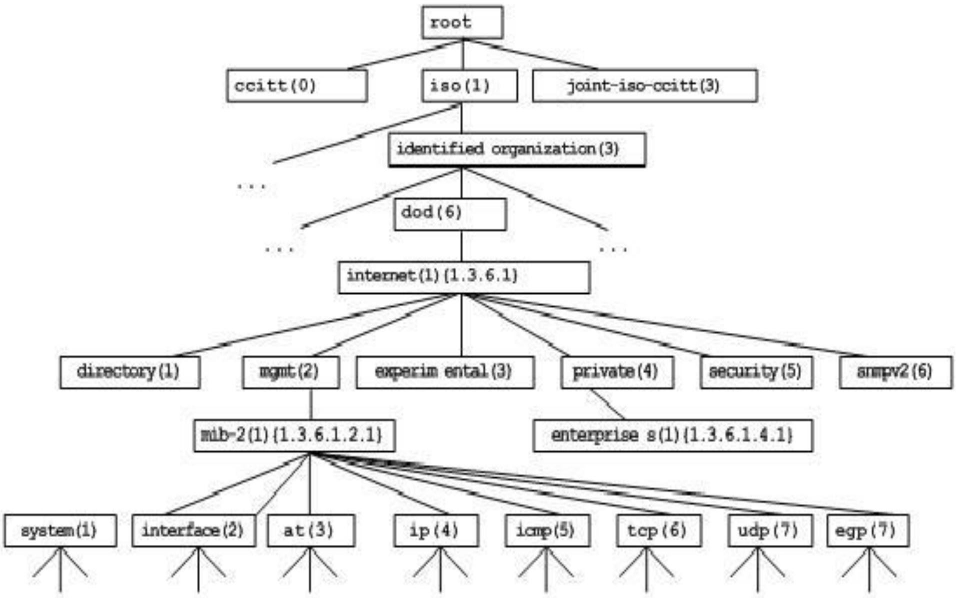
The relationship between the NMS and the SNMP Agent can be indicated as follows:

Figure 1 Relationship between the NMS and the SNMP Agent



The MIB (Management Information Base) is a virtual information base for network management. There are large volumes of information for the managed network equipment. In order to uniquely identify a specific management unit in the SNMP message, the tree-type hierarchy is used to by the MIB to describe the management units in the network management equipment. The node in the tree indicates a specific management unit. Take the following figure of MIB as an example to name the objectives in the tree. To identify a specific management unit **system** in the network equipment uniquely, a series of numbers can be used. For example, the number string {1.3.6.1.2.1.1} is the object identifier of a management unit, so the MIB is the set of object identifiers in the network equipment.

Figure 2 Tree-type MIB hierarchy



9.1.2. SNMP Versions

This software supports these SNMP versions:

- SNMPv1: The first formal version of the Simple Network Management Protocol, which is defined in RFC1157.
- SNMPv2C: Community-based Administrative Framework for SNMPv2, an experimental Internet protocol defined in RFC1901.
- SNMPv3: Offers the following security features by authenticating and encrypting packets:

r. Ensure that the data are not tampered during transmission.

s. Ensure that the data come from a valid data source.

t. Encrypt packets to ensure the data confidentiality.

Both the SNMPv1 and SNMPv2C use a community-based security framework. They restrict administrator’s operations on the MIB by defining the host IP addresses and community string.

With the GetBulk retrieval mechanism, SNMPv2C sends more detailed error information type to the management station. GetBulk allows you to obtain all the information or a great volume of data from the table at a time, and thus reducing the times of request and response. Moverover, SNMPv2C improves the capability of handing errors, including expanding error codes to distinguish different kinds of errors, which are represented by one error code in SNMPv1. Now, error types can be distinguished by error codes. Since there may be the management workstations supporting SNMPv1 and SNMPv2C in a network, the SNMP agent must be able to recognize both SNMPv1 and SNMPv2C messages, and return the corresponding version of messages.

9.1.3. SNMP Management Operations

For the information exchange between the NMS and the SNMP Agent, six types of operations are defined:

u. Get-request: The NMS gets one or more parameter values from the SNMP Agent.

v. Get-next-request: The NMS gets the next parameter value of one or more parameters from the SNMP Agent.

w. Get-bulk: The NMS gets a bulk of parameter values from the SNMP Agent.

x. Set-request: The NMS sets one or more parameter values for the SNMP Agent.

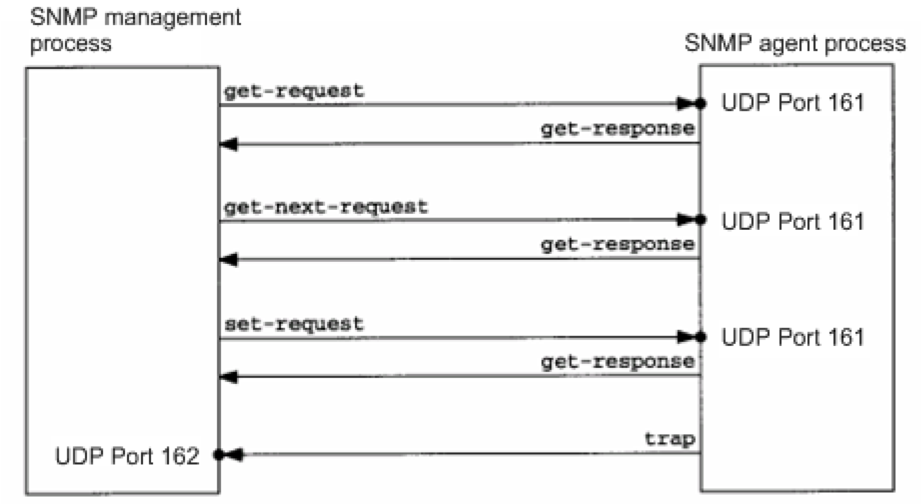
y. Get-response: The SNMP Agent returns one or more parameter values, the response of the SNMP Agent to any of the above 3 operations of the NMS.

z. Trap: The SNMP Agent proactively sends messages to notify the NMS that some event will occur.

The first four messages are sent from the NMS to the SNMP Agent, and the last two messages are sent from the

SNMP Agent to the NMS (Note: SNMPv1 does not support the Get-bulk operation). These operations are described in the following figure:

Figure 3 Message types in SNMP



NMS sends messages to the SNMP Agent in the first three operations and the SNMP Agent responds a message through the UDP port 161. However, the SNMP Agent sends a message in the Trap operation through the UDP port 162.

9.1.4. SNMP Security

Both SNMPv1 and SNMPv2 use the community string to check whether the management workstation is entitled to use MIB objects. In order to manage devices, the community string of NMS must be identical to a community string defined in the devices.

A community string features:

- Read-only: Authorized management workstations are entitled to read all the variables in the MIB.
- Read-write: Authorized management workstations are entitled to read and write all the variables in the MIB.

Based on SNMPv2, SNMPv3 can determine a security mechanism for processing data by security model and security level. There are three types of security models: SNMPv1, SNMPv2C and SNMPv3.

The table below describes the supported security models and security levels.

Model	Level	Authentication	Encryption	Description
SNMPv1	noAuthNoPriv	Community string	None	Ensures the data validity through community string.
SNMPv2c	noAuthNoPriv	Community string	None	Ensures the data validity through community string.
SNMPv3	noAuthNoPriv	User name	None	Ensures the data validity through user name.
SNMPv3	authNoPriv	MD5 or SHA	None	Provides HMAC-MD5 or HMAC-SHA-based authentication mechanism.
SNMPv3	authPriv	MD5 or SHA	DES	Provides HMAC-MD5 or HMAC-SHA-based authentication mechanism and CBC-DES-based encryption mechanism.

9.1.5. SNMP Engine ID

The engine ID is designed to identify a SNMP engine uniquely. Every SNMP entity contains a SNMP engine, a SNMP engine ID identifies a SNMP entity in a management domain. So every SNMPV3 entity has a unique identifier named SNMP Engine ID.

The SNMP Engine ID is an octet string of 5 to 32 bytes, which is defined in RFC3411:

- The first four bytes indicate the private enterprise number of an enterprise (assigned by IANA) in hex system.
- The fifth byte indicates how to identify the rest bytes.

0: Reserved

1: The following 4 bytes indicate an IPv4 address.

2: The following 16 bytes indicate an IPv6 address.

3: The following 6 bytes indicate an MAC address

4: Texts of up to 27 bytes defined by manufacturers

5: A hexadecimal value of up to 27 bytes defined by manufacturers

6-127: Reserved

128-255: In the format specified by manufacturers.

9.2. SNMP Configuration

The SNMP configuration is performed in global configuration mode on network devices. To configure SNMP, enter the global configuration mode.

9.2.1. Setting the Community String and Access Authority

SNMPv1 and SNMPv2C adopt community string-based security scheme. The SNMP Agent supports only the management operations from the management workstations of the same community string. The SNMP messages without matching the community string will be discarded. The community string serves as the password between the NMS and the SNMP Agent.

- Configure an ACL rule to allow the NMS of the specified IP address to manage devices.
- Set the community's operation permission,: ReadOnly or ReadWrite.
- Specify a view for view-based management. By default, no view is configured. That is, the management workstation is allowed to access to all MIB objects.
- Indicate the IP address of the NMS who can use this community string. If it is not indicated, any NMS can use this community string. By default, any NMS can use this community string.

To configure the SNMP community string, run the following command in global configuration mode:

	Command	Function
Step 1	Qtech(config)# **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**host** *host-ip*] [**ipv6** *ipv6-aclname*][*aclnum* | *aclname*]	Sets the community string and its right.

One or more coummnity strings can be specified for the NMS of different rights. To remove the community name and its right, run the **no snmp-server community** *string* command in global configuration mode.

9.2.2. Configuring MIB Views and Groups

With view-based access control model, you can determine whether the object of a management operation is in a view or not. Only the management objects in a view are allowed to be accessed. For access control, generally some users are associated with a group and then the group is associated with a view. The users in a group have the same access right.

- Set an inclusion view and an exclusion view.
- Set a Read-only view and a Read-write view for a group.
- Set security levels, whether to authenticate, and whether to encrypt for SNMPv3 users.

To configure the MIB views and groups, run the following commands in global configuration mode:

	Command	Function
Step 1	Qtech(config)# **snmp-server view** *view-name oid-tree* **{include | exclude}**	Creates a MIB view to include or exclude associated MIB objects.
Step 2	Qtech(config)# **snmp-server group** *groupname* {**v1 | v2c |v3 {auth | noauth | priv**}} [read *readview*] [**write** *writeview*] [**access** {[**ipv6** *ipv6_aclname*] [*aclnum* | *aclname*] }]	Creates a group and associate it with the view.

You can delete a view by using the **no snmp-server view** *view-name* command, or delete a tree from the view by using the **no snmp-server view** *view-name oid-tree* command. You can also delete a group by using the **no snmp-server group** *groupname* {**v1 | v2c | v3**} command.

9.2.3. Configuring SNMP Users

User-based security model can be used for security management. In this mode, you should configure user information first. The NMS can communicate with the SMP Agent by using a valid user account.

For SNMPv3 users, you can specify security level, authentication algorithm (MD5 or SHA), authentication password, encryption algorithm (only DES now) and encryption password.

To configure a SNMP user, run the following commands in global configuration mode:

Command	Function
Qtech(config)# **snmp-server user** *username roupname* {**v1 | v2c | v3 [encrypted] [auth { md5|sha }** *auth-password*] **[priv des56** *priv-password*] } [**access {[ipv6** *ipv6_aclname*] [*aclnum* | *aclname*] }]	Configures the user information.

To remove the specified user, use the **no snmp-server user** *username groupname* {**v1 | v2c | v3**} command.

9.2.4. Configuring Host Address

In special cases, the SNMP Agent may also proactively send messages to the NMS.

To configure the NMS host address that the SNMP Agent proactively sends messages to, run the following commands in global configuration mode:

Command	Function
Qtech(config)# **snmp-server host** { *host-addr* | ipv6 *ipv6-addr* } [**vrf** *vrfname*] [**traps**] [**version { 1 | 2c | 3 { auth | noauth | priv** }] *community-string* [**udp-port** *port-num*] [*notification-type*]	Sets the SNMP host address, host port, vrf options, message type, community string, (user name in SNMPv3) and security level (only supported in SNMPv3).

9.2.5. Configuring SNMP Agent Parameters

You can configure the basic parameters of the SNMP Agent, including contact, device location and sequence number. With these parameters, the NMS knows the contact, location and other information of the device.

To configure the SNMP agent parameters, run the following commands in global configuration mode:

Command	Function
Qtech(config)# **snmp-server contact** *text*	Configures the contact for the system.
Qtech(config)# **snmp-server location** *text*	Configure the location of the system.
Qtech(config)# **snmp-server chassis-id** *number*	Configure the sequence number of the system.

9.2.6. Defining the Maximum Message Size of the SNMP Agent

In order to reduce influence on network bandwith, you can configure the maximum packet size of the SNMP Agent. To configure the maximum packet size of the SNMP Agent, run the following command in global configuration mode:

Command	Function
Qtech(config)# **snmp-server packetsize** *byte-count*	Sets the maximum packet size of the SNMP Agent.

9.2.7. Shielding the SNMP Agent

The SNMP Agent service is a service provided by Qtech product and it is enabled by default. You can shield the

SNMP agent service and related configuration by executing the following command in global configuration mode:

Command	Function
Qtech(config)# **no snmp-server**	Shields the SNMP agent service.

9.2.8. Disabling the SNMP Agent

Qtech products provide a different command from the shield command to disable the SNMP Agent. This command will directly disable all SNMP services (the SNMP agent function is disabled, no message is received and no response or Trap message is sent) instead of shielding the configuration information of the SNMP Agent. To disable the SNMP agent service, run the following command in global configuration mode:

Command	Function
Qtech(config)# **no enable service snmp-agent**	Disables the SNMP agent service.

9.2.9. Configuring the SNMP Agent to Send the Trap Message to the NMS Initiatively

The TRAP message is a message automatically sent by the SNMP Agent to the NMS, and is used to report some critical and important events. By default the SNMP Agent is not allowed to automatically send the TRAP message. To enable it, run the following command in global configuration mode:

Command	Function
Qtech(config)# **snmp-server enable traps** [*type*] [*option*]	Allows the SNMP Agent to send the TRAP message proactively.
Qtech(config)# **no snmp-server enable traps** [*type*] [*option*]	Forbids the SNMP Agent to send the TRAP message proactively.

9.2.10. Configuring LinkTrap Policy

You can configure whether to send the LinkTrap message on an interface. When this function is enabled and the link status of the interface changes, the SNMP will send the LinkTrap message. Otherwise, it will not. By default, this function is enabled.

Command	Function
Qtech(config)# **interface** *interface-id*	Enters interface configuration mode.
Qtech(config-if)# **[no] snmp trap link-status**	Enables or disables sending the LinkTrap message on the interface.

The following example configures not to send LinkTrap message on the intereface:

```
Qtech(config)# interface gigabitEthernet 1/1
Qtech(config-if)# no snmp trap link-status
```

9.2.11. Configuring the Parameters for Sending the Trap Message

To set the parameters for the SNMP Agent to send the Trap message, run the following commands:

Command	Function
Qtech(config)# **snmp-server trap-source** *interface*	Specifies the source port sending the Trap message.
Qtech(config)# **snmp-server queue-length** *length*	Specifies the queue length of each Trap message.
Qtech(config)# **snmp-server trap-timeout** *seconds*	Specifies the interval at which the Trap message is sent.

9.2.12. Configuring SNMP Attack Protection

Enable SNMP attack protection by confining limited times of failed SNMP consecutive authentications and specifying the solution after consecutive authentications fail. After SNMP authentications fail, the system will blacklist the source IP. When the failed times exceed the limit, the system will restrict the source IP address according to the solutions configured by the device:

- The source IP address that is prevented from authentication permanently cannot try access authentication again unless it is relieved by the administrator manually.
- The source IP address that is prevented from authentication for a while can try access authentication again when the **lock-time** times out or it is relieved by the administrator manually.
- When you try access authentication again, the non-restricted source IP address will pass it as long as you use correct community (for SNMPv1and SNMPv2c) or username (for SNMPv3)..

Run this command in global configuration mode to limit the times of failed SNMP consecutive authentications and specify the solution after consecutive authentications fail.

Command	Function
Qtech(config)# **snmp-server authentication attempt** *times* **exceed** { **lock** | **lock-time** *minutes* | **unlock** }	Sets limited times of failed SNMP consecutive authentications and specifies the solution after authentications fail. **attempt** *times*: The limit of failed SNMP authentications. **Lock**: The source IP address is prevented from access authentication permanently. It is blacklisted unless relieved by the administrator manually. **lock-time** *minutes*: The source IP address is prevented from access authentication for a while and then allowed to be authenticated again. *minutes* refers to the period when the source IP address is prevented. **unlock**: The source IP address continues to be allowed after consecutive authentications fail, similar to the case that SNMP attack protection is not enabled.

Run the **no snmp-server authentication attempt** command to restore SNMP attack protection. By default, the solution taken after consecutive authentications fail is **unlock**. Namely, the IP address is allowed to try access authentication.

9.3. SNMP Monitoring and Maintenance

9.3.1. Checking the Current SNMP Status

To monitor the SNMP status and troubleshoot SNMP configurations, Qtech product provides monitoring commands for SNMP, with which it is possible to easily check the SNMP status of the current network device. In privileged user mode, run the **show snmp** command to check the current SNMP status.

```
Qtech# show snmp
Chassis: 1234567890 0987654321
Contact: admin@qtech.ru
Location: Moscow
2381 SNMP packets input
    5 Bad SNMP version errors
    6 Unknown community name
    0 Illegal operation for community name supplied
    0 Encoding errors
    9325 Number of requested variables
    0 Number of altered variables
    31 Get-request PDUs
    2339 Get-next PDUs
    0 Set-request PDUs
2406 SNMP packets output
    0 Too big errors (Maximum packet size 1500)
    4 No such name errors
    0 Bad values errors
    0 General errors
    2370 Get-response PDUs
    36 SNMP trap PDUs
SNMP global trap: disabled
SNMP logging: enabled
SNMP agent: enabled
.
```

The above statistics is explained as follows:

Showing Information	Description
Bad SNMP version errors	SNMP version is incorrect.
Unknown community name	The community name cannot be identified.
Illegal operation for community name supplied	Illegal operation
Encoding errors	Code error
Get-request PDUs	Get-request message
Get-next PDUs	Get-next message
Set-request PDUs	Set-request message
Too big errors (Maximum packet size 1500)	Too large response message
No such name errors	The specified management unit does not exist.
Bad values errors	Specified value type error
General errors	General error
Get-response PDUs	Get-response message
SNMP trap PDUs	SNMP trap message

9.3.2. Checking the MIB Objects Supported by the Current SNMP Agent

To check the MIB objects supported by the current SNMP Agent, run the **show snmp mib** command in privileged user mode:

```
Qtech# show snmp mib
sysDescr
sysObjectID
sysUpTime
sysContact
sysName
sysLocation
sysServices
sysORLastChange
snmpInPkts
snmpOutPkts
…
```

9.3.3. Viewing SNMP Users

To view the SNMP users configured on the current SNMP agent, run the **show snmp user** command in privileged user mode:

```
Qtech# show snmp user
User name: test
Engine ID: 8000131103000000000000
storage-type: permanent     active
Security level: auth priv
Auth protocol: SHA
Priv protocol: DES
Group-name: g1
```

9.3.4. Viewing SNMP Views and Groups

To view the group configured on the current SNMP agent, run the **show snmp group** command in privileged user mode:

```
Qtech# show snmp group
groupname: g1
securityModel: v3
securityLevel:authPriv
readview: default
writeview: default
notifyview:
groupname: public
securityModel: v1
securityLevel:noAuthNoPriv
readview: default
writeview: default
notifyview:
groupname: public
```

```
securityModel: v2c
securityLevel:noAuthNoPriv
readview: default
writeview: default
notifyview:
```

To view the view configured on the current SNMP agent, run the **show snmp view** command in privileged user mode:

```
Qtech# show snmp view
default(include) 1.3.6.1
test-view(include) 1.3.6.1.2.1
```

9.3.5. Viewing Host Information

To view the host information configured on the SNMP agent, run the **show snmp host** command in privileged user mode:

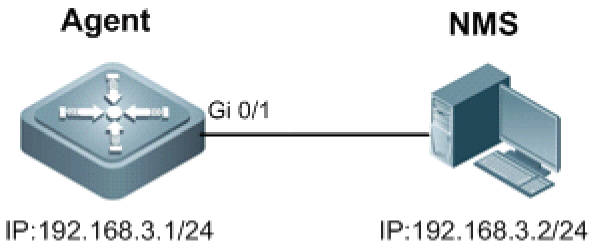
```
Qtech# show snmp host
Notification host: 192.168.64.221
udp-port: 162    type: trap
user: public     security model: v1
Notification host: 2000:1234::64
udp-port: 162    type: trap
user: public     security model: v1
```

9.4. Typical SNMP Configuration Example

9.4.1. SNMP v1/v2 Configuration Example

9.4.1.1. Topological Diagram

Figure 4 Topology for SNMP v1/2 application



9.4.1.2. Application Requirements

The Network Management Station (NMS) manages the network device (Agent) by applying the community-based authentication model, and the network device can control the operation permission (read or write) of the community to access the specified MIB objects. For example, community "user1" can only read and write objects under System (1.3.6.1.2.1.1) node.

The network device can only be managed by NMS with a specific IP (i.e., 192.168.3.2/24).

The network device can actively send messages to NMS.

The NMS can acquire the basic system information of the device, such as contact, location, and ID.

9.4.1.3. Configuration Tips

By creating MIB view and associating authentication name (Community) and access permission (Read or Write), the first application need can be met.

While configuring the community string and access permission, associate ACL or specify the IP of administrator using this community string to meet the second application need (this example associates the ACL).

Configure the address of SNMP host and enable the Agent to actively send Trap messages.

Configure the parameters of SNMP agent.

9.4.1.4. Configuration Steps

Step 1: Configure MIB view and ACL.

! Create an MID view named "v1", which contains the associated MIB object (1.3.6.1.2.1.1).

```
Qtech(config)#snmp-server view v1 1.3.6.1.2.1.1 include
```

! Create an ACL named "a1" to permit the IP address of 192.168.3.2/24.

```
Qtech(config)#ip access-list standard a1
Qtech(config-std-nacl)#permit host 192.168.3.2
Qtech(config-std-nacl)#exit
```

Step 2: Configure community string and access permission.

! Configure Community of "user1", associate write permission for MIB view of "v1", and associate the ACL of "a1".

```
Qtech(config)#snmp-server community user1 view v1 rw a1
```

Step 3: Configure the Agent to actively send messages to NMS.

! Configure the address of SNMP host as 192.168.3.2, message format as Version 2c and community string as "user1".

```
Qtech(config)#snmp-server host 192.168.3.2 traps version 2c user1
```

! Enable the Agent to actively send traps.

```
Qtech(config)#snmp-server enable traps
```

Step 4: Configure parameters of SNMP agent.

! Configure system location.

```
Qtech(config)#snmp-server location moscow
```

! Configure system contact.

```
Qtech(config)#snmp-server contact admin@qtech.ru
```

! Configure system serial number.

```
Qtech(config)#snmp-server chassis-id 1234567890
```

Step 5: Configure the IP address for the Agent.

! Configure the IP address of Gi 0/1 as 192.168.3.1/24.

```
Qtech(config)#interface GigabitEthernet 0/1
Qtech(config-if-GigabitEthernet 0/1)#ip address 192.168.3.1 255.255.255.0
Qtech(config-if-GigabitEthernet 0/1)#exit
```

9.4.1.5. Verification

Step 1: Display configurations of the device.

```
Qtech#show running-config
!
ip access-list standard a1
 10 permit host 192.168.3.2
!
interface GigabitEthernet 0/1
 no ip proxy-arp
 ip address 192.168.3.1 255.255.255.0
!
snmp-server view v1 1.3.6.1.2.1.1 include
snmp-server location moscow
snmp-server host 192.168.3.2 traps version 2c user1
snmp-server enable traps
snmp-server contact admin@qtech.ru
snmp-server community user1 view v1 rw a1
snmp-server chassis-id 1234567890
```

Step 2: Display information about SNMP view and group.

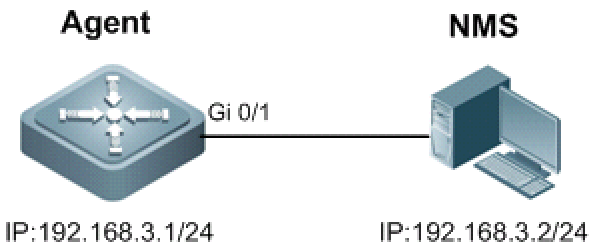
```
Qtech#show snmp view
v1(include) 1.3.6.1.2.1.1                    //define MIB object of “v1”
default(include) 1.3.6.1                     //default MIB object
Qtech#show snmp group
groupname: user1                      //Configure Community as SNMP group
securityModel: v1
securityLevel:noAuthNoPriv
readview: v1
writeview: v1
notifyview:
groupname: user1
securityModel: v2c
securityLevel:noAuthNoPriv
readview: v1
writeview: v1
notifyview:
```

Step 3: Install MIB-Browser. Type in device IP of "192.168.3.1" in the field of IP Address; type in "user1" in the field of Community Name; click **Add Item** button and select the specific management unit for MIB query, such as the System shown below. Click **Start** to implement MIB query of network device.

9.4.2. SNMP v3 Configuration Example

9.4.2.1. Topological Diagram

Figure 6 SNMPv3 application topology



9.4.2.2. Application Requirements

Network Management Station manages the network device (Agent) by applying user-based security model. For example: the user name is "user1", authentication mode is MD5, authentication key is "123", encryption algorithm is DES56, and the encryption key is "321".

The network device can control user's permission to access MIB objects. For example: "User1" can read the MIB objects under System (1.3.6.1.2.1.1) node, and can only write MIB objects under SysContact (1.3.6.1.2.1.1.4.0) node.

The network device can actively send authentication and encryption messages to the network management station.

9.4.2.3. Configuration Tips

Create MIB view and specify the included or excluded MIB objects.

Create SNMP group and configure the version as "v3"; specify the security level of this group, and configure the read/write permission of the view corresponding to this group.

Create user name and associate the corresponding SNMP group name to further configure the user's permission to access MIB objects; meanwhile, configure the version number as "v3" and the corresponding authentication mode, authentication key, encryption algorithm and encryption key.

Configure the address of SNMP host, configure the version number as "3" and configure the security level to be adopted.

9.4.2.4. Configuration Steps

Step 1: Configure MIB view and group.

! Create an MIB view of "view1" and include the MIB object of 1.3.6.1.2.1.1; further create an MIB view of "view2" and include the MIB object of 1.3.6.1.2.1.1.4.0.

```
Qtech(config)#snmp-server view view1 1.3.6.1.2.1.1 include
Qtech(config)#snmp-server view view2 1.3.6.1.2.1.1.4.0 include
```

! Create a group named "g1" and select the version number of "v3"; configure security level to "priv" to read "view1" and write "view2".

```
Qtech(config)#snmp-server group g1 v3 priv read view1 write view2
```

Step 2: Configure SNMP user.

! Create a user named "user1", which belongs to group "g1"; select version number of "v3" and configure authentication mode as "md5", authentication key as "123", encryption mode as "DES56" and encryption key as "321".

```
Qtech(config)#snmp-server user user1 g1 v3 auth md5 123 priv des56 321
```

Step 3: Configure the address of SNMP host.

! Configure the host address as 192.168.3.2 and select version number of "3"; configure security level to "priv" and associate the corresponding user name of "user1".

```
Qtech(config)#snmp-server host 192.168.3.2 traps version 3 priv user1
```

! Enable the Agent to actively send traps to NMS.

```
Qtech(config)#snmp-server enable traps
```

Step 4: Configure the IP address of Agent.

! Configure the IP address of Gi 0/1 as 192.168.3.1/24.

```
Qtech(config)#interface gigabitEthernet 0/1
Qtech(config-if-GigabitEthernet 0/1)#ip address 192.168.3.1 255.255.255.0
Qtech(config-if-GigabitEthernet 0/1)#exit
```

9.4.2.5. Verification

Step 1: Display configurations of device.

```
Qtech#show running-config
!
interface GigabitEthernet 0/1
 no ip proxy-arp
 ip address 192.168.3.1 255.255.255.0
!
snmp-server view view1 1.3.6.1.2.1.1 include
snmp-server view view2 1.3.6.1.2.1.1.4.0 include
snmp-server user user1 g1 v3 encrypted auth md5 7EBD6A1287D3548E4E52CF8349CBC93D priv
des56 D5CEC4884360373ABBF30AB170E42D03
snmp-server group g1 v3 priv read view1 write view2
snmp-server host 192.168.3.2 traps version 3 priv user1
snmp-server enable traps
```

Step 2: Display SNMP user.

```
Qtech# show snmp user
User name: user1
Engine ID: 800013110300d0f8221120
storage-type: permanent     active
Security level: auth priv
Auth protocol: MD5
Priv protocol: DES
Group-name: g1
```

Step 3: Display SNMP view.

```
Qtech#show snmp view
view1(include) 1.3.6.1.2.1.1
view2(include) 1.3.6.1.2.1.1.4.0
default(include) 1.3.6.1
```

Step 4: Display SNMP group.

```
Qtech# show snmp group
groupname: g1
securityModel: v3
securityLevel:authPriv
readview: view1
writeview: view2
notifyview:
```

Step 5: Display host information configured by the user.

```
Qtech#show snmp host
Notification host: 192.168.3.2
udp-port: 162
type: trap
user: user1
security model: v3 authPriv
```

Step 6: Install MIB-Browser. Type in device IP of "192.168.3.1" in the field of IP Address; type in "user1" in the field of UserName; select "AuthPriv" from Security Level; type in "123" in the field of AuthPassWord; select "MD5" from AuthProtocol; type in "321" in the field of PrivPassWord. Click **Add Item** button and select the specific management unit for querying MIB, such as the System shown below. Click **Start** to implement MIB query of network device

10. CONFIGURING CWMP

10.1. Overview

CPE WAN Management Protocol (CWMP) provides a general framework of unified device management, related message specifications, management methods, and data models, so as to solve difficulties in unified management and maintenance of dispersed customer-premises equipment (CPEs), improve troubleshooting efficiency, and save O&M costs.

CWMP provides the following functions:

- **Auto configuration and dynamic service provisioning.** CWMP allows an Auto-Configuration Server (ACS) to automatically provision CPEs who initially access the network after start. The ACS can also dynamically re-configure running CPEs.
- **Firmware management.** CWMP manages and upgrades the firmware and its files of CPEs.
- **Software module management.** CWMP manages modular software according to data models implemented.
- **Status and performance monitoring.** CWMP enables CPEs to notify the ACE of its status and changes, achieving real-time status and performance monitoring.
- **Diagnostics.** The ACE diagnoses or resolves connectivity or service problems based on information from CPEs, and can also perform defined diagnosis tests.

Protocols and Standards

For details about TR069 protocol specifications, visit http://www.broadband-forum.org/technical/trlist.php.

Listed below are some major CWMP protocol specifications:

- TR-069_Amendment-4.pdf: CWMP standard
- TR-098_Amendment-2.pdf: Standard for Internet gateway device data model
- TR-106_Amendment-6.pdf: Standard for CPE data model
- TR-181_Issue-2_Amendment-5.pdf: Standard for CPE data model 2
- tr-098-1-4-full.xml: Definition of Internet gateway device data model
- tr-181-2-4-full.xml: Definition 2 of CPE data model 2

10.2. Applications

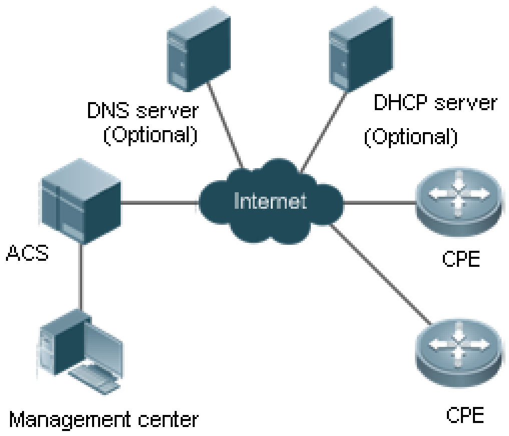
Typical Application	Scenario
CWMP Network Application Scenario	Initiate CPE-ACS connection, so as to upgrade the CPE firmware, upload the configuration files, restore the configuration, and realize other features.

10.2.1. CWMP Network Application Scenario

Application Scenario

The major components of a CWMP network architecture are CPEs, an ACS, a management center, a DHCP server, and a Domain Name System (DNS) server. The management center manages a population of CPEs by controlling the ACS on a Web browser.

Figure 10-1



Note

- If the Uniform Resource Locator (URL) of the ACS is configured on CPEs, the DHCP server is optional. If not, the DHCP is required to dynamically discover the ACS URL.
- If the URLs of the ACS and CPEs contain IP addresses only, the DNS server is optional. If their URLs contain domain names, the DNS server is required to resolves the names.

Functional Deployment

HTTP runs on both CPEs and the ACS.

10.3. Features

Basic Concept

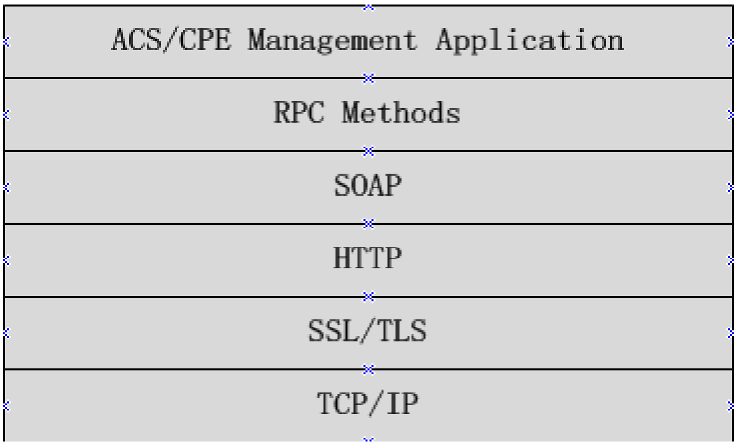
Major Terminologies

- **CPE**: Customer Premises Equipment
- **ACS**: Auto-Configuration Server
- **RPC**: Remote Procedure Call
- **DM**: Data Model

Protocol Stack

Figure 10-2 shows the protocol stack of CWMP.

Figure 10-2 CWMP Protocol Stack



As shown in **Ошибка! Источник ссылки не найден.**, CWMP defines six layers with respective functions as follows:

- **ACS/CPE Application**

The application layer is not a part of CWMP. It is the development performed by various modules of the CPEs/ACS to support CWMP, just like the Simple Network Management Protocol (SNMP), which does not cover the MIB management of functional modules.

- **RPC Methods**

This layer provides various RPC methods for interactions between the ACS and the CPEs.

- **SOAP**

The Simple Object Access Protocol (SOAP) layer uses a XML-based syntax to encode and decode CWMP messages.. Thus, CWMP messages must comply with the XML-based syntax.

- **HTTP**

All CWMP messages are transmitted over Hypertext Transfer Protocol (HTTP). Both the ACS and the CPEs can behave in the role of HTTP clients and servers. The server function is used to monitor reverse connections from the peer.

- **SSL/TLS**

The Secure Sockets Layer (SSL) or Transport Layer Security (TLS) layer guarantees CWMP security, including data integrity, confidentiality, and authentication.

- **TCP/IP**

This layer is the (Transmission Control Protocol/Internet Protocol (TCP/IP) protocol stack.

↘ RPC Methods

The ACS manages and monitors CPEs by calling mostly the following RPC methods:

- **Get RPC Methods**

The Get methods enable the ACS to remotely obtain the set of RPC methods, as well as names, values and attributes of the DM parameters supported on CPEs.

- **Set RPC Methods**

The Set methods enable the ACS to remotely set the values and attributes of the DM parameters supported on CPEs.

- **Inform RPC Methods**

The Inform methods enable CPEs to inform the ACS of their device identifiers, parameter information, and events whenever sessions are established between them.

- **Download RPC Methods**

The Download method enables the ACS to remotely control the file download of CPEs, including firmware management, upgrade, and Web package upgrade.

- **Upload RPC Methods**

The Upload method enables the ACS to remotely control the file upload of CPEs, including upload of firmware and logs.

- **Reboot RPC Methods**

The Reboot method enables the ACS to remotely reboot the CPEs.

↘ Session Management

CWMP sessions or interactions are the basis for CWMP. All CWMP interactions between the ACS and CPEs rely on their sessions. CWMP helps initiate and maintain ACS-CPE sessions to link them up for effective management and monitoring. An ACS-CPE session is a TCP connection, which starts from the Inform negotiation to TCP disconnection. The session is classified into CPE Initiated Session and ACS Initiated Session according to the session poster.

↘ DM Management

CWMP operates based on CWMP Data Model (DM). CWMP manages all functional modules by a set of operations performed on DM. Each functional module registers and implements a respective data model, just like the MIBs implemented by various functional modules of SNMP.

A CWMP data model is represented in the form of a character string. For a clear hierarchy of the data model, a dot (.) is used as a delimiter to distinguish an upper-level data model node from a lower-level data model node. For instance, in the data model **InternetGatewayDevice.LANDevice**, **InternetGatewayDevice** is the parent data model node of **LANDevice**, and **LANDevice** is the child data model node of **InternetGatewayDevice**.

DM nodes are classified into two types: object nodes and parameter nodes. The parameter nodes are also known as leaf nodes. An object node is a node under which there are child nodes, and a parameter node is a leaf node under which there is no any child node. Object nodes are further classified into single-instance object nodes and multi-instance object nodes. A single-instance object node is an object node for which there is only one instance, whereas a multi-instance object node is an object node for which there are multiple instances.

DM nodes can also be classified into readable nodes and readable-and-writable nodes. A readable node is a node whose parameter values can be read but cannot be modified, and a readable-and-writable node is a node whose parameter values can be both read and modified.

A data model node has two attributes. One attribute relates to a notification function; that is, whether to inform the ACS of changes (other than changes caused by CWMP) to parameter values of the data model. The other attribute is an identifier indicating that the parameters of the data model node can be written using other management modes (than the ACS); that is, whether the values of the parameters can be modified using other management modes such as Telnet. The ACS can modify the attributes of the data models using RPC methods.

CWMP manages the data models using corresponding RPC methods.

↘ Event Management

When some events concerned by the ACS occur on the CPE, the CPE will inform the ACS of these events. The ACS monitors these events to monitor the working status of the CPE. The CWMP events are just like Trap messages of SNMP or product logs. Using RPC methods, to the ACS filters out the unconcerned types of events. CWMP events are classified into two types: single or (not cumulative) events and multiple (cumulative) events. A single event means that there is no quantitative change to the same event upon re-occurrence of the event, with the old discarded and the newest kept. A multiple event means that the old are not discarded and the newest event is kept as a complete event when an event re-occurs for multiple times later; that is, the number of this event is incremented by 1.

All events that occur on the CPE are notified to the ACS using the INFORM method.

Features

Feature	Description
Upgrading the Firmware	The ACS controls the upgrade of the firmware of a CPE using the Download method.

Upgrading the Configuration Files	The ACS controls the upgrade of the configuration files of a CPE using the Download method.
Uploading the Configuration Files	The ACS controls the upload of the configuration files of a CPE using the Upload method.
Backing up and Restoring a CPE	When a CPE breaks away from the management center, this feature can remotely restore the CPE to the previous status.

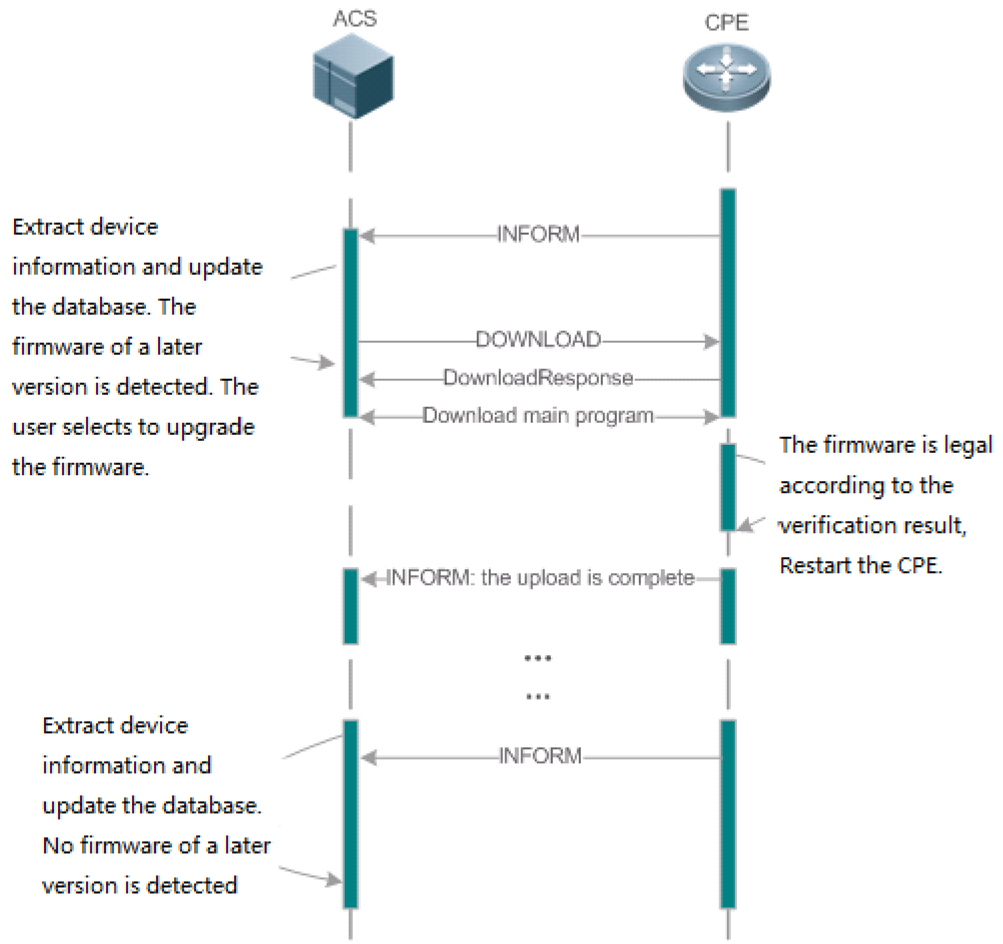
10.3.1. Upgrading the Firmware

Upgrading the Firmware means the firmware of a network element (NE) can be upgraded, so as to implement device version upgrade or replacement.

Working Principle

↘ **Sequence Diagram of Upgrading the Firmware**

Figure 10-3


ACS
CPE
Extract device information and update the database. The firmware of a later version is detected. The user selects to upgrade the firmware.
INFORM
DOWNLOAD
DownloadResponse
Download main program
The firmware is legal according to the verification result, Restart the CPE.
INFORM: the upload is complete
...
...
Extract device information and update the database. No firmware of a later version is detected
INFORM

Users specify a CPE for the ACS to deliver the Download method for upgrading the firmware. The CPE receives the request and starts to download the latest firmware from the destination file server, upgrade the firmware, and then reboot. After restart, the CPE will indicate the successful or unsuccessful completion of the method application.

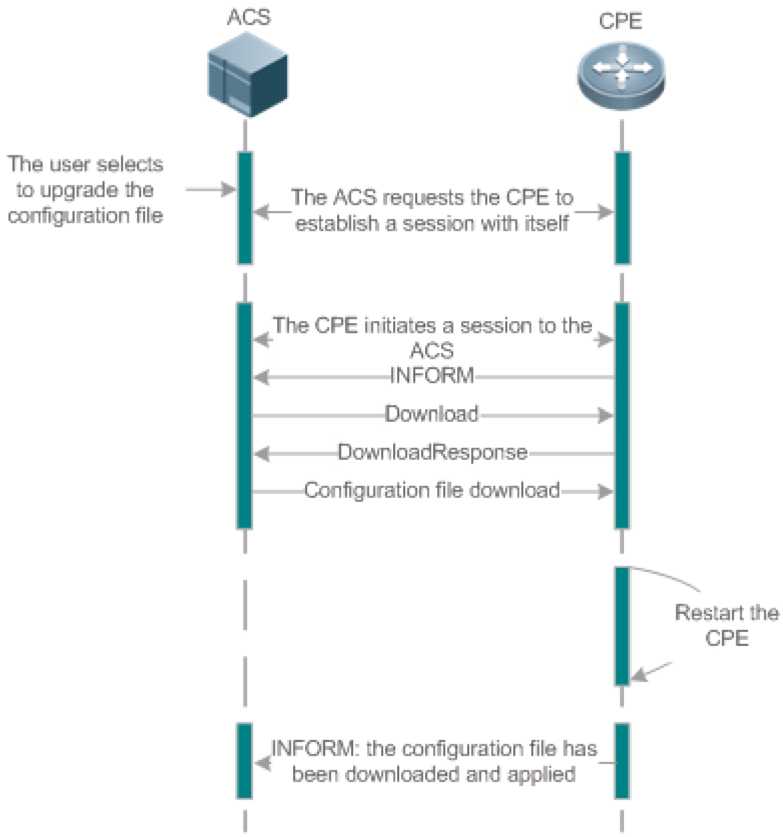
Note The file server can be ACS or separately deployed.

10.3.2. Upgrading the Configuration Files

Upgrading the Configuration Files means the current configuration files of a CPE can be replaced with specified configuration files, so that the new configuration files act on the CPE after reset.

Working Principle

Figure 10-4


ACS
CPE
The user selects to upgrade the configuration file
The ACS requests the CPE to establish a session with itself
The CPE initiates a session to the ACS
INFORM
Download
DownloadResponse
Configuration file download
Restart the CPE
INFORM: the configuration file has been downloaded and applied

Users specify a CPE for the ACS to deliver the Download methods for upgrading its configuration files. The CPE downloads the configuration files from the specified file server, upgrade configuration files, and then reboot. After that, the CPE will indicate successful or unsuccessful completion of the method application.

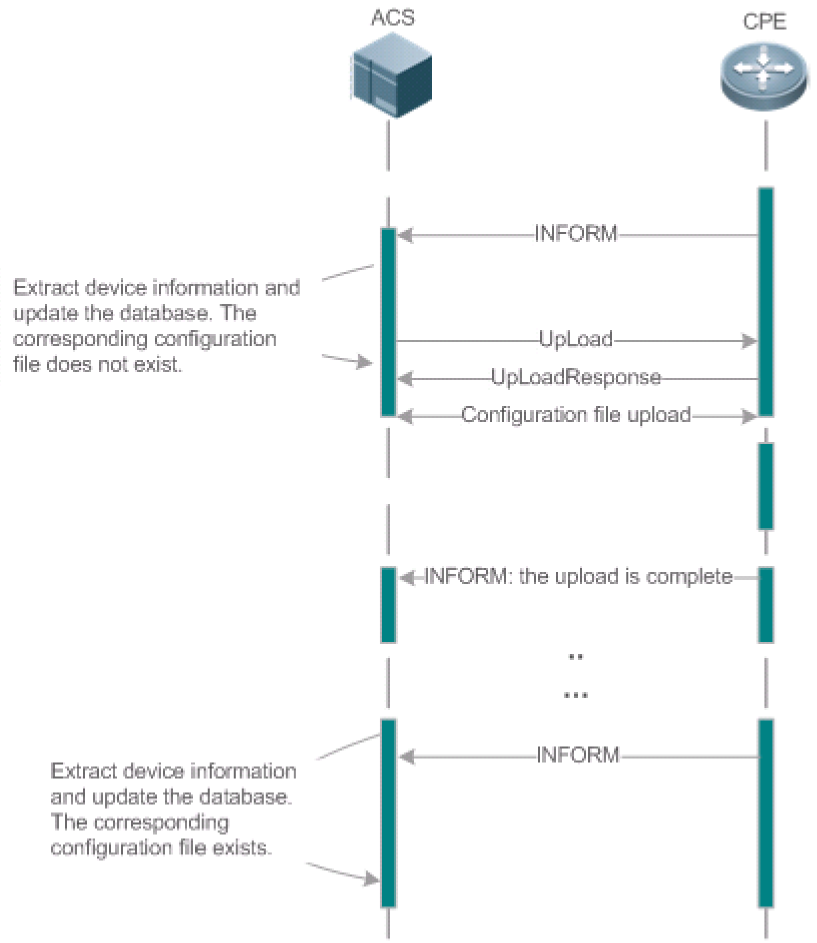
Note The file server can be ACS or separately deployed.

10.3.3. Uploading the Configuration Files

Uploading the Configuration Files means the ACS controls the configuration files of CPEs by using the Upload method.

Working Principle

Figure 10-5


ACS
CPE
INFORM
Extract device information and update the database. The corresponding configuration file does not exist.
UpLoad
UpLoadResponse
Configuration file upload
INFORM: the upload is complete
...
INFORM
Extract device information and update the database. The corresponding configuration file exists.

When a CPE initially accesses the ACS, the ACS attempts to learn the configuration files of the CPE in the following sequence:

- When the ACS initially receives an Inform message from the CPE, it locates the corresponding database information according to device information carried in the message.
- If the database does not contain the configuration files of the CPE, the ACS delivers the Upload method to the CPE for uploading the configuration files.
- The CPE uploads its current configuration files to the ACS.
- The CPE returns a successful or unsuccessful response to the Upload request.

10.3.4. Backing Up and Restoring a CPE

When a remote CPE breaks away from the management center due to abnormal operations, the CPE backup and restoration feature helps restore the CPE to the previous status, so that the management center can resume the supervision of the CPE as necessary.

Working Principle

You can configure the restoration function on a CPE, so that the CPE can restore itself from exceptions of its firmware or configuration files. Then when the CPE fails to connect to the ACS and breaks away from the management center after its firmware or configuration files are upgraded, the previous firmware or configuration files of the CPE can be restored in time for the ACS to manage the CPE. This kind of exception is generally caused by delivery of a wrong version or configuration file.

Before the CPE receives a new firmware or configuration files to upgrade, the CPE will back up its current version and configuration files. In addition, there is a mechanism for determining whether the problem described in the preceding scenario has occurred. If the problem has occurred, the CPE is restored to the previous manageable status.

10.4. Configuration

Action	Suggestions and Related Commands	
Establishing a Basic CWMP Connection	⚠ (Mandatory) You can configure the ACS or CPE usernames and passwords to be authenticated for CWMP connection.	
	cwmp	Enables CWMP and enters CWMP configuration mode.
	acs username	Configures the ACS username for CWMP connection.
	acs password	Configures the ACS password for CWMP connection.
	cpe username	Configures the CPE username for CWMP connection.
	cpe password	Configures the CPE password for CWMP connection.
	⚠ (Optional) You can configure the URLs of the CPE and the ACS.	
	acs url	Configures the ACS URL.
	cpe url	Configures the CPE URL.
Configuring CWMP-Related Attributes	⚠ (Optional) You can configure the basic functions of the CPE, such as upload, backup and restoration of firmware, configuration files or logs.	
	cpe inform	Configures the periodic notification function of the CPE.
	cpe back-up	Configures the backup and restoration of the firmware and configuration file of the CPE.
	disable download	Disables the function of downloading firmware and configuration files from the ACS.
	disable upload	Disables the function of uploading configuration and log files to the ACS.
	timer cpe- timeout	Configures the ACS response timeout on CPEs.

10.4.1. Establishing a Basic CWMP Connection

Configuration Effect

- A session connection is established between the ACS and the CPE.

Precautions

- N/A

Configuration Method

↘ Enabling CWMP and Entering CWMP Configuration Mode

- (Mandatory) The CWMP function is enabled by default.

Command	**cwmp**
Parameter Description	N/A
Defaults	CWMP is enabled by default.
Command Mode	Global configuration guide
Usage Guide	N/A

↘ Configuring the ACS Username for CWMP Connection

- This configuration is mandatory on the ACS.
- Only one username can be configured for the ACS. If multiple are configured, the latest configuration is applied.

Command	**acs username** *username*
Parameter Description	**username** *username*: The ACS username for CWMP connection
Defaults	The ACS username is not configured by default.
Command Mode	CWMP configuration mode
Usage Guide	N/A

↘ Configuring the ACS Password for CWMP Connection

- This configuration is mandatory on the ACS.
- The password of the ACS can be in plaintext or encrypted form. Only one password can be configured for the ACS. If multiple are configured, the latest configuration is applied.

Command	**acs password** {*password* | *encryption-type encrypted-password*}
Parameter Description	*password:* ACS password *encryption-type:* 0 (no encryption) or 7 (simple encryption) *encrypted-password:* Password text
Defaults	*encryption-type*: 0 *encrypted-password*: N/A
Command Mode	CWMP configuration mode
Usage Guide	N/A

↘ Configuring the CPE Username for CWMP Connection

- This configuration is mandatory on the CPE.
- Only one username can be configured for the CPE. If multiple are configured, the latest configuration is applied.

Command	**cpe username** *username*
Parameter Description	*username*: CPE username
Defaults	No CPE username is configured by default.
Command Mode	CWMP configuration mode
Usage Guide	N/A

↘ Configuring the CPE Password for CWMP Connection

- This configuration is mandatory on the CPE.
- The password of the CPE can be in plaintext or encrypted form. Only one password can be configured for the CPE. If multiple are configured, the latest configuration is applied.

Command	**cpe password** {*password* | *encryption-type encrypted-password*}
Parameter Description	*password:* CPE password *encryption-type:* 0 (no encryption) or 7 (simple encryption)

	encrypted-password: Password text
Defaults	*encryption-type*: 0 *encrypted-password*: N/A
Command Mode	CWMP configuration mode
Usage Guide	Use this command to configure the CPE user password to be authenticated for the ACS to connect to the CPE. In general, the encryption type does not need to be specified. The encryption type needs to be specified only when copying and pasting the encrypted password of this command. A valid password should meet the following format requirements: ● Contain 1 to 26 characters including letters and figures. ● The leading spaces will be ignored, while the trailing and middle are valid. ● If 7 (simple encryption) is specified, the valid characters only include 0 to 9 and a (A) to f (F).

Configuring the ACS URL for CMWP Connection

- This configuration is optional on the CPE.
- Only one ACS URL can be configured. If multiple are configured, the latest configuration is applied. The ACS URL must be in HTTP format.

Command	**acs url** *url*
Parameter Description	*url*: ACS URL
Defaults	No ACS URL is configured by default.
Command Mode	CWMP configuration mode
Usage Guide	If the ACS URL is not configured but obtained through DHCP, CPEs will use this dynamic URL to initiate connection to the ACS. The ACS URL must: ● Be in format of **http://host[:port]/path**. ● Contain 256 characters at most.

Configuring the CPE URL for CWMP Connection

- This configuration is optional on the CPE.
- Only one CPE URL can be configured. If multiple are configured, the latest configuration is applied. The CPE URL must be in HTTP format instead of domain name format.

Command	**cpe url** *url*
Parameter Description	*url:* CPE URL
Defaults	No CPE URL is configured by default.
Command Mode	CWMP configuration mode
Usage Guide	If CPE URL is not configured, it is obtained through DHCP. The CPE URL must: ● Be in format of **http://ip [: port]/**. ● Contain 256 characters at most.

Verification

- Run the **show cwmp configuration** command.

Command	**show cwmp configuration**
Parameter Description	N/A
Command Mode	Privileged EXEC mode
Usage Guide	N/A
Configuration Examples	The following example displays the CWMP configuration.

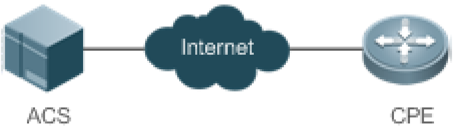
```
Qtech(config-cwmp)#show cwmp configuration
CWMP Status                     : enable
ACS URL                         : http://www.Qtech.ru/acs
ACS username                    : admin
ACS password                    : ******
CPE URL                         : http://10.10.10.2:7547/
CPE username                    : Qtech
CPE password                    : ******
CPE inform status                : disable
CPE inform interval              : 60s
CPE inform start time            : 0:0:0 0 0 0
CPE wait timeout                 : 50s
CPE download status              : enable
CPE upload status                : enable
CPE back up status               : enable
CPE back up delay time           : 60s
```

Configuration Examples

The following configuration examples describe CWMP-related configuration only.

↘ Configuring Usernames and Passwords on the CPE

Network Environment
Figure 10-6



Configuration Method

- Enable CWMP.
- On the CPE, configure the ACS username and password to be authenticated for the CPE to connect to the ACS.
- On the CPE, configure the CPE username and password to be authenticated for the ACS to connect to the CPE.

CPE

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# cwmp
Qtech(config-cwmp)# acs username USERB
Qtech(config-cwmp)# acs password PASSWORDB
Qtech(config-cwmp)# cpe username USERB
Qtech(config-cwmp)# cpe password PASSWORDB
```

Verification

- Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech # show cwmp configuration
CWMP Status                         : enable
ACS URL                              : http://10.10.10.1:7547/acs
ACS username                        : USERA
ACS password                        : ******
CPE URL                              : http://10.10.10.2:7547/
CPE username                        : USERB
CPE password                        : ******
```

↘ Configuring the URLs of the ACS and the CPE

Network Environment See **Ошибка! Источник ссылки не найден.**.

Configuration Method

- Configure the ACS URL.
- Configure the CPE URL.

CPE

```
Qtech# configure terminal
Qtech(config)# cwmp
Qtech(config-cwmp)# acs url http://10.10.10.1:7547/acs
Qtech(config-cwmp)# cpe url http://10.10.10.1:7547/
```

Verification Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech #show cwmp configuration
CWMP Status                         : enable
ACS URL                              : http://10.10.10.1:7547/acs
ACS username                        : USERA
ACS password                        : ******
CPE URL                              : http://10.10.10.2:7547/
```

Common Errors

- The user-input encrypted password is longer than 254 characters, or the length of the password is not an even number.
- The user-input plaintext password is longer than 100 characters.
- The user-input plaintext password contains illegal characters.
- The user-input encrypted password contains illegal characters (the legitimate characters includes only 0~9, a~f and A~F)
- The URL of the ACS is set to **NULL**.
- The URL of the CPE is set to **NULL**.

10.4.2. Configuring CWMP-Related Attributes

Configuration Effect

- You can configure common functions of the CPE, such as the backup and restoration of its firmware or configuration file, whether to enable the CPE to download firmware and configuration files from the ACS, and whether to enable the CPE to upload its configuration and log files to the ACS.

Configuration Method

↘ Configuring the Periodic Notification Function of the CPE

- (Optional) The value range is from 30 to 3,600 in seconds. The default value is 600 seconds.
- Perform this configuration to reset the periodical notification interval of the CPE.

Command	**cpe inform [interval** *seconds*] **[starttime** *time*]
Parameter Description	*seconds*: Specifies the periodical notification interval of the CPE. The value range is from 30 to 3,600 in seconds. *time*: Specifies the date and time for starting periodical notification in *yyyy-mm-ddThh:mm:ss* format.
Command Mode	CWMP configuration mode
Defaults	The default value is 600 seconds.
Usage Guide	Use this command to configure the periodic notification function of the CPE. ● If the time for starting periodical notification is not specified, periodical notification starts after the periodical notification function is enabled. The notification is performed once within every notification interval. ● If the time for starting periodical notification is specified, periodical notification starts at the specified start time. For instance, if the periodical notification interval is set to 60 seconds and the start time is 12:00 am next day, periodical notification will start at 12:00 am next day and once every 60 seconds.

↘ Disabling the Function of Downloading Firmware and Configuration Files from the ACS

- (Optional) The CPE can download firmware and configuration files from the ACS by default.
- Perform this configuration if the CPE does not need to download firmware and configuration files from the ACS.

Command	**disable download**
Parameter Description	N/A
Defaults	The CPE can download firmware and configuration files from the ACS by default.
Command Mode	CWMP configuration mode
Usage Guide	Use this command to disable the function of downloading main program and configuration files from the ACS. ● This command does not act on configuration script files. The configuration scripts can still be executed even if this function is disabled.

↘ Disabling the Function of Uploading Configuration and Log Files to the ACS

- (Optional.) The CPE can upload configuration and log files to the ACS by default.
- Perform this configuration if the CPE does not need to upload configuration and log files to the ACS.

Command	**disable upload**
Parameter Description	N/A
Defaults	The CPE can upload configuration and log files to the ACS by default.

Command Mode	CWMP configuration mode
Usage Guide	Use this command to disable the function of uploading configuration and log files to the ACS.

Configuring the Backup and Restoration of the Firmware and Configuration Files of the CPE

- (Optional) The backup and restoration of the firmware and configuration files of the CPE is enabled by default. The value range is from 10 to 600 in seconds. The default value is 30 seconds.
- The longer the delay-time is, the longer the reboot will be complete.
- Perform this configuration to modify the function of backing up and restoring the firmware and configuration files of the CPE.

Command	**cpe back-up [delay-time** *seconds*]
Parameter Description	*seconds*: Specifies the delay for backup and restoration of the firmware and configuration file of the CPE.
Defaults	The default value is 30 seconds.
Command Mode	CWMP configuration mode
Usage Guide	N/A

Configuring the ACS Response Timeout

- (Optional) The value range is from 10 to 600 in seconds. The default value is 30 seconds.
- Perform this configuration to modify the ACS response timeout period on the CPE.

Command	**timer cpe- timeout** *seconds*
Parameter Description	*seconds*: Specifies the timeout period in seconds. The value range is from 10 to 600.
Defaults	The default value is 30 seconds.
Command Mode	CWMP configuration mode
Usage Guide	N/A

Verification

- Run the **show cwmp configuration** command.

Command	**show cwmp configuration**
Parameter Description	N/A
Command Mode	Privileged EXEC mode
Usage Guide	N/A
Configuration Examples	The following example displays the CWMP configuration.

```
Qtech(config-cwmp)#show cwmp configuration
CWMP Status                     : enable
ACS URL                         : http://www.Qtech.ru/acs
ACS username                    : admin
ACS password                    : ******
CPE URL                         : http://10.10.10.2:7547/
CPE username                    : Qtech
CPE password                    : ******
CPE inform status               : disable
CPE inform interval             : 60s
CPE inform start time           : 0:0:0 0 0 0
CPE wait timeout                : 50s
CPE download status             : enable
CPE upload status               : enable
CPE back up status              : enable
CPE back up delay time          : 60s
```

Configuration Examples

Configuring the Periodical Notification Interval of the CPE

Network Environment

See **Ошибка! Источник ссылки не найден.**.

Configuration Steps

- Enable the CWMP function and enter CWMP configuration mode.
- Set the periodical notification interval of the CPE to 60 seconds.

CPE

```
Qtech#config
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)#cwmp
Qtech(config-cwmp)#cpe inform interval 60
```

Verification

Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech #show cwmp configuration
CWMP Status                                 : enable
……
CPE inform interval                     : 60s
```

Disabling the Function of Downloading Firmware and Configuration Files from the ACS

Network Environment

See **Ошибка! Источник ссылки не найден.**.

Steps

- Enable the CWMP function and enter CWMP configuration mode.
- Disable the function of downloading firmware and configuration files from the ACS.

CPE

```
Qtech#config
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)#cwmp
Qtech(config-cwmp)#disable download
```

Verification

Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech #show cwmp configuration
CWMP Status                                 : enable
……
CPE download status                     : disable
```

Disabling the Function of Uploading Configuration and Log Files to the ACS

Network Environment

See **Ошибка! Источник ссылки не найден.**.

Configuration Steps

- Enable the CWMP function and enter CWMP configuration mode.
- Disable the CPE's function of uploading configuration and log files to the ACS.

CPE

```
Qtech#config
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)#cwmp
Qtech(config-cwmp)# disable upload
```

Verification

Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech #show cwmp configuration
CWMP Status                                 : enable
……
CPE upload status                       : disable
```

Configuring the Backup and Restoration Delay

Network Environment See **Ошибка! Источник ссылки не найден.**.

Configuration Steps

- Enable the CWMP function and enter CWMP configuration mode.
- Set the backup and restoration delay to 100 seconds.

CPE

```
Qtech#config
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)#cwmp
Qtech(config-cwmp)# cpe back-up Seconds 30
```

Verification

- Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech #show cwmp configuration
CWMP Status                       : enable
……
CPE back up delay time          : 30s
```

Configuring the ACS Response Timeout of the CPE

Network Environment See **Ошибка! Источник ссылки не найден.**.

Configuration Steps

- Enable the CWMP function and enter CWMP configuration mode.
- Set the response timeout of the CPE to 100 seconds.

CPE

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# cwmp
Qtech(config-cwmp)# timer cpe-timeout 100
```

Verification

- Run the **show** command on the CPE to check whether the configuration commands have been successfully applied.

CPE

```
Qtech#show cwmp configuration
CWMP Status                      : enable
……
CPE wait timeout                : 100s
```

Common Errors

N/A

10.5. Monitoring

Displaying

Command	Function
show cwmp configuration	Displays the CWMP configuration.
show cwmp status	Displays the CWMP running status.

11. CONFIGURING USB/SD

11.1. Overview

This document describes usage of USB/SD storage devices (mainly U disk/SD). The system only recognizes the U-disk/SD card partitioned by FAT. Other file systems cannot be identified.

After inserting a U disk/SD, the system prompts that U disk/SD is found. The files in this U disk/SD card can be positioned and accessed through URL, such as **usb0:/abc/1.txt** or **sd0:/abc/1.txt**.

Caution

Version 10.4 (2) and the later versions support the access to U disk/SD by URL. For earlier software, use the mount path of the file system to position and access U disk/SD, such as using /mnt/usb0 to access the USB device on port 0, and using /mnt/sd to access the SD card. The mount path is prompted when the device is inserted, or is displayed when users run the **show usb** command. The USB mobile disk (USB-HDD) is not supported.

11.2. Inserting the Device

Just insert a USB device into the USB slot. Messages as below are displayed if the system finds the device and loads the driver.

```
      *Jan  1 00:03:21: %USB-5-USB_DISK_FOUND: USB Disk<USB DISK Pro>Found!

*Jan  1 00:03:21: %USB-5-USB_DISK_PARTITION: USB_DISK_PARTITION found,
/dev/uba/disc0/part4 to /mnt/usb0, size 4007657472B(3822MB)
```

Just insert an SD card into an SD slot. Messages as below are displayed if the system finds the device and loads driver.

```
*Jan  1 00:09:42: %USB-5-USB_DISK_PARTITION_MOUNT: Mount sd (type:FAT32),size :
1050673152B(1002MB)
```

SD: is the first SD partition and **size** is the partition size. This SD card has 1002 MB space.

11.3. Using the Device

After loading U disk/SD card to the system, directly run file system commands (dir, copy, del, and others) to operate U disk/SD card. Operations below show how to copy the file of U disk/SD card to flash.

Access the U disk partition.

```
Qtech# cd usb0:/
Access the SD card partition.
Qtech# cd sd0:/
```

Copy the **a.txt** file in the U disk to root directory of the device.

```
Qtech# copy usb0:/a.txt  flash:/b.txt
```

Copy the a.txt file in the SD card to device’s the root directory of the device.

```
Qtech# copy sd0:/a.txt  flash:/b.txt
```

Run the **dir** command. The result shows that the b.txt file has been added to the USB/SD card.

For other operation commands, see the “File System Management” section.

Caution

If there are multiple partitions in U disk/SD card, only the first FAT partition can be accessed through the device.

Note

Only the version 10.4(2) and the later versions allow users to access U disc/SD card by URL. For the earlier versions, use path to position and access the device. For example:
Access the U disk partition:
Qtech# cd /mnt/usb0
Access the SD card partition:
Qtech# cd /mnt/sd0

Copy the a.txt file under root directory to U disk.
Qtech# **copy** flash:/a.txt usb0:/a.txt
Copy the a.txt file under root directory to SD card.
Qtech# **copy** flash:/a.txt sd0:/a.txt

11.3.1. Showing USB Device/SD Card Information

Command	Function
Qtech# **show usb**	Shows the USB device information of the system
Qtech# **show sd**	Shows the SD device information of the system

In CLI command mode, use the **show usb** or the **show sd** command to view the USB / SD device information of the system. The displayed information is as follows:

```
Qtech# show usb
Device: Mass Storage:
ID: 0
URL prefix: usb0
Disk Partitions:
usb0(type:FAT32)

Size : 131,072,000B(125MB)
Available size: 1,260,020B (1.2MB)

Qtech# show sd

Device: Mass Storage:
ID: 1
URL prefix: sd0
Disk Partitions:
SD(type:FAT32)

Size : 131,072,000B(125MB)
Available size: 1,260,020B (1.2MB)
```

USB Mass Storage Device is the name of the device.

URL means which prefix can be used by U disk/SD card to access U disk/SD card.

Size means the available space in U disk/SD card that can be accessed.

Available size means the remaining space in U disk/SD card.

11.3.2. Unplugging USB Device/SD Card

Before pulling out USB device/SD card, run the command on the CLI to uninstall the device in case system is using the USB device/SD card to avoid an error.

Command	Function
Qtech# **usb remove** *Device_ID*	Uninstalls the USB device with ID Device_ID

Command	Function
Qtech# **sd remove** *Device_ID*	Uninstalls the SD device with ID Device_ID

As shown above, IDO indicates a USB device, and ID1 indicates an SD card. The commands below can uninstall the corresponding USB device and SD card.

```
Qtech# usb remove 0
After the uninstall command is used, the system will print:
OK, now you can pull out the device 0.
*Jan  1 00:18:16: %USB-5-USB_DISK_REMOVED: USB Disk <Mass Storage> has been removed
from USB port 0!
Qtech# sd remove 1
After the uninstall command is used, the system will print:
OK, now you can pull out the device 1
Now, you can pull out the USB device/SD card.
```

Sometimes, it may lead to failure to uninstall the device for the device is being used. Wait a while, and then run the uninstall command to pull out the device.

Caution Be sure to uninstall the device first and then unplug the device to prevent the system error.

11.4. USB/SD Faults

When the system prints the following message:

```
*Jan 2 00:00:39: %USB-3-OHCI_ERR: USB1.0 controller is not available now.
```

USB/SD 1.0 controller is not available, while USB/SD card 2.0 is still available. In this case, reset the whole system to use corresponding version U disk/SD card.

When the system prints the following message:

```
*Jan 2 00:00:39: %USB-3-EHCI_ERR: USB2.0 controller is not available now.
```

USB/SD 2.0 controller is not available, while U disc/SD card 1.0 is still available. In this case, reset the whole system to use corresponding version U disk/SD card.

12. CONFIGURING SYSTEM MANAGEMENT

12.1. Basic System Management

12.1.1. Showing CPU Utilization

Use the **show cpu** command to show the total CPU utilization of the system and the CPU utilization of each process:

Command	Function
Qtech# **show cpu**	Shows CPU utilization.

By default, the device name is **Qtech**.

The following example shows the output result of this command.

```
Qtech#show cpu
=======================================
     CPU Using Rate Information
CPU utilization in five seconds: 25%
CPU utilization in one minute: 20%
CPU utilization in five minutes: 10%
  NO   5Sec   1Min   5Min   Process
  0     0%     0%     0%    LISR INT
  1     7%     2%     1%    HISR INT
  2     0%     0%     0%    ktimer
  3     0%     0%     0%    atimer
  4     0%     0%     0%    printk_task
  5     0%     0%     0%    waitqueue_process
  6     0%     0%     0%    tasklet_task
  7     0%     0%     0%    kevents
  8     0%     0%     0%    snmpd
  9     0%     0%     0%    snmp_trapd
 10     0%     0%     0%    mtdblock
 11     0%     0%     0%    gc_task
 12     0%     0%     0%    Context
 13     0%     0%     0%    kswapd
 14     0%     0%     0%    bdflush
 15     0%     0%     0%    kupdate
 16     0%     3%     1%    ll_mt
 17     0%     0%     0%    ll main process
 18     0%     0%     0%    bridge_relay
 19     0%     0%     0%    d1x_task
 20     0%     0%     0%    secu_policy_task
 21     0%     0%     0%    dhcpa_task
 22     0%     0%     0%    dhcpsnp_task
 23      0%      0%      0%     igmp_snp
 24      0%      0%      0%     mstp_event
 25      0%      0%      0%     GVRP_EVENT
 26      0%      0%      0%     rldp_task
 27      0%      2%      1%     rerp_task
 28      0%      0%      0%     reup_event_handler
 29      0%      0%      0%     tpp_task
 30      0%      0%      0%     ip6timer
 31      0%      0%      0%     rtadvd
 32      0%      0%      0%     tnet6
 33      2%      0%      0%     tnet
 34      0%      0%      0%     Tarptime
 35      0%      0%      0%     gra_arp
 36      0%      0%      0%     Ttcptimer
 37      8%      1%      0%     ef_res
 38      0%      0%      0%     ef_rcv_msg
 39      0%      0%      0%     ef_inconsistent_daemon
 40      0%      0%      0%     ip6_tunnel_rcv_pkt
 41      0%      0%      0%     res6t
 42      0%      0%      0%     tunrt6
 43      0%      0%      0%     ef6_rcv_msg
 44      0%      0%      0%     ef6_inconsistent_daemon
 45      0%      0%      0%     imid
 46      0%      0%      0%     nsmd
 47      0%      0%      0%     ripd
 48      0%      0%      0%     ripngd
 49      0%      0%      0%     ospfd
 50      0%      0%      0%     ospf6d
 51      0%      0%      0%     bgpd
 52      0%      0%      0%     pimd
 53      0%      0%      0%     pim6d
 54      0%      0%      0%     pdmd
 55      0%      0%      0%     dvmrpd
 56      0%      0%      0%     vty_connect
 57      0%      0%      0%     aaa_task
 58      0%      0%      0%     Tlogtrap
 59      0%      0%      0%     dhcp6c
 60      0%      0%      0%     sntp_recv_task
 61      0%      0%      0%     ntp_task
 62      0%      0%      0%     sla_deamon
 63      0%      3%      1%     track_daemon
 64      0%      0%      0%     pbr_guard
 65      0%      0%      0%     vrrpd
 66      0%      0%      0%     psnpd
 67      0%      0%      0%     igsnpd
 68      0%      0%      0%     coa_recv
 69      0%      0%      0%     co_oper
 70      0%      0%      0%     co_mac
 71      0%      0%      0%     radius_task
 72      0%      0%      0%     tac+_acct_task
 73      0%      0%      0%     tac+_task
 74      0%      0%      0%     dhcpd_task
 75      0%      0%      0%     dhcps_task
 76      0%      0%      0%     dhcpping_task
 77      0%      0%      0%     dhcpc_task
 78      0%      0%      0%     uart_debug_file_task
 79      0%      0%      0%     ssp_init_task
 80      0%      0%      0%     rl_listen
 81      0%      0%      0%     ikl_msg_operate_thread
 82      0%      0%      0%     bcmDPC
 83      0%      0%      0%     bcmL2X.0
 84      3%      3%      3%     bcmL2X.0
 85      0%      0%      0%     bcmCNTR.0
 86     0%     0%     0%     bcmTX
 87     0%     0%     0%     bcmXGS3AsyncTX
 88     0%     2%     1%     bcmLINK.0
 89     0%     0%     0%     bcmRX
 90     0%     0%     0%     mngpkt_rcv_thread
 91     0%     0%     0%     mngpkt_recycle_thread
 92     0%     0%     0%     stack_task
 93     0%     0%     0%     stack_disc_task
 94     0%     0%     0%     redun_sync_task
 95     0%     0%     0%     conf_dispatch_task
 96     0%     0%     0%     devprob_task
 97     0%     0%     0%     rdp_snd_thread
 98     0%     0%     0%     rdp_rcv_thread
 99     0%     0%     0%     rdp_slot_change_thread
100     4%     2%     1%     datapkt_rcv_thread
101     0%     0%     0%     keepalive_link_notify
102     0%     0%     0%     rerp_msg_recv_thread
103     0%     0%     0%     ip_scan_guard_task
104     0%     0%     0%     ssp_ipmc_hit_task
105     0%     0%     0%     ssp_ipmc_trap_task
106     0%     0%     0%     hw_err_snd_task
107     0%     0%     0%     rerp_packet_send_task
108     0%     0%     0%     idle_vlan_proc_thread
109     0%     0%     0%     cmic_pause_detect
110     1%     1%     1%     stat_get_and_send
111     0%     1%     0%     rl_con
112    75%    80%    90%     idle
```

As shown in the above list, the first three lines indicate the total CPU utilization in the last 5 seconds, 1 minute, and 5 minutes respectively, including the CPU utilization of LISRs, HISRs and tasks, followed by the CPU utilization of various processes. The parameters in the columns are described as follows:

- **No**: number
- **5Sec**: CPU utilization in the last 5 seconds
- **1Min**: CPU utilization in the last 1 minute
- **5Min**: CPU utilization in the last 5 minutes
- **Process**: process name

The first two lines indicate the CPU utilization of all LISRs and all HISRs respectively. All the lines starting from the third line indicate the CPU utilization of processes. The last line indicates the CPU utilization of idle processes. As with **System Idle Process** in the Windows operating system, it indicates an idle status. The above example shows that the CPU utilization of idle processes in the last 5 seconds is 75%, meaning that 75% of the CPU resources are available.

12.1.2. Configuring CPU Utilization Log Thresholds

Use the following command to configure CPU utilization log thresholds.

Command	Function
cpu-log log-limit *low_num high_num*	Configures the high and low thresholds for triggering CPU utilization logs.

By default, the high threshold is 100% and the low threshold is 90%.

The following example sets the low threshold to 70% and the high threshold to 80%.

```
Qtech# configure terminal                          // Enter the global configuration mode.
Qtech(config)# cpu-log log-limit 70 80     // Configure the thresholds for triggering
CPU logs.
```

If the CPU utilization is higher than 80%, the following information is displayed:

Oct 20 15:47:01 %SYSCHECK-5-CPU_USING_RATE_LOG: CPU utilization rate in one minute: 95%. rl_con occupied most CPU utilization rate: 94%.

If the CPU utilization is lower than 70%, the following information is displayed:

Oct 20 15:47:01 %SYSCHECK-5-CPU_USING_RATE_LOG: CPU utilization rate in one minute: 68%. rl_con occupied most CPU utilization rate: 60%.

Oct 20 15:47:01 %SYSCHECK-5-CPU_USING_RATE_LOG: The CPU utilization ratio has been decreased.

13. CONFIGURING SYSTEM MEMORY DISPLAY

13.1. Showing the Usage of System Memory

Use the **show memory** command to show the usage and status of system memory:

Command	Function
Qtech# **show memory**	Shows the usage of system memory.

The switch name is Qtech by default.

Below is the result of executing this command:

```
Qtech#show memory
System Memory Statistic:
  Free pages: 174164
    watermarks : min 2012, lower 4024, low 6036, high 9048
  System Total Memory : 1024MB, Current Free Memory : 740580KB
  Used Rate : 29%
```

The above information includes:

Parameter		Description
Free pages		The total free pages of all areas.
watermarks	min	Memory resources are extremely insufficient. It can only keep the kernel running. All application modules fails to run if the minimum watermark has been reached.
	lower	Memory resources are severely insufficient. One route protocol will auto-exit and release the memory if the lower watermark has been reached. For the details, see the **memory-lack exit-policy** command.
	low	Memory resources are insufficient. The route protocol will be in OVERFLOW state if the low watermark has been reached. In the overflow state, the routers do not learn new routes any more. The commands are not allowed to be executed when the memory lacks.
	high	There is plenty of memory resources. Each route protocol restores the state from OVERFLOW to normal.
System Total Memory		System total memory
System Free Memory		System free memory, including free pages space and all free space in the cache pool
Used Rate		Memory utilization rate

14. CONFIGURING MIB

14.1. MIB Lists

The followings are the supported standard MIBs:

- BRIDGE-MIB (RFC1493)
- EtherLike-MIB(RFC1643)
- IF-MIB(RFC2863)
- RFC1213-MIB
- RMON1-MIB(supports RMON etherStats, etherHistory,alarms, events)
- SNMPv2-MIB
- SNMPv3-MIB(supports USM, VACM)

The followings are the private MIBs:

- QTECH-AAA-MIB
- QTECH -ENTITY-MIB
- QTECH -RIP-MIB
- QTECH -MEMORY-MIB

You can use the **show snmp mib** command to view the supported MIBs in the current system:

```
Qtech# show snmp mib
sysDescr
sysObjectID
sysUpTime
sysContact
sysName
sysLocation
sysServices
sysORLastChange
ifNumber
ifEntry
ifEntry.ifIndex
ifEntry.ifDescr
ifEntry.ifType
ifEntry.ifMtu
ifEntry.ifSpeed
ifEntry.ifPhysAddress
ifEntry.ifAdminStatus
ifEntry.ifOperStatus
ifEntry.ifLastChange
ifEntry.ifInOctets
ifEntry.ifInUcastPkts
ifEntry.ifInNUcastPkts
ifEntry.ifInDiscards
ifEntry.ifInErrors
ifEntry.ifInUnknownProtos
ifEntry.ifOutOctets
ifEntry.ifOutUcastPkts
ifEntry.ifOutNUcastPkts
ifEntry.ifOutDiscards
ifEntry.ifOutErrors
ifEntry.ifOutQLen
ifEntry.ifSpecific
......
```

14.2. Obtaining the MIB Files

You can obtain the standard MIB information on the http://ietf.org/rfc.html.

You can obtain the private MIB information on the website.

15. CONFIGURING DEBUGGING IMPROVEMENT

15.1. Overview

Debugging information of devices, including output information of syslog, show commands and debug commands, can help post-sales or R&D engineers fast locate problems. However, it is inconvenient to use debugging information at present. For example, syslog information is buffered only in the memory. Once upon exception restart, previously saved syslog information will be lost, which is unfavorable to troubleshooting. In addition, a command for collecting all fault information is unavailable. This document describes how to configure debugging improvement based on the preceding problems.

15.1.1. One-key Collection

One-key collection comprises two parts:

The CLI **show tech-support** command can be used to display platform information in all scenarios.

The **show tech-support** <*module name*> command can be used to display module information in certain scenarios.

15.1.2. One-key Collection of Platform Information

Run the following command for one-key collection of platform information:

Command	**Description**
Qtech# **show tech-support**	Collects platform information using the one-key mode.

At present, the show command for one-key collection of platform information integrates the following commands:

1. System platform

----------------- show version -----------------

----------------- show running-config -----------------

----------------- show user -----------------

----------------- dir -----------------

----------------- show memory-----------------

----------------- show cpu -----------------

----------------- show task statistic-----------------

----------------- show timer statistic-----------------

----------------- show queue statistic----------------

----------------- show skbuffer statistic-------------

----------------- show environment--------------------

----------------- show version slot--------------------

----------------- show clock--------------------

----------------- show exception--------------------

2. Express forwarding platform

----------------- show core -----------------

----------------- show efb --------------------

----------------- show core interface all--------------------

3. Flow platform

----------------- show ip fpm statistics --------------------

----------------- show ip fpm counters--------------------

----------------- show ipv6 fpm statistics--------------------

4. Routing module

----------------- show ip route summary all--------------------

----------------- Show ip route count--------------------

----------------- show ip ref route statistic--------------------

----------------- show ip ref adjacency statistic-----------------

----------------- show ipv6 ref route statistic --------------------

----------------- show ipv6 route summary all-----------------

----------------- show ipv6 ref adjacency statistic-----------------

5. Interface

----------------- show interfaces -----------------

----------------- show ip interface brief -----------------

----------------- show ef-interfaces -----------------

------------------ show controllers e1 ------------------

In addition, the show command for platform information integrates the show commands for all modules. For the collection of module information, see the next section.

15.1.3. One-key Collection of Module Information

Run the following command for one-key collection of module information:

Command	Description
Qtech# **show tech-support** *<module name>*	Collects information of modules using the one-key mode.

At present, the command can be used to collect information of the following modules:

CE1/MLP/PPP/NAT/IPSEC/L2TP/BFD/QOS/RIP/OSPF/BGP/ETHERNET/MSTP/FR/PPPOE/CA/DLDP/MPLS

The show commands to be integrated for the modules are as follows:

CE1:

show interface

show controller e1

MLP:

show ppp mul

show ref ppp mul

NAT:

show ip nat translations

show ip nat statistics rule nouse/syn

IPSec:

show crypto isakmp policy

show crypto ipsec sa

show crypto ipsec transform-set

show crypto map

show crypto acl

show crypto detail

show crypto state

L2TP:

show vpdn

show vpdn tunnel

show vpdn session

BFD:

show bfd neighbor details

QoS:

show rate-limit

show policy-map

show queue cq/pq/rtpq/wfq

show traffic-shape queue

show traffic-shape statistics

RIP:

show ip protocol

show ip route

show ip rip

show ip rip database

show ip rip interface

show ip rip peer

show ip rip external

show ip route summary

OSPF:

show ip protocol

show ip route

show ip ospf neighbor detail

show ip ospf interface

show ip ospf database

show ip ospf sham-links

show ip ospf virtual-links

show ip ospf spf

show ip ospf topo

show ip ospf border-routers

show ip ospf summary-address

show ip ospf route

show ip ospf route summary

BGP:

show ip protocol

show ip route

show ip bgp neighbor

show ip bgp summary

show ip bpg all summary

show bgp all summary

Ethernet/MSTP:

show arp

FR:

show fr map

show fr lmi

show fr pvc

PPPoE:

show pppoe session

CA:

show crypto pki trustpoints

show crypto pki certificates

DLDP:

show dldp interface

MPLS:

show ip vrf interfaces

show ip ref adjacency

show ip ref mpls packet debug-buf

show mpls forwarding-table

16. CONFIGURING FLOW PLATFORM

16.1. Understanding the Flow Platform

16.1.1. Overview

The flow platform achieves a perfect combination of services and performance, because services (such as QoS, ACL, NAT, and PBR) enabled on a service processing board have nearly no impact on forwarding performance.

16.1.2. Basic Concepts

Service packets based on Layer 3 usually can be abstracted as flows. A flow identifies a sequence of packets from a specific source to a specific destination. Generally, a flow is indentified by a sextuplet, which includes a source address, a destination address, a source port ID, a destination port ID, a transport layer protocol, and VRF.

16.1.3. Working Principle

When the first packet is forwarded through an entire routing process and service processing, the sextuplet of the packet identifies a flow. If a packet received later matches the sextuplet, it is forwarded in the same way as the first packet. The sextuplet and an outbound interface of the packet compose a flow entry. Packets that match the flow entry are forwarded directly without experiencing the entire routing process or service processing any longer. Therefore, enabled services (such as QoS, ACL, NAT, and PBR) have nearly no impact on forwarding performance.

16.1.4. Protocols and Standards

None.

16.1.5. Typical Application

When services (such as QoS, ACL, NAT, and PBR) are enabled, the flow platform is automatically enabled to accelerate service processing.

16.2. Configuring the IPv4 Function of the Flow Platform

16.2.1. Default Configuration

The following table describes the default configuration of the flow platform.

Feature	Default Setting
Flow platform	Disabled
Flow overflow alarm interval of the flow platform	5 seconds
Flow overflow alarm threshold of the flow platform	95%
Maximum number of flow entries in the IPv4 flow table	180,223

16.2.2. Prerequisites

When service modules (such as QoS, ACL, NAT, and PBR service modules) that rely on the flow platform are configured, the flow platform is automatically enabled.

16.2.3. Configuration Steps

Step	Configuration Task	Description
Step 1	Configure service modules.	(Mandatory) You can configure service modules such as QoS, ACL, NAT, and PBR.
Step 2	Configure the maximum number of flow entries in the IPv4 flow table.	(Optional) This step is performed if you need to change the memory occupied by the flow table.
Step 3	Configure the IPv4 flow overflow alarm interval of the flow platform.	(Optional) This step is performed if you need to change the flow overflow alarm interval.
Step 4	Configure the IPv4 flow overflow alarm threshold of the flow platform	(Optional) This step is performed if you need to change the flow overflow alarm threshold.

16.2.4. Configuring ACL to Enable the Flow Platform

Command	**Function**
Qtech(config)# **access-list** *id* { **deny | permit** } { src *src-wildcard* | **host** *src* | **any** } [**time-range** *tm-rng-nam e*] [**log**]	Defines an ACL.
Qtech(config)# **interface** *interface*	Specifies the interface to which the ACL is applied.
Qtech(config-if)# **ip access-group** *id* { in | out } [**unreflect**]	Applies the ACL to the specific interface.

The following example enables the ACL function on the port GigabitEthernet 0/1.

```
Qtech# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Qtech(config)# access-list 101 permit tcp 192.168.12.0 0.0.0.255 any
Qtech(config)# interface GigabitEthernet 0/1
Qtech(config-if)# ip address 192.168.12.1 255.255.255.0
Qtech(config-if)# ip access-group 101 in
```

16.2.5. Configuring the Maximum Number of Flow Entries in the IPv4 Flow Table

Command	**Function**
Qtech(config)# **ip fpm flow max-entries** *flow-number*	Configures the maximum number of flow entries in the IPv4 flow table.

The following example configures the maximum number of IPv4 flow entries as 120,000.

```
Qtech# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Qtech(config)# ip fpm flow max-entries 120000
FPM subsystem is reinitializing...
Qtech(config)#*Oct  6 17:35:21: %FPM-5-RESTARTED: The device IPv4 flow max-entries changed.
```

16.2.6. Configuring the IPv4 Flow Overflow Alarm Interval of the Flow Platform

Command	Function
Qtech(config)# **ip fpm flow alert interval** *seconds*	Configures the IPv4 flow overflow alarm interval of the flow platform.

The following example configures the IPv4 flow overflow alarm interval of the flow platform as 120s.

```
Qtech# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Qtech(config)# ip fpm flow alert interval 120
```

16.2.7. Configuring the IPv4 Flow Overflow Alarm Threshold of the Flow Platform

Command	Function
Qtech(config)# **ip fpm flow alert threshold** *percent-value*	Configures the IPv4 flow overflow alarm threshold of the flow platform.

The following example configures the IPv4 flow overflow alarm threshold of the flow platform as 80%.

```
Qtech# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Qtech(config)# ip fpm flow alert threshold 80
```

16.3. Configuring the IPv6 Function of the Flow Platform

16.3.1. Default Configuration

The following table describes the default configuration of the flow platform.

Feature	Default Setting
Flow platform	Disabled
Flow overflow alarm interval of the flow platform	5 seconds
Flow overflow alarm threshold of the flow platform	95%
Maximum number of flow entries in the IPv6 flow table	81,920

16.3.2. Prerequisites

When service modules (such as QoS v6, ACLv6, NAT64, and PBRv6 service modules) that rely on the flow platform are configured, the flow platform is automatically enabled.

16.3.3. Configuration Steps

Step	**Configuration Task**	**Description**
Step 1	Configure service modules.	(Mandatory) You can configure service modules such as QoS v6, ACLv6, NAT64, and PBRv6.
Step 2	Configure the maximum number of flow entries in the IPv6 flow table.	(Optional) This step is performed if you need to change the memory occupied by the flow table.
Step 3	Configure the IPv6 flow overflow alarm interval of the flow platform.	(Optional) This step is performed if you need to change the flow overflow alarm interval.
Step 4	Configure the IPv6 flow overflow alarm threshold of the flow platform.	(Optional) This step is performed if you need to change the flow overflow alarm threshold.

16.3.4. Configuring ACLv6 to Enable the Flow Platform

Command	**Function**
Qtech(config)# **ipv6 access-list** *name*	Enters ACL configuration mode.
Qtech (config-ipv6-nacl)#[*sn*] {**permit** | **deny** } *port* { *src-ipv6-prefix/prefix-len* | **host** *src-ipv6-addr* | **any** } { *dst-ipv6-pfix/pfix-len* | **any** | **host** *dst-ipv6-addr* } [**dscp** *dscp*] [**flow-label** *flow-label*] [**fragments**] [**range** lower upper] [**time-range***tm-rng-name*]	Adds an entry to the ACL. For details about the command, see the Command Reference.
Qtech(config-exp-nacl)# **exit**	Exits ACL configuration mode.
Qtech(config)# **interface** *interface*	Specifies the interface to which the ACL is applied.
Qtech(config-if)# **ipv6 traffic-filter** *name* { **in** | **out** }	Applies the ACL to the specific interface.

The following example enables the ACL function on the port GigabitEthernet 0/1.

```
Qtech# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Qtech(config)# ipv6 access-list v6-list
Qtech(config-ipv6-acl)# permit ipv6 2001:db8:1::1/64 any

Qtech(config-ipv6-acl)# deny ipv6 any any

Qtech(config-ipv6-acl)# exit

Qtech(config)# interface GigabitEthernet 0/1

Qtech(config-if)# ipv6 traffic-filter v6-list in
```

16.3.5. Configuring the Maximum Number of Flow Entries in the IPv6 Flow Table

Command	**Function**
Qtech(config)# **ipv6 fpm flow max-entries** *flow-number*	Configures the maximum number of flow entries in the IPv6 flow table.

The following example configures the maximum number of flow entries in the IPv6 flow table as 70,000.

```
Qtech# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Qtech(config)# ipv6 fpm flow max-entries 70000

FPM subsystem is reinitializing...

Qtech(config)#*Oct  6 17:35:21: %FPM-5-RESTARTED: The device IPv6 flow max-entries
changed.
```

16.3.6. Configuring the IPv6 Flow Overflow Alarm Interval of the Flow Platform

Command	**Function**
Qtech(config)#**ipv6 fpm flow alert interval** *seconds*	Configures the IPv6 flow overflow alarm interval of the flow platform.

The following example configures the IPv6 flow overflow alarm interval of the flow platform as 120s.

```
Qtech# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Qtech(config)#ipv6 fpm flow alert interval 120
```

16.3.7. Configuring the IPv6 Flow Overflow Alarm Threshold of the Flow Platform

Command	**Function**

Qtech(config)# **ipv6 fpm flow alert threshold** *percent-value*	Configures the IPv6 flow overflow alarm threshold of the flow platform.

The following example configures the IPv6 flow overflow alarm threshold of the flow platform as 80%.

```
Qtech# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Qtech(config)# ipv6 fpm flow alert threshold 80
```

16.4. Monitoring and Maintaining the Flow Platform

Command	**Function**
Qtech#**clear ip fpm counters**	Clears IPv4 packet statistics of the flow platform.
Qtech#**clear ip fpm flows**	Clears the IPv4 flow table of the flow platform.
Qtech(config)# **ip fpm flow alert interval** *seconds*	Configures the IPv4 flow overflow alarm interval of the flow platform.
Qtech(config)# **ip fpm flow alert threshold** *percent-value*	Configures the IPv4 flow overflow alarm threshold of the flow platform.
Qtech(config)#**ip fpm flow max-entries** *flow-number*	Configures the maximum number of flow entries in the IPv4 flow table.
Qtech(config)#**ip fpm frq** *queue-number*	Configures the number of concurrent IPv4 fragment reassembly queues.
Qtech(config)**ip fpm session filter** *acl-number*	Protects the IPv4 flow table against attacks.
Qtech#**show ip fpm counters**	Displays IPv4 packet counters of the flow platform.
Qtech#**show ip fpm flows** [**filter** *protocol-number src-ip src-mask dst-ip dst-mask*]	Displays the IPv4 flow table.
Qtech#**show ip fpm statistics**	Displays IPv4 statistics of the flow platform.
Qtech#**show ip fpm users**	Displays the number of IPv4 user connections of the flow platform.

Qtech#**clear ipv6 fpm flows**	Clears the IPv6 flow table of the flow platform.
Qtech#**clear ipv6 fpm statistics**	Clears IPv6 statistics of the flow platform.
Qtech(config)**ipv6 fpm flow alert interval** *seconds*	Configures the IPv6 flow overflow alarm interval of the flow platform.
Qtech(config)**ipv6 fpm flow alert threshold** *percent-value*	Configures the IPv6 flow overflow alarm threshold of the flow platform.
Qtech(config)**ipv6 fpm flow max-entries** *flow-number*	Configures the maximum number of flow entries in the IPv6 flow table.
Qtech(config)**ipv6 fpm frq** *queue-number*	Configures the number of concurrent IPv6 fragment reassembly queues.
Qtech(config)**ipv6 fpm session filter** *acl-name*	Protects the IPv6 flow table against attacks.
Qtech#**show ipv6 fpm statistics**	Displays IPv6 statistics of the flow platform.
Qtech#**show ipv6 fpm statistics fragment**	Displays IPv6 fragment reassembly statistics of the flow platform.
Qtech#**show ipv6 fpm flows** [**filter** *protocol-number src-ip dst-ip*]	Displays the IPv6 flow table.

16.5. Configuration Examples

16.5.1. Example of IPv4 Configuration of the Flow Platform

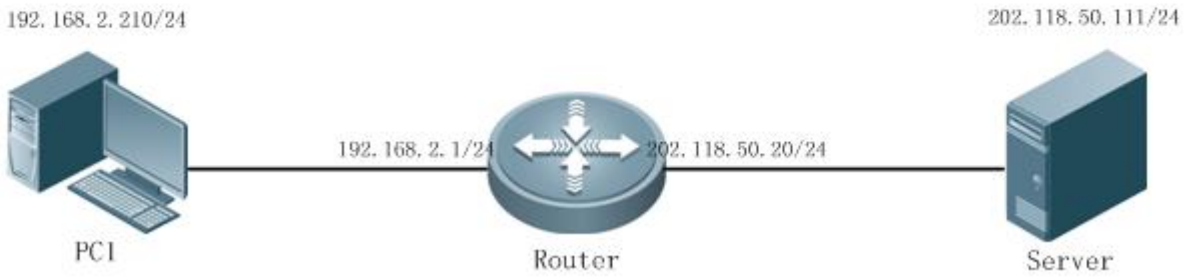
Networking Requirements

As shown in Figure 1-1, a router is connected to a PC and a server. Configure the ACL function on the router to implement the following functions:

- Enable the flow platform on the router.
- Adjust the maximum number of flow entries in the IPv4 flow table of the flow platform on the router.
- Protect the IPv4 flow table on the router by allowing flow establishment for only IP packets from the network segment 192.168.2.0/24 instead of packets from other network segments.
- Configure the IPv4 flow overflow alarm interval as 30s and the IPv4 flow overflow alarm threshold as 80% on the router.

Networking Topology

Figure 1-1 Example of IPv4 Configuration of the Flow Platform



Configuration Tips

None.

Configuration Steps

- Apply the ACL function to the loopback interface on the router to enable the flow platform.

```
Qtech # configure terminal

Qtech(config)# ip access-list standard 1

Qtech(config-std-nacl)# permit any

Qtech(config-std-nacl)# exit

Qtech(config)# interface Loopback 0

Qtech(config-if-Loopback 0)# ip access-group 1 in

Qtech(config-if-Loopback 0)# exit
```

- Configure the maximum number of flow entries in the IPv4 flow table as 100,000.

```
Qtech(config)#  ip fpm flow max-entries 100000

FPM subsystem is reinitializing...

Qtech(config)#*Oct  6 17:35:21: %FPM-5-RESTARTED: The device IPv4 flow max-entries
changed.
```

- Configure an ACL numbered 2 on the router so as to protect the IPv4 flow table against attacks.

```
Qtech(config)#  ip access-list standard 2

Qtech(config-std-nacl)# permit 192.168.2.0 0.0.0.255

Qtech(config-std-nacl)# exit

Qtech(config)# ip fpm session filter 2
```

- Configure the IPv4 flow overflow alarm interval as 30s and the IPv4 flow overflow alarm threshold as 80% on the router.

```
Qtech(config)# ip fpm flow alert interval 30

Qtech(config)# ip fpm flow alert threshold 80
```

Verification

Run the **ping 202.118.50.111 ntimes 1** command on PC 1.

```
C:\>ping 192.168.50.1 -n 1

Pinging 202.118.50.111 with 32 bytes of data:

Reply from 202.118.50.111: bytes=32 time=1ms TTL=64

Ping statistics for 202.118.50.111:

    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

    Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

Run the **show ip fpm flows** command on the router. A flow entry is generated during the ping operation.

Construct a packet with a source IP address in another network segment on PC 1, and send the packet to 202.118.50.111. No corresponding flow entry can be seen in the flow table on the router.

16.5.2. Example of IPv6 Configuration of the Flow Platform

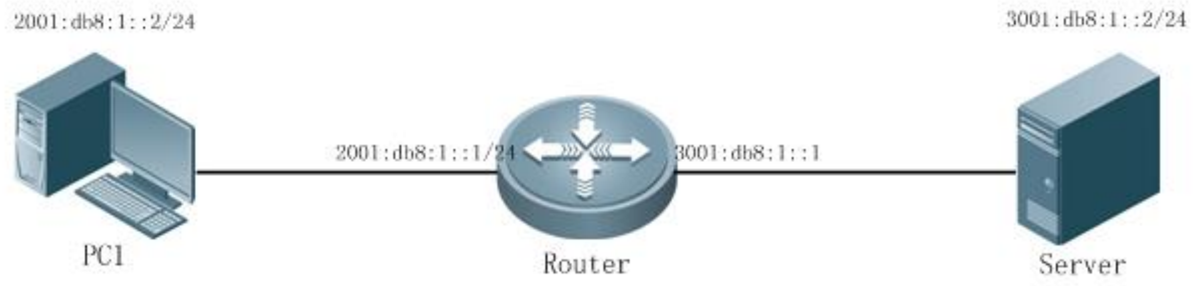
Networking Requirements

As shown in Figure 1-2, a router is connected to a PC and a server. Configure the ACL function on the router to implement the following functions:

- Enable the flow platform on the router.
- Adjust the maximum number of flow entries in the IPv6 flow table of the flow platform on the router.
- Protect the IPv6 flow table on the router by allowing flow establishment for only IP packets from the network segment 2001:db8:1::2/64 instead of packets from other network segments.
- Configure the IPv6 flow overflow alarm interval as 30s and the IPv6 flow overflow alarm threshold as 80% on the router.

Networking Topology

Figure 2-1 Example of IPv6 Configuration of the Flow Platform



Configuration Tips

None

Configuration Steps

1) Apply the ACL function to the loopback interface on the router to enable the flow platform.

```
Qtech # configure terminal
Qtech(config)# ip access-list standard 1
Qtech(config-std-nacl)# permit any
Qtech(config-std-nacl)# exit
Qtech(config)# interface Loopback 0
Qtech(config-if-Loopback 0)# ip access-group 1 in
Qtech(config-if-Loopback 0)# exit
```

aa. Configure the maximum number of flow entries in the IPv6 flow table as 100,000.

```
Qtech(config)# ipv6 fpm flow max-entries 100000
FPM subsystem is reinitializing...
Qtech(config)#*Oct  6 17:35:21: %FPM-5-RESTARTED: The device IPv6 flow max-entries
changed.
```

bb. Configure an IPv6 ACL named "virus_filter" on the router to protect the IPv6 flow table against attacks.

```
Qtech(config)# ipv6 access-list virus_filter
Qtech(config-ipv6-acl)# permit ipv6 2001:db8:1::/64 any
Qtech(config-ipv6-acl)# permit icmp 2001:db8:1::/64 any
Qtech(config)# ipv6 fpm session filter virus_filter
```

cc. Configure the IPv6 flow overflow alarm interval as 30s and the IPv6 flow overflow alarm threshold as 80% on the router.

```
Qtech(config)# ipv6 fpm flow alert interval 30

Qtech(config)# ipv6 fpm flow alert threshold 80
```

Verification

Run the **ping 3001:db8:1::2 ntimes 1** command on PC 1.

```
Qtech#ping 3001:db8:1::2 ntimes 1

Sending 1, 100-byte ICMP Echoes to 3001:db8:1::2, timeout is 2 seconds:

  < press Ctrl+C to break >

!

Success rate is 100 percent (1/1), round-trip min/avg/max = 10/10/10 ms
```

Run the **show ipv6 fpm flows** command on the router. A flow entry is generated during the ping operation.

Construct a packet with a source IP address in another network segment on PC 1, and send the packet to 3001:db8:1::2. No corresponding flow entry can be seen in the flow table on the router.