Security Configuration

Table of Contents

Chapter 1 AAA Configuration

1.1 AAA Overview

The access control is used to control users for accessing the router or the Network Access Server (NAS), and limit service kinds they can use. It provides authentication, authorization and accounting for enhancing the network security.

1.1.1 AAA Security Service

AAA is an architecture that uses a consistent method to configure three independent security functions. It provides the modular method to complete the following services:

- Authentication

  provides the method of identifying users, includes the enquiry of username and password and makes encryption according to the security protocols you select.

  Authentication is the method to identify users before accepting their access to the network and network services. You configure AAA authentication through the definition of a named list of authentication methods, and then apply that list to various interfaces. The method list defines the authentication types and their order of execution; any defined authentication method should be applied to a specific interface before it will be performed. The only exception is the default method list (named default). If no other method list is defined, the default method list is automatically applied to all interfaces. The definition of other method list will replace the default method list. For detailed information about all authentication configurations, please refer to “authentication configuration”.

- Authorization

  Provides a method of remote access control to restrict the service rights of the users.

  AAA authorization functions through a set of attributes of the users. These attributes described what rights are awarded for the users. These attributes are compared to the information in the database for a specific user, and the result is returned to AAA, in order to determine the actual rights of the user. This database can be put on the local access server or router, or on the remote RADIUS or TACACS+ security servers. The remote security servers such as RADIUS and TACACS+ make authorization to the users through their attribute value (AV) pairs, which define the authorized rights. All authorization methods should be defined through AAA. Like authentication, you define an authorization method list first, and then apply it to various interfaces. For detailed information of using AAA to make authorization configuration, please refer to “authorization configuration”.

- Accounting

  Provides a kind of method to collect user service information and forward it to the security server. This information can be used for billing, auditing and reporting, like user tag, start and stop time, command executed, the number of data packets and bytes.

Accounting function can not only trace the service users are accessing, but also trace the network resources they are consuming at the mean time. When the accounting function of AAA is activated, the network access server reports the activity of the user to TACACS+ or RADIUS server in the form of accounting. Each account includes account of attribute value pair, and is saved on the security server. These data can be used for network management, customer account list or audit analysis. Like authentication and authorization, it should first define an accounting method list, and then applies this list to various interfaces. For detailed information about using AAA for accounting configuration, please refer to “Account configuration”.

1.1.2 Advantages of AAA

AAA provides the following advantages:

- Flexibility and easy to control
- Easily update
- Standardized authentication methods, such as RADIUS, TACACS+
- Multiple backup systems

1.1.3 Principles of AAA

AAA is used to dynamically configure the authentication or authorization type based on every connection (every customer) or every service (for example, IP, IPX or VPDN). It defines the authentication and authorization type by creating method lists, then applying these method lists to specific services or interfaces.

1.1.4 Method List

An authentication method list defines various methods used to identify the users. The administrators can configure one or more protocols in the method list. So, even if the previous authentication method failed, it is guaranteed to have a backed-up authentication method. First, use the listed first method to identify users. If this method receives no response, select the next authentication method in that list; this process will continue until all listed authentication methods are used to guarantee successful authentication, or the resource of the authentication method list are used up, in which case the authentication fails.

Notes:

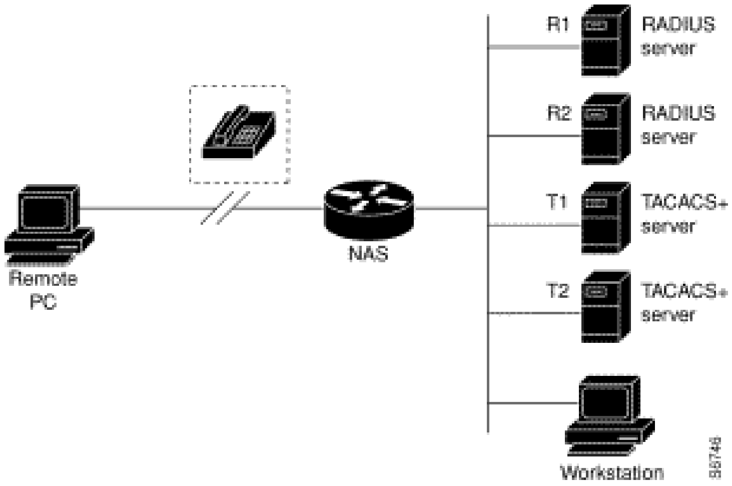
Only when the previous authentication method makes no response may you try to use the next method to make authentication. As long as authentication fails at any point-----that is to say, the response from the security server or local username database denies the access of the user access-------the authentication process stops and no other authentication methods will be tried.

To configure authentication, you should first define a named authentication method list and then apply this list to various interfaces. This method list defines the authentication types to be performed, and the order of their execution; any defined authentication method list, before performed, should be applied to a specific interface. The only exception is the default method list (default). The default method list will automatically be applied to all the interfaces except when the interface clearly quotes other method list. Then this method list will replace the default method list.

A method list is the list of the authentication methods orderly queried when identifying users. In the method list, you can designate one or more security protocols, thus guaranteeing that there is an authentication system as the backup in case the first method encounters failure. Our router software uses the first authentication method in the method list to identify the users; if this method receives no response, it will automatically use the next method in the list. This process will continue until one of the methods is successful in authentication, or all the methods are used up.

Notice this is vital, that is, our router software only tries the authentication method listed next when the pervious method receives no response. If the authentication fails in any point during this process, which is, the response of security server or local user database is the denial of the access of the user, then the authentication process stops, and will not try other authentication method.

Figuer 1-1 a representational AAA network configuration shows a representational AAA network configuration which includes four security servers, R1 and R2 are RADIUS servers, T1 and T2 are TACAC+ servers.



Figuer 1-1 a representational AAA network configuration

Suppose the system administrator decides to apply the same authentication method to all interfaces to identify the connections based on PPP protocol in his/her security scheme: first R1 will be connected for authentication information, then if R1 does not respond, connects R2, if R2 does not respond, connects T1, if T1 does not respond, connects T2, if all designated servers do not respond, the authentication work is forwarded to the local username database of the access server. When the remote user tries to enter the network through dial-up method, the network access server first queries the related authentication information on R1, if the user is legal after authentication, it sends a PASS reply to the network access server, so as to permit the user to access the server. If the reply is FAIL, this user is denied and the dialogue is ended. If R1 does not respond, the network access server will consider it as an ERROR, and queries the related authentication information on R2. This mode continues to function in the remaining methods, until the user is accepted or denied or the dialogue is ended.

Notes:

This item is quite important to remember. A “FAIL” reply is totally different from an “ERROR”. A “FAIL” means that the user does not meet the required standards included in the authentication database to be successfully authenticated. The authentication ends with a “FAIL” reply. An “ERROR” means that this security server does not give response to the authentication query. Only when AAA finds error will it select the next authentication method defined in the authentication method list.

To realize that, the system administrator should input the following command to create a default method list: aaa authentication ppp default radius local.

In this example, default is the name of the method list. The protocols included in this method list are listed after the name, in the order they are to be queried. The default list is automatically applied to all the interfaces.

When the remote user tries to enter the network through the dial-up method, the network access server first queries related authentication information on R1. If the user is legal after authentication, R1 sends a PASS reply to the network access server, so as to permit the user to access the server. If R1 return a FAIL reply, this user is denied and the dialogue is ended. If R1 does not respond, the network access server will consider it as an error, and queries the related authentication information on R2. This mode continues to function in the following methods, until the user is accepted or denied or the dialogue is ended.

This item is quite important to remember, “FAIL” reply is totally different from “ERROR”. “FAIL” means the user does not meet the required standards included in the authentication database to be successfully authenticated. The authentication ends with a “FAIL” reply. “ERROR” means this security server does not give response to the authentication query. Only when AAA detects error will it select the next authentication method in the authentication method list.

Suppose the system administrator only wants to apply the method list to a specific interface or set of interfaces. In this case, the system administrator should create a non-default method list and apply this list to an applicable interface. The following example demonstrates the process of how the system administrator implements an authentication method to be applied only to asynchronous interfaces:

```
aaa authentication ppp default radius local
aaa authentication ppp async0 radius tacacs+ local none
interface async 0/0
ppp authentication chap async0
```

In this example, async0 is the name of the method list. The authentication protocols included in this method list is orderly listed after it, and the protocols may be used in order. After the method list has been created, it is applied to an applicable interface. NOTICE, the method list name in the aaa authentication command and ppp authentication command should match.

1.1.5 Configuration Process

First, you should decide which type of security scheme you want to implement. You should evaluate the security risks in your network, and set up appropriate methods to prevent unauthorized login and attack.

After understanding the basic process related to configuration, configuring AAA is relatively easy. Follow the following steps when using AAA to configure security on a router or access server of our company:

- If you decide to use a security server, first configure security protocol parameters, such as RADIUS, TACACS+.
- Use the “AAA authentication” command to define the method list for authentication.
- If required, apply this method list to a specific interface or line.
- Use Commandaaa authorization to authorize configuration (optional).。
- Use Commandaaa accounting to authorize configuration (optional).

1.2 AAA Configuration Procedure

The security solution must be decided before AAA configuration. Users have to assess security risks in their network and then find proper methods to prevent the unauthorized logon and the attack.

1.2.1 AAA Configuration Procedure Overview

Before you configure AAA, you need know the basic configuration procedure. To do AAA security configuration on OUR routers or access servers, perform the following steps:

- Configure the security protocol parameter such as RADIUS and TACACS+ before the security server is used.
- Run **aaa authentication** to define the authentication method list.
- If necessary, apply the authentication method list to a specific interface or line.
- Run **aaa authorization** to configure the authorization, which is an optional step.
- Run **aaa accounting** to record the configuration procedure, which is also an optional step.

1.3 AAA authentication configuration tast list

- Use AAA to configure login authentication
- Use AAA to perform PPP authentication
- Turn on password protection when entering into privilege level
- Change the character string while prompted to input password
- Establish local authentication database

1.4 AAA Authentication Configuration Tast

1.4.1 Using AAA to Configure Login Authentication

AAA security service makes the use of various authentication methods easier. Whatever login method used, the aaa authentication command is used to enable AAA authentication. With the aaa authentication login command, you create one or several authentication method lists that are used at login. Use line configuration command “login authentication” to apply these lists. To configure, use the following commands beginning in global configuration mode:

Command	Purpose
aaa authentication login {**default** | *list-name*}*method1* [*method2...*]	Create a global authentication list.
line [**aux** | **console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter line configuration mode.
login authentication {**default** | *list-name*}	Apply the authentication list to a line or set of lines.

Key word “list-name” is a character string used to name the list you are creating. Key word “method” designates the actual method adopted in the designated authentication process. Only when the previous method returns with ERROR will it start to use other authentication method. If the previous method returns with FAIL, no other methods will be used. To specify that even if all methods return with ERROR, the user still can successfully login, define “none” as the last authentication method in the command line. For example: even if TACACS+ server returns ERROR, login can still be successful, please use the following command:

```
aaa authentication login default tacacs+ none
```

A default list can be created by using default parameter. The default list is automatically applied to all the interfaces. For example: To designate RADIUS as the default method for user authentication at login, use the following command:

```
aaa authentication login default radius
```

Notes:

As key word “none” enables all users logging in to successfully authenticate, it should be used only as a backup authentication method.

The following table listed all the currently supported login authentication methods:

Key word	Description
enable	Use “enable” password to authenticate
group	Use server group to authenticate
group-restrict	Use server group to authenticate, but when the user designates a certain server, this server group is invalid.
line	Use line password to authenticate
local	Use local database to authenticate
local-case	Use local username database to authenticate (case sensitive for the username)

none	Unconditionally pass the authentication
radius	Use RADIUS authentication
tacacs+	Use TACACS+ authentication

1. Login authentication using enable password

Command	Purpose
aaa authentication login default enable	Use the “enable” key word in the “aaa authentication login” command to designate “enable” password as the method of login authentication. For example: to specify “enable” password as the user authentication method at login.

2. Login authentication using line password

Command	Purpose
aaa authentication login default line	Use the “line” method keyword in “aaa authentication login” command to designate line password as the login authentication method. For example, to specify line password as the user authentication method at login, but do not define any other methods ,Before being able to use line password as the login authentication method, you need define a line password.

3. Login authentication using local password

Command	Purpose
aaa authentication login default local	Use the “local” method key word in the “aaa authentication login” command to designate local username database as the login authentication method. For example, to specify local username database as the method of user authentication at login when no other methods are defined ,for detailed information about adding users to the local username database, please refer to “establishing local authentication database”.

4. Login authentication using RADIUS

Command	Purpose
aaa authentication login default radius	Use the “radius” method key word in the “aaa authentication login” command to designate RADIUS as the login authentication method. For example, to specify RADIUS as the method of user authentication at login when no other methods are defined, Before being able to use RADIUS as the login authentication method, you should first configure RADIUS service.

5. Login authentication using TACACS+

Command	Purpose
aaa authentication login default tacacs+	Use the "TACACS+" method key word in the "aaa authentication login" command to designate TACACS+ as the login authentication method. For example, to specify TACACS+ as the method of user authentication at login when no other methods are defined, Before being able to use TACACS+ as the authentication method, you should first configure TACACS+ service.

1.4.2 Using AAA to Perform PPP Authentication

Many users access the network access servers through dialup via asynchronous or ISDN. AAA security service makes the authentication while running mass PPP on serial interface easier. Whatever PPP authentication method you decide to use, you can use the aaa authentication ppp command to turn on AAA authentication. When configuring, use the following commands in global configuration mode:

Command	Purpose
aaa authentication ppp {**default** | *list-name*} *method1* [*method2...*]	Create a local authentication list.
interface *interface-type number*	Enter interface configuration mode for the interface to which authentication list will be applied.
ppp authentication {**chap** | **pap** | **chap pap** | **pap chap**} {**default** | *list-name*}	Apply the authentication list on a line or set of lines.

With the "aaa authentication" command, you can create one or several lists of authentication method. These lists are used when the user begins to run PPP. These lists are applied using the "ppp authentication" configuration command. To create a default list, you may use the "default" key word followed by the method to be used in default situations. For example, to designate local username database as the default authentication method, input the following command line:

aaa authentication ppp default local

The "list-name" key word is any character strings used to name the created list. The "method" key word designates the actual method the authentication uses. Only when the previous method returns with "ERROR" will it use other authentication methods. If the previous method returns with "FAIL", it will not use other authentication methods. If you want to designate that, even if all methods return with "ERROR", the user could still be successfully authenticated, you can simply designate "none" as the last authentication method. In the following example, even if TACACS+ server returns with "ERROR", authentication will succeed; just input the following command line:

aaa authentication ppp default tacacs+ none

Notes:

As the "none" key word enables all users logging in to successfully authenticate, this key word should be used as a backup authentication method.

The following table listed the usable authentication methods for PPP:

Key word	Desription
group	Use server group to authenticate.
group-restrict	Use server group to authenticate, but when user designate to use a certain server, this server group becomes invalid .
local	Use local username database to authenticate.
local-case	Use local username database to authenticate (case sensitive).
none	Unconditional pass.
radius	Use RADIUS to authenticate.
tacacs+	Use TACACS+ to authenticate.

1. PPP authentication using local password

Command	Purpose
aaa authentication ppp default local	In the "aaa authentication ppp" command, use the "local" key word to designate local username database for authentication. For example, to designate local username database as the authentication method on lines running PPP when no other methods are needed.

2. PPP authentication using RADIUS

Command	Purpose
aaa authentication ppp default radius	In the "aaa authentication ppp" command, use the "RADIUS" key word as the authentication method while running PPP. For example, to designate RADIUS as the user authentication method when no other methods are defined, While using RADIUS as the PPP authentication method, it is required to configure RADIUS service.

3. PPP authentication using TACACS+

Command	Purpose
aaa authentication ppp default tacacs+	In the "aaa authentication ppp" command, use the "TACACS+" key word to designate TACACS+ as the authentication method for interfaces running PPP. For example, to designate TACACS+ as the user authentication method when no other methods are defined;Before using TACACS+ as the PPP authentication method, it is required to configure TACACS+ service first.

1.4.3 Turning on Password Protection when Entering into Privilege Level

Use the "aaa authentication enable default" command to create an authentication method list.These methods decide whether a user could execute the commands in privilege EXEC level. Four authentication methods can be designated at most. Only when the previous method returns with "ERROR", will it use other authentication methods. If the previous method returns with “FAIL”, then it will not use other authentication methods. If you want to designate that, even if all methods return with “ERROR”, the user could still be successfully authenticated, you can simply designate “none” as the last authentication method. When configuring, use the following command in global configuration mode:

Command	Purpose
aaa authentication enable default *method1* [*method2...*]	Enable password authentication when the user enters privileged EXEC level

The ”method” key word designates the actual method used during the authentication process. Use it according to the order of input while authenticating.

The following table lists the supported password protection authentication method:

Key word	Description
enable	Use password “enable” to authenticate
group	Use server group to authenticate
group-restrict	Use server group to authenticate, but when user designate to use certain server, this server group becomes invalid
line	Use line password to authenticate
none	Unconditional pass
radius	Use RADIUS to authenticate
tacacs+	Use TACACS+ to authenticate

While authentication method “enable” is configured to provide remote authentication (that is, when the “group, group-restrict, radius or tacacs+” key words are configured), the username of using RADIUS to authentication is different from the one using TACACS+, and will be separately introduced below:

1. Enable authentication using RADIUS

The username to be authenticated is $*level*$. “level” indicates the privilege level the user wants to enter, which is the number of the privilege level after the command “enable”. For example, if certain user wants to enter privilege level 7, he/she enters the command “enable7”. If currently RADIUS is configured to authenticate, then the username handed to Radius Server is 7. By default, the privilege level that “enable” can enter is 15, which indicates that while using RADIUS to authenticate, the username handed to Radius Server is 15. This requires configuring username and password in advance on Radius Server. The thing to be pointed out is that: in the Radius Server user database, the service-type used for privilege authentication user is 6, which is Admin-User.

2. Enable authentication using TACACS+

The username used for “enable” authentication is the one used to log in the router. For example, if the username that a certain user inputted to log in the router is “chen”, then the username used to making “enable” authentication is also “chen”. If the user is not requested to be authenticated or to input the username while logging in the router, the username is “DEFAULT” after successfully logged in, which requires the corresponding configuration in the TACACS+ Server user database.

1.4.4 Configuring the Message Banner of AAA Authentication

The banner of configurable, personal logon or failed logon is supported. When AAA authentication fails during system login, the configured message banner will be displayed no matter what the reason of the failed authentication is.

Configuring the registration banner

Run the following command in global configuration mode.

Command	Purpose
aaa authentication banner *delimiter text-string delimiter*	Configures a personal logon registration banner.

Configuring the banner of failed logon

Run the following command in global configuration mode.

Command	Purpose
aaa authentication fail-message *delimiter text-string delimiter*	Configures a personal banner about failed logon.

Explanation

To create a banner, you have to configure a delimiter and then the text character string. The delimiter is to notify the system that its following text character string will be displayed as the banner. The delimiter appears repeatedly at the end of the text character string, indicating that the banner is ended.

1.4.5 Modifying the Notification Character String for Username Input

To modify the default text of the username input prompt, run **aaa authentication username-prompt**. You can run **no aaa authentication username-prompt** to resume the password input prompt.

username：

The **aaa authentication username-prompt** command does not change any prompt information provided by the remote TACACS+ server or the RADIUS server. Run the following command in global configuration mode:

Command	Purpose
aaa authentication username-prompt *text-string*	Modifies the default text of the username input prompt.

1.4.6 Changing the Character String While Prompted to Input Password

Use the “aaa authentication password-prompt” command can change the displayed default text while the user is prompted to input password. This command can not only change the password prompt of “enable” password, but also change the password prompt while making remote logging in at mean time. The “no” form of this command returns the password prompt to the default value shown in the following format:

Password：

The “aaa authentication password-prompt” command does not change any prompt information provided by remote TACACS+ or RADIUS server. While configuring, use the following command in global configuration mode:

Command	Purpose
aaa authentication *password-prompt text-string*	Change the default text displayed while prompting the user to input the password.

1.4.7 Establishing the Local Authentication Database

You may create local authentication system based on username, which is applicable to the following situation:

- Provide the username or encrypted password authentication system like TACACS+ for networks which cannot support TACACS+.

- Provide a flexible login environment; for example, accessing list verification, autocommand at login and etc.

In order to establish local username authentication, you can use the following command to configure in global configuration mode:

Command	Purpose
username *name* password { *password* | [encryption-type] *encrypted-password* }	Create username and the corresponding password.

1.4.8 Creating the Authentication Database with the Local Privilege

To create the **enable** password database with the local privilege level, run **enable password** {[***encryption-type***] ***encrypted-password*}** **[level *level*]** in global configuration mode. To cancel the **enable** password database, run **no enable password [level level]**.

enable password { [***encryption-type***] *encrypted-password*} [**level** *level*]

no enable password [level level]

1.5 Examples of AAA Authentication Configuration

1.5.1 Example of RADIUS Authentication

This section provides a configuration example of using RADIUS to authenticate, illustrating how to configure router to authenticate and authorize using RADIUS:

```
aaa authentication login radius-login radius local
aaa authentication ppp radius-ppp radius
aaa authorization network radius-network radius
line tty/vty
login authentication radius-login
interface serial 1/0
```

In this example, the meaning of each command line is:

- The “aaa authentication login radius-login radius local” command configures the router to use RADIUS as the authentication method while authenticating logging users. If RADIUS returns “ERROR”, then use the local database to authenticate the users.
- The “aaa authentication ppp radius-ppp radius” command configures the router to use ppp authentication method like chap or pap if the user has not already logged in. But if exec has already authenticated the user, there is no need to execute that again.
- The “aaa authorization network radius-network radius” command requests RADIUS for the authorization of NETWORK service, such as address allocation and other access control items.
- The “login authentication radius-login” command enables the “radius-login” method list for line 3.

1.5.2 Examples of TACACS+ Authentication

1. Example1

The following example shows how to configure TACACS+ as the security protocol to be used for PPP authentication:

```
aaa authentication ppp test tacacs+ local
interface serial1\0
ppp authentication chap pap test
tacacs server 1.2.3.4
tacacs key testkey
```

In this authentication configuration of TACACS+, the meaning of each command line is:

- The “aaa authentication ppp test tacacs+ local” command defines the “test” method list, which is applied to the serial interfaces running ppp. The tacacs+ key word means the authentication will be done through TACACS+.

If TACACS+ returns a certain kind of error while authenticating, the "local" key word indicates that it will be using the local database on the network access server to try authentication.

- The "interface" command selects the interface.
- The "ppp authentication" command applies the method list on this interface.
- The "tacacs server" command specifies the IP address of TACACS+ server as 1.2.3.4.
- The "tacacs key" command defines the shared secret "testkey"

2. Example2

The following example shows how to configure AAA authentication for PPP:

```
aaa authentication ppp default if-needed tacacs+ local
```

In this example, the "default" key word indicates the default ppp authentication method list. Key word "if-needed" means: if the user has passed the authentication at login, this authentication can be omitted. If authentication is still required, the "tacacs+" key word means to authenticate with TACACS+ server. If TACACS+ returns a certain kind of error while authenticating, the "local" key word means to use the local database on the router to authenticate.

3. Example3

The following example shows how to create the same authentication process for PAP. The thing different from the above is that this example uses the test-list method list rather than "default" method list:

```
aaa authentication pap test-list if-needed tacacs+ local
interface serial1/0
ppp authentication pap test-list
```

In this example, because no method list is applied to any interface, the administrator should use the "interface" command to select interface, so as to apply this authentication method to this interface. Then, the administrator should use the "ppp authentication" command to apply the method list to a specific interface.

1.6 AAA Authorization Configuration Task List

- Configuring EXEC authorization through AAA

1.7 AAA Authorization Configuration Task

To configure AAA authorization, perform the following steps:

1. Run **aaa authorization** to define the authorization method list. The authorization service is not provided by default.
2. If necessary, apply the authorization method list to a specific interface or line.

Note: Currently only the EXEC authorization is supported.

1.7.1 Configuring EXEC Authorization Through AAA

To enable AAA authorization, run **aaa authorization**. The **aaa authorization exec** command can create one or several authorization method lists and enable the EXEC authorization to decide whether the EXEC hull program is run by the users or not, or decide whether the users are authorized with the priviledge when entering the EXEC hull program. After the authorization method lists are configured, you can apply these lists by running **login authorization**.

Command	Purpose
aaa authorization exec {**default** | *list-name*}*method1* [*method2...*]	Creates the global authorization list.
line [**aux** | **console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enter the configuration mode of a line.
login authorization {**default** | *list-name*}	Applies the authorization list to one or several lines.

The **list-name** keyword is used to name any character string of the established authorization list. The **method** keyword is used to designate the real method for the authorization process. Only when the previously-used method returns the authorization error can other authorization methods be used. If the authorization fails because of the previous method, other authorization methods will not be used. If you require the EXEC shell to be entered even when all authorization methods return the authorization errors, designate **none** as the last authorization method in the command line. For example, if you want the authorization to be successful even if the TACACS+ service returns the errors, you can run the following command:

```
aaa authorization exec default tacacs+ none
```

The default parameter can create a default authorization list, which will be automatically applied to all interfaces. For example, you can run the following command to designate RADIUS as the default authorization method of EXEC:

```
aaa authorization exec default radius
```

Note:

If the authorization method list cannot be found during authorization, the authorization will be directly passed without the authorization service conducted.

The following table lists currently-supported EXEC authorization methods:

Keyword	Description
group *WORD*	Uses the named server group to conduct authorization.
group radius	Performs the RADIUS authorization.
group tacacs+	Uses the TACACS+ authorization.
local	Uses the local database to perform authorization.
if-authenticated	Automatically authorizes the authenticated user with all required functions.
none	Passes the authorization unconditionally.

1.8 AAA Authorization Examples

1. Example of Local EXEC Athorization

The following example shows how to perform the local authentication and local authorization by configuring the router:

```
aaa authentication login default local
aaa authorization exec default local
!
username exec1 password 0 admin priviledge 15
username exec2 password 0 admin priviledge 10
username exec3 nopassword
username exec4 password 0 admin user-maxlinks 10
username exec5 password 0 admin autocommand telnet 172.16.20.1
!
```

The following shows the meaning of each command line:

- The **aaa authentication login default local** command is used to define the default login-authentication method list, which will be automatically applied to all login authentication services.
- The command is used to define the default EXEC authorization method list, which will be automatically applied to all users requiring entering the EXEC shell.
- The user is **exec1**, the login password is **admin** and the EXEC privilege level is **15** (the highest level). In this case, if user **exec1** logs on to EXEC shell, he will own privilege level **15** and then can browse and execute all commands.
- The user is **exec2**, the login password is **admin** and the EXEC privilege level is **10** (the highest level). In this case, if user **exec2** logs on to EXEC shell, he will own privilege level **10** and then can browse and execute all commands within privilege level **10**.
- User **exec3** does not need the login password.
- The login password for user **exec4** is **admin** and user **exec4** can limit up to 10 sessions.
- The user is **exec5** and the login password is **admin**. After user **exec5** logs on to the EXEC shell, immediately run **telnet 172.16.20.1**.

1.9 AAA Accounting Configuration Task List

- Configuring the connection accounting through AAA
- Configuring the network accounting through AAA

1.10 AAA Accounting Configuration Tasks

To configure AAA accounting, perform the following steps:

1. Run **aaa accounting** to define the accounting method list. The accounting service is not provided by default.

2. If necessary, apply the accounting method list to a specific interface or line.

1.10.1 Configuring the Connection Accounting Through AAA

You can enable AAA accounting by running **aaa accounting**. The **aaa accountinh connection** command can be used to establish one or several accounting method lists. If the connection accounting is enabled, the information about all router-generated outgoing connections will be provided. These outgoing connections include Telnet, PAD, H323, rlogin, etc. Currently, only H323 is supported.

Command	Purpose
aaa accounting connection {start-stop | stop-only | none} {**default** | *list-name*} **group *groupname***	Establishes the global accounting list.

The **list-name** keyword is used to name any character string of the established accounting method list. The **method** keyword is used to designate the real method for the accounting process.

The following table lists currently-supported connection accounting methods:

Keyword	Description
group *WORD*	Uses the named server group to conduct accounting.
group radius	Performs the RADIUS accounting.
group tacacs+	Performs the TACACS+ accounting.
none	Disables the accounting service.
stop-only	Indicates that the designated accounting method sends a notification to stop recording the accounting only when the requested user process is over.
start-stop	Indicates that the designated accouniting method sends a notification to start recording the accounting when the requested event starts and sends a notification to stop recording the accounting when the event is over.

1.10.2 Configuring the Network Accounting Through AAA

You can enable AAA accounting by running **aaa accounting**. The **aaa accounting network** command can be used to establish one or multiple accounting method lists. The network accounting is enabled to provide information to all PPP/SLIP sessions, these information including packets, bytes and time accounting. You can run the following command in global configuration mode to start the configuration:

Command	Purpose
aaa accounting network {start-stop | stop-only | none} {**default** | *list-name*}	Establishes the global accounting list.

group ***groupname***	

The **list-name** keyword is used to name any character string of the established accounting method list. The **method** keyword is used to designate the real method for the accounting process.

The following table lists currently-supported network accounting methods:

Keyword	Description
group *WORD*	Uses the named server group to conduct accounting.
group radius	Performs the RADIUS accounting.
group tacacs+	Performs the TACACS+ accounting.
none	Disables the accounting service.
stop-only	Indicates that the designated accounting method sends a notification to stop recording the accounting only when the requested user process is over.
start-stop	Indicates that the designated accounting method sends a notification to start recording the accounting when the requested event starts and sends a notification to stop recording the accounting when the event is over.

1.10.3 Configuring Accounting Update Through AAA

To activate the AAA accounting update function for AAA to send the temporary accounting record to all users in the system, run the following command:

Command	Purpose
aaa accounting update [**newinfo**] [**periodic** *number*]	Activates the AAA accounting update.

If the **newinfo** keyword is used, the temporary accounting record will be sent to the accounting server when there is new accounting information to be reported. For example, after IPCP negotiates with the IP address of the remote terminal, the temporary accounting record, including the IP address of the remote terminal, will be sent to the accounting server.

When the **periodic** keyword is used, the temporary accounting record will be sent periodically. The period is defined by the **number** parameter. The temporary accounting record includes all accounting information occurred before the accounting record is sent.

The two keywords are contradictable, that is, the previously-configured parameter will replace the latter-configured one. For example, if **aaa accounting update periodic** and then **aaa accounting update newinfo** are configured, all currently-registered users will generate temporary accounting records periodically. All new users have accounting records generated according to the **newinfo** algorithm.

1.10.4 Limiting User Accounting Without Username

To prevent the AAA system from sending the accounting record to the users whose username character string is null, run the following command in global configuration mode:

aaa accounting suppress null-username

Chapter 2 Configuring RADIUS

2.1 Overview

This chapter introduces RADIUS（Remote Authentication Dial-In User Service）security system. Define its operation and introduce the network environment suitable or not suitable for using RADIUS. The section “RADIUS configuration procedure” introduces how to use authentication, authorization and accounting (AAA) command collection to configure RADIUS. The last section of this chapter “Examples of RADIUS configuration” provides two examples. For complete description about the command “RADIUS” used in this chapter, please refer to “RADIUS configuration command”.

2.1.1 A brief introduction of RADIUS

RADIUS is a distributed client/server system, which protects the network from disturbance from unauthorized access. The RADIUS client runs on the router, and sends authentication request to the central RADIUS server, the central server here includes all the user authentication and network access service information. We use AAA security mode on the router to support RADIUS, RADIUS has already been applied to various network environments which require not only high level security, but also the maintenance of remote user access.

RADIUS can be used in the network environments, which have the following security requirements:

- The network environment with many manufacturer access servers, each supports RADIUS. For example, the access server provided by many factories can use the single RADIUS security database based on the server. In the network based on IP and provided by many factories, the dial-up user makes authentication through RADIUS server.

- In the network where user should only access single service. Use RADIUS may control the user to access a single host, single application(like Telnet) or single protocol (like point-to-point protocol PPP). For example, when user logs on, RADIUS regulates and restricts this user to run PPP using IP address 10.2.3.4, and starts defined access-list.

- The network commands resource accounting. It can use RADIUS accounting which is not related to RADIUS authentication or authorization. RADIUS accounting permits to send data at the start and end of the service, in order to denote the resource volume used while carrying out the dialogue (like time, bytes, etc.).

RADIUS is not suitable for the following network security situations:

- RADIUS does not support the following protocol:

  ARA，AppleTalk Remote Access protocol

  NBFCP，NetBIOS Frame Control Protocol

- NASI，NetWare Asynchronous Services Interface

- X.25 PAD connections

- The situation from router to router. RADIUS does not provide bilateral authentication. Running RADIUS on the router, can only realize incoming authentication, for outcoming authentication, (which is, local router should pass the authentication of remote router while logging on to the remote router) it is not applicable.

- The network using various services. RADIUS normally bundle the user to a service model.

2.1.2 Operation of RADIUS protocol

When the user uses RADIUS to perform logging authentication, the following process occurs:

- Prompt the user to input username and password.

- The username and encrypted password is sent to the RADIUS server through the network.

- User receives one of the following responses from RADIUS server:

accept	The user passes the authentication.
reject	The user does not pass the authentication, prompt the user to input the username and password again, or the access will be denied.
challenge	The server send “challenge” request . This request collects additional data from the users.

Response of accept and reject returns with the additional authorization information, for EXEC or NETWORK authorization. Before using RADIUS authorization, it should first complete RADIUS authentication. The additional data included in accept and reject packet is consisted of the following contents:

- The services the user can access, include Telnet, rlogin, PPP, SLIP or EXEC.

- Connection parameters, includes the IP addresses of the host or client, the access-list and timeout setting of the user.

2.1.3 The configuration procedure of RADIUS

In order to configure RADIUS on the router or access server, you should execute the following tasks:

1. Use global configuration command “aaa authentication” to define the method list of using RADIUS authentication method. For more information about using command “aaa authentication”, please refer to “authentication configuration”.

2. Use “line” and “interface” command to quote the defined method list, for more information, please refer to “authentication configuration”.

3. Users may choose the following configuration task as needed:

If necessary, use aaa authorization global Command to authorize user request. -{}-For more information about aaa authorization Command, see “Authorization Configuration”.

If necessary, use global command "aaa accounting" to record the service process to the users. For more information regarding the using of command "aaa accounting", please refer to "accounting configuration".

2.2 RADIUS Configuration Procedure

To configure RADIUS on the router or the access server, perform the following tasks according to your real requiremtns:

- If necessary, run **aaa authorization** in global configuration mode to authorize the user’s service request. For more information about using command “aaa authentication”, please refer to “authentication configuration”.

- If necessary, run **aaa accounting** in global configuration mode to record the whole service procedure. For more information about running **aaa accounting**, see *Record Configuration*.

2.3 RADIUS Configuration Tast List

- Configure the communication between the router and RADIUS server
- Use RADIUS attributes specially used by the manufacturer to configure the router
- Configure RADIUS authentication
- Configure RADIUS authorization
- Configure RADIUS accounting

2.4 RADIUS Configuration Tast

2.4.1 Configure the Communication Between the Router and Radius Server

RADIUS server normally runs multi-user system of RADIUS server software provided by Livingston, Merit, Microsoft or other software providers, RADIUS server and router use shared key to encrypt passwords and exchange responses. Use command “radius server” to define RADIUS server, use command “radius key” to designate shared secret. While configuring, use the following commands under global configuration status:

Command	Purpose
radius server *ip-address* [**auth-port** *port-number*][**acct-port** *portnumber*]	Designate IP address of remote RADIUS server, designate the service port number of authentication and accounting.

radius key *string*	Designate the shared secret used between the router and RADIUS server.

In addition, in order to define the communication between the router and RADIUS server, please use the following optional radius global configuration command:

Command	Purpose
radius retransmit *retries*	Designate the times for the router to retransmit every RADIUS request to the server before giving up retrying.
radius timeout *seconds*	The waiting seconds before re-transmittance of RADIUS request
radius deadtime *minutes*	The duration of the server to be tagged “dead” when RADIUS server does not respond to the authentication request.

2.4.2 Using RADIUS Attributes Specially Used by the Manufacturer to Configure the Router

Internet Engineering Task Force (IETF) drafts standards passes to use vendor-specific attributes (attribute26), which provide a method for the network access server and RADIUS server to exchange special extension attributes based on the manufacturer. VSA allows manufacturer to support the extended attributes belonging to them yet not suitable for general usages. For more information about related manufacturer ID and VSA, please refer to RFC 2138: RADIUS. To configure the mode which enables the network server to identify and use VSA, please use the following commands under global configuration status:

Command	Purpose
radius vsa send [**authentication**]	Enable the network access server to identify and use VSA defined in RADIUS IETF attribute 26.

2.4.3 Configuring RADIUS Authentication

After configured RADIUS server and defined RADIUS authentication key, you should define a method list for RADIUS authentication. As RADIUS authentication is carried out through AAA, so you need to input command “aaa authentication”, and designate RADIUS as the authentication method. For more related information, please refer to “authentication configuration”.

2.4.4 Configure RADIUS authorization

Using AAA authorization we could set parameters and restrict the network access of the user. Using the authorization of RADIUS provides a method for remote access control, includes once authorization or the authorization of every service. Because RADIUS authorization is carried out through AAA, so you need to use command “ aaa authorization” to designate RADIUS as the authorization method. For more related information, please refer to “authorization configuration”.

2.4.5 Configuring RADIUS Accounting

AAA accounting feature enables us to trace the services the user accessed and their occupation of the network resource. As the RADIUS accounting feature is provided through AAA, you need to use command “aaa accounting”, designate RADIUS as the accounting method. For more related information, please refer to “Accounting configuration”.

2.5 Examples of RADIUS Configuration

The examples regarding RADIUS configuration in this section contain the following contents:

2.5.1 Examples of RADIUS Authentication and Authorization

The following example illustrates the way to configure the router so that RADIUS may be used for authentication:

```
aaa authentication login use-radius radius local
aaa authentication ppp use-radius if-needed radius
aaa authorization exec radius
aaa authorization network radius
```

In this example, the meaning of each command line is:

Command “aaa authentication login use-radius radius local” configures the router to use RADIUS to make authentication during the login process. If RADIUS server has no response, use local database to authenticate then. In this example, use-radius is the name of the method list, it designates to perform RADIUS authentication first, followed by local authentication.

Command “aaa authentication ppp use-radius if-needed radius” enables the CHAP or PAP authentication process for PPP runs through RADIUS server before the user is authenticated. If the user has already been authenticated before the provision of EXEC service, no more RADIUS authentication needs to be carried out then. In this example, use-radius is the name of method list, it defines an authentication method to carry out only if needed.

aaa authorization exec radiusCommand configuration has authorize EXEC request.

aaa authorization network radiusCommand configuration has authorize NETWORK (PPP、SLIP) service.

2.5.2 Example of Applying RADIUS in AAA

This is an example of using AAA command collection to define general configuration:

```
radius server 1.2.3.4
radius key myRaDiUSpassWoRd
username root password AlongPassword
aaa authentication ppp dialins radius local
aaa authentication login admins local
line 1 16
```

```
login authentication admins
interface async0/0
encap ppp
ppp authentication pap dialins
```

In this example, the meaning of each command line is:

Command “radius server” defines the IP address of RADIUS server;

Command “ radius key defines the shared secret between the network access server and RADIUS server host;

Command “aaa authentication ppp dialins radius local” defines authentication method list “dialins”, it designates that it should first make authentication through RADIUS, and followed by local authentication (if RADIUS server does not respond);

Command “ppp authentication pap dialins” applies authentication method list “dialins” to the designated lines;

Command “aaa authentication login admins local” defines another method list “admins” for login authentication;

Command “login authentication admins” designates to use method list “admins” in login authentication.

Chapter 3 Configuring TACACS+

3.1 Overview

TACACS+ is a kind of control protocol for safe access, which provides centralized authentication for users to obtain rights to access router or network access server. Due to the encrypted format of information exchange between network access server and TACACS+ serving programs, it may ensure the safety of communication.

Before using the characteristics of TACACS+ configured on network access server, it is necessary to be able to access and configure TACACS+ server. TACACS+ provides the ability of independent authentication, authorization and accounting of modulization.

Authentication not only supports several authentication methods, such as ASCII, PAP and CHAP, etc., but also provides and deals with the capability of any conversations with users, for example, asking some enquires to users after users entered usernames and passwords, such as home addresses, service types and ID No., etc… In addition, TACACS+ authentication service supports sending information to users' screen, for instance, inform users that they should change passwords right away since the aging policies of the company.

Authentication meticulously controls the serving purview of users during the period of offering service, including configuring automatic commands, access controlling, durable time for conversations, etc… It may also compulsively restrict the commands that users may execute.

Accounting gathers and delivers the information used in creating charging bills, auditing or conducting statistics of using status of network resources. Network administrator may use the accounting ability to safely audit the activities of the traced users or provide information for bills of account of users. Function of accounting records users identification, start as well as ending time, executed commands, amounts of bags and bytes, etc…

3.1.1 Protocol Operation of TACACS+

1. Authentication of ASCII format

When users login into the network access server using TACACS+ and are required to undertake simple authentications in ASCII format, the following processes may come out in typical situations:

After connection constructed, network access server contacts with TACACS+ server program to obtain username-prompts and then displays to users. When users input usernames, network access server contacts with TACACS+ serving program again to obtain password-prompts and then displays the them to users while users input passwords, then the passwords are sent to TACACS+ server programs.

Notes:

TACACS+ allows discretionary conversations between server programs and users until enough information is gathered to conduct authentication to users. This normally achieves through prompting the combination of usernames and passwords, also including other items, such as ID No., etc., all are undertaken under the control of TACACS+ server programs.

Network access server finally receives one of the following responses from TACACS+ server:

accept	Users have passed authentication and service may start. If network access server is configured to require service authorization, it is time to begin authorization.
reject	Users have not passed authentication. Users may be rejected to conduct further access or be prompted to re-login, depending on disposal manners of TACACS+ server.
error	Error occurs during authentication, which may be due to the server or the network connection between the server and network access server. If there is a response of ERROR, generally, network access server may attempts to authenticate users in another way.
continue	Prompt users to input additional authentication information.

2. Authentication of PAP and CHAP formats

Login of PAP is similar with that of ASCII except for that usernames and passwords reached network access server are in the PAP messages rather than being input by users, thus no news to prompt users to input relevant information. Login of CHAP is similar with it on main contents. After authentication, if network access server requires users to conduct authentication, users need to enter the stage of authorization but before dealing with authorization of TACACS+, it is necessary to first successfully complete authentication of TACACS+.

If authorization of TACACS+ is required, contact with TACACS+ server program again and back to authorization responses of ACCEPT or REJECT. If the response of ACCEPT is back, it may include AV (attribute-value) data, EXEC or NETWORK conversations to regulate the users and services to ensure users' possibility to access.

3.2 TACACS+ Configuration Flow

To configure the router to support the TACACS+ mode, you must perform the following tasks:

Firstly, run **tacacs server** to designate one or multiple IP addresses of the TACACS+ server. Secondly, run **tacacs key** to designate an encryption key for the information exchange between the access server and the TACACS+ server. You need to note that even the same encryption key must be configured in the TACACS+ server program.

Thirdly, run **aaa authentication** in global configuration mode to define the TACACS+ authentication method list. For more information about running **aaa authentication**, see *Authentication Configuration*.

Fourthly, run **line** and **interface** to apply the defined method list to the interface or line. For more information about the two commands, see *Authentication Configuration*.

3.3 TACACS+ Configuration Tast List

In order to configure router as a way to support TACACS+, it is necessary to execute following tasks:

Use TACACS+ server command and appoint one or more IP addresses of TACACS+ server. Use TACACS+ key command to appoint encrypted secret for information exchanges between network access server and TACACS+ server. Same secret key also must be configured in the TACACS+ server programs.

Use the overall configuration command of AAA authentication to define method lists of TACACS+ authentication. For more relevant information on AAA authentication authentication command, please refer to 'authentication configuration'.

Use Line and Interface commands to apply defined method lists for ports and circuitries. For more relevant information, please refer to 'authentication configuration'.

- Designation of TACACS+ Server
- Configuration of Encrypted Secret Key of TACACS+
- Designation of Using TACACS+ for Authentication
- Designation of Using TACACS+ for Authorization
- Designation of Using TACACS+ for Accounting

3.4 TACACS+ Configuration Tast

3.4.1 Designating TACACS+ Server

TACACS+ server command may let you be able to designate the IP address of TACACS+ server. Since TACACS+ software searches for the host according to sequence of configuration, this characteristic is helpful to set up different server priorities. In order to designate TACACS+ host, apply the following commands in the mode of overall configuration:

command	Purpose
tacacs server *ip-address* [**single-connection**| **multi-connection**] [**port** *integer*] [**timeout** *integer*] [**key** *string*]	Designate IP address and corresponding properties of TACACS+ server.

It is still possible to configure following options when using TACACS+ commands:

- Use the keyword single-connection to specify a single connection, which allows the server program to handle more TACACS+ operations in a more effective way. The multi-connection keyword refers to multiple TCP connections.

- Use port parameter to designate the TCP port No. applied in TACACS+ server program. Default port No. is 49.
- Use timeout parameter to designate the upper limit of time (in seconds) for router to wait for responses from server.
- Use key parameter to designate the secret key to encrypt and decode messages.

Notes:

Use the timeout value appointed by TACACS+ server may cover the overall timeout value configured by TACACS+ timeout command; use the encrypted secret key appointed by TACACS+ server may cover the default secret key configured by the overall configuration command TACACS+ key. Therefore, it is possible to enhance the safety of the network from the exclusive TACACS+ connection configured by applying this command.

3.4.2 Configuring Encrypted Secret Key of TACACS+

In order to configure the encrypted secret key for TACACS+ messages, it is necessary to use the following commands in the mode of overall configuration:

command	Purpose
tacacs key *keystring*	Configure the encrypted secret key matched with that used by TACACS+ server.

Notes:

In order to successfully encrypt, it is necessary to configure the same secret key to TACACS+ server.

3.4.3 Using TACACS+ for Authentication

After identifying TACACS+ server and defining the encrypted secret key related to it, it is necessary to define method lists for TACACS+ authentication. Since TACACS+ authentication is undertaken through AAA, it is necessary to set up AAA authentication command to appoint TACACS+ as its authentication method. For more relevant information, please refer to ‘authentication configuration’.

3.4.4 Using TACACS+ for Authorization

AAA authorization may set up parameters to limit the network access purview for users. TACACS+ authorization may be used in much service, such as commands, network connections and EXEC conversations, etc…Since TACACS+ authorization is undertaken through AAA, it is necessary to set up AAA authorization command to appoint TACACS+ as its authorization method. For more relevant information, please refer to ‘authorization configuration’.

3.4.5 Using TACACS+ for Accounting

AAA accounting may trace the service that users are using and the amount of network resources the service consumes. Since TACACS+ accounting is provided through AAA, it is necessary to set up AAA authentication command to appoint TACACS+ as its accounting method. For more relevant information, please refer to 'accounting configuration'.

3.5 Samples of TACACS+ Configuration

This section consists of following:

- Samples of TACACS+ Authentication
- Samples of TACACS+ Authorization
- Samples of TACACS+ Accounting

3.5.1 Samples of TACACS+ Authentication

1. Samples 1

The following samples of PPP configuration are completed by TACACS+:

```
aaa authentication ppp test tacacs+ local
tacacs server 1.2.3.4
tacacs key testkey
interface serial 1/1
ppp authentication chap pap test
```

In this sample:

AAA authentication command defines the test of method lists of authentication used in the serial ports for circulating PPP. Keyword of TACACS+ means that authentication is conducted through TACACS+ and if some kind of ERROR is back during the TACACS+ authentication, keyword of local instructs to use local database in the network access server to authenticate.

TACACS+ server command identifies the IP address of TACACS+ server as 1.2.3.4. TACACS+ key command defines the shared encrypted secret as testkey.

Interface command selects the port while PPP authentication command applies the method list test in the port.

2. Samples 1

The following sample configures TACACS+ as safe protocol for PPP authentication without using method list test any more but method list default:

```
aaa authentication ppp default if-needed tacacs+ local
tacacs-server host 1.2.3.4
```

```
tacacs-server key goaway
interface serial 1/1
ppp authentication default
```

In this sample:

AAA authentication command defines to use method list default of authentication in serial ports for circulating PPP. Keyword if-needed means that if users have passed the authentication in the process of login, PPP authentication is needless, but if authentication is needed, then keyword TACACS+ means that authentication is undertaken through TACACS+. If some kind of ERROR is back during the period of TACACS+ authentication, then keyword local instructs to use local database in the network access server for authentication.

TACACS+ server command identifies the IP address of TACACS+ server as 1.2.3.4. TACACS+ key command defines the shared encrypted secret as testkey.

Interface command selects the port while PPP authentication command applies method list test in the port.

3.5.2 Samples of TACACS+ Authorization

The following example takes TACACS+ as one method in the default authentication method list, and shows how to perform network service authorization through TACACS+.

```
aaa authentication ppp default if-needed tacacs+ local
aaa authorization network default tacacs+
tacacs server 10.1.2.3
tacacs key goaway
interface serial 1/1
ppp authentication default
ppp authorization default
```

In this sample:

AAA authentication command defines to use method list default of authentication in serial ports for circulating PPP. Keyword if-needed means that if users have passed the authentication in the process of login, PPP authentication is needless, but if authentication is needed, then keyword TACACS+ means that authentication is undertaken through TACACS+. If some kind of ERROR is back during the period of TACACS+ authentication, then keyword local instructs to use local database in the network access server for authentication.

AAA authorization command configures to conduct network authorization through TACACS+.

TACACS+ server command identifies the IP address of TACACS+ server as 10.2.3.4. TACACS+ key command defines the shared encrypted secret as goaway.

Interface command selects ports while both PPP authentication and PPP authorization commands apply default authentication or method lists of authorization to the port.

3.5.3 Samples of TACACS+ Accounting

The following example shows how to configure the PPP authentication method list through TACACS+, and how to conduct recording through TACACS+.

```
aaa authentication ppp default if-needed tacacs+ local
aaa accounting network default stop-only tacacs+
tacacs server 10.1.2.3
tacacs key goaway
interface serial 1/1
ppp authentication default
ppp accounting default
```

In this sample:

AAA authentication command defines method list default of authentication for using of PPP protocol. Keyword if-needed means that if users have passed the authentication in the process of login, PPP authentication is needless any more, but if authentication is needed, then keyword TACACS+ means that authentication is undertaken through TACACS+. If some kind of ERROR is back during the period of TACACS+ authentication, then keyword local instructs to use local database in the network access server for authentication.

AAA accounting command configures to conduct accounting of network service by TACACS+. In this sample, only record corresponding information when the service is finished, which will be sent to TACACS+ server when network connection finishes.

TACACS+ server command identifies the IP address of TACACS+ server as 10.2.3. TACACS+ key command defines the shared encrypted secret as goaway.

Interface command selects ports while PPP authentication command applies default method list of authentication in the port and PPP accounting command applies default method list of accounting in the port.

Chapter 4 Configuring IPSec

4.1 Overview

4.1.1 About Configuration of IPSec

This chapter expatiates how to configure IPSec. IPSec is the public standard frame developed by IETF. IPSec provides safety for transferring sensitive data on unsafe networks (e.g.Internet). IPSec functions in the network level, safeguarding and authenticating the IP bags transferring among IPSec facilities (e.g. the company's router).

IPSec provides the following service for network safety, which is optional. Generally, local safety strategies will regulate to use one or more of following services:

- Privacy of data-----sender of IPSec encrypts the message before transferring it by network.
- Integrality of data-----Receiver of IPSec authenticates the message sent by senders to ensure that the data has not been changed during transfer.
- Authentication of source of data------receiver of IPSec authenticates the source address of IPSec message. This service is based on the service of integrality of data.
- Antireplay-----receiver of IPSec may test and reject rebroadcast messages.

With IPSec, there is no need to worry about the data to be supervised, changed and forged during transfer.

To obtain the complete expatriation of IPSec commands used in this chapter, please refer to 'IPSec configuration command'.

The safety scheme provided by IPSec is very strong and healthy as well as based on standards. As a complement to privacy of data, IPSec also provides the service of data demonstration and Antireplay.

4.1.2 Supported Standards

Router IPSec of our company realized the following standards:

- IPSec----IP Security protocol. IPSec is an open standard framework which provides data confidentiality, data integrity, and data authentication. IPSec provides these security services at IP level; it uses IKE to negotiate protocol and algorithm, and generates encryption and authentication key for IPSec. IPSec can provide protection for a pair of host computers, a pair of security gateways or one or several data streams.

- Internet key exchange (IKE)-----A mixed protocol to realize Oakley and SKEME key exchange in the ISAKMP framework. Although IKE can be jointly used with other protocols, it is initially used with IPSec protocol. IKE provides authentication of IPSec corresponding peers, negotiates IPSec security association, and generates IPSec keys. For detailed information about IKE, please refer to “Configuring internet key exchange security protocol”.

The IPSec technique consists of:

- DES----Data Encryption Standard（DES）is used to encrypt packet data, it is a kind of symmetrical encryption algorithm, our router uses DES-CBC with an IV of 56 bit. CBC requires an Initial Value to make encryption.

- 3DES----3DES is used to encrypt packet data, it is a kind of symmetrical encryption algorithm, our router uses DES-CBC with an IV of 168 bit. 3DES is safer than DES.

- MD5(HMAC variable)----MD5 is a kind of hash algorithm. HMAC is an encrypted hash variable used to make authentication towards data.

- SHA(HMAC variable)-----SHA is a kind of hash algorithm. HMAC is an encrypted hash variable used to make authentication towards data.

The IPSec of our router software also supports the following standards:

- AH----A kind of security protocol used to provide data integrity, data original identity authentication, and several optional anti-replay services. AH can be used to protect a superstratum protocol (transport mode) or a complete IP data packet (tunnel mode). AH can be used independently, or be used together with ESP. The definition is detailed in RFC2402.

- ESP----A kind of security protocol used to provide security services include confidentiality, data source authentication, anti-replay and data integrity and etc. ESP can be used to protect a superstratum protocol (transport mode) or a complete IP data packet (tunnel mode). The definition is detailed in RFC2406.

4.1.3 Limitation

Currently, IPSec can only be use for IP data packet uni-cast. As IPSec workgroup is not engaged in group key distribution, IPSec currently does not support multicast or broadcasting IP data packets. If Network address translation (NAT) is used, static NAT should be configured to enable normal operation. All in all, NAT should be executed before the router carries out IPSec encapsulation；in other words, IPsec should use a global address.

4.1.4 Summarization of IPSec Working Process

IPSec can provide a security tunnel between two routers. And define which packets should be transferred through these security tunnels. Also, defines the parameters to protect these sensitive packets through specifying the parameters of these tunnels. Then, when IPSec receives such message, it creates corresponding security tunnel, to transfer the data packets to the corresponding peer through the tunnel.

More exactly, the tunnel is the set of security associations established on the two peers of IPSec. These security tunnels define which protocols and algorithms will be applied in the sensitive message, meanwhile defines the key to be used on the two peers of IPSec. The security association is unilateral, every security protocol (AH or ESP) is established respectively.

In IPsec, the specified communication will be protected through the configuration of the access list, putting the access list in the encryption map, and applying it to the interface. Therefore, the communication can be filtered based on source and destination address, any fourth level protocol or port（when applying to access list of IPSec, it only defines which communications will receive protection from IPSec, but does not define which communications will be passed or prohibited from a certain port. A set of encryption map may include several encryption maps, each corresponding with one different access list.

When the message matches the first permit rules of certain access list and the corresponding type of encryption map is ipsec-isakmp, it will be processed by IPSec. If there is no existence of security association to protect messages, IPSec uses IKE to establish security association to negotiate with the corresponding port.

When the corresponding type of encryption map is ipsec-manual, it will be processed by IPSec. If there is no existence of security association to protect messages, messages will be discarded. Under this situation, the security association does not need IKE through manual configuration. If there is no existence of security association, then, not all required parameters are configured.

Once the set of security associations is established, it is applied to process messages and the succedent applicable messages. The applicable messages indicate those messages that match the same access list conditions.

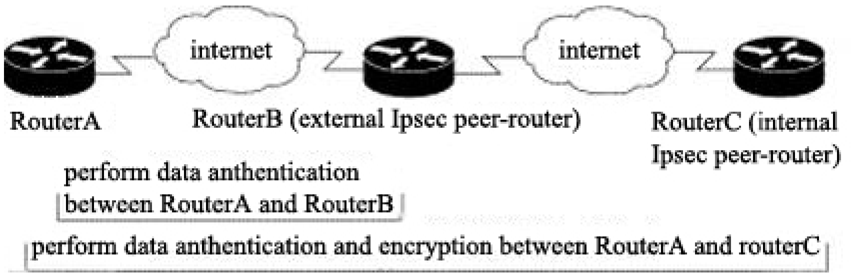
If IKE is used to establish security association, then these security associations will have a life cycle, they will periodically have overtime, and require re-negotiation (this provides an additional level of security).

The two peers of IPSec may have several IPSec tunnels to encrypt different datastreams, each corresponds to one independent set of security associations. For example, some data streams may only require authentication, while others require encryption and authentication. The access list set by the IPSec encryption map specifies which communications require IPSec protection. Entering the message and processing according to the encryption map-----if one unencrypted message matches a permit in the certain access list set by an IPSec encryption map, this message will be discarded, because it is not transferred as a message protected by IPSec. Encryption map also includes a transform set. A transform set is a combination of security protocol, algorithm and other configuration of communication protected by the IPSec. During the process of IPSec security association negotiation, identically uses a certain transform set to protect a data message.

1. IPSec nesting

You can practice IPSec communication nesting on a series of IPSecrouter. For example, in order to enable the communication to penetrate several firewalls (All these firewalls have corresponding policies, the communication, without the ability to make authentication itself, will not be passed), the router should set IPSec tunnels according to the order and each firewall.

In the following examples listed in the following figure, routerA uses IPSec to make encapsulation of the communications towards routerC (routerC is the corresponding port of IPSec). But, before routerA can transfer communication, it should use IPSec to make pre-encapsulation towards communication, with the aim to transfer the message to routerB.


internet
internet
RouterA
RouterB (external Ipsec peer-router)
RouterC (internal Ipsec peer-router)
perform data anthentication between RouterA and RouterB
perform data anthentication and encryption between RouterA and routerC

Figuer 4-1 An example of IPSec router nesting

There may be one same protection between two peers of external IPSec (such as data authentication), also there may be different protections between two peers of internal IPSec. (for example: data authentication and encryption)

4.2 IPSec Configuration Tast List

After completion of IKE configuration, IPSec configuration can be processed. If the user wants to configure IPSec, he/she should complete the following operations on each IPSec router participates in communication.

- Guarantee the compatibility between access list and IPSec
- Establishing encryption access list
- The definition of transform set
- Establishing encryption map
- The application of encryption map set to the interface

4.3 IPSec Configuration Tast

4.3.1 Guaranteeing the Compatibility Between Access List and IPSec

IKE uses UDP port 500. IPSec ESP and AH protocol uses IP protocol No. 50, 51. Guarantee that when the user is configuring-access list, the communications on the protocol 50, 51 and UDP port 500 are not prohibited on the interface used by IPSec.

4.3.2 Establishing Encryption Access List

The encryption access list is used to define which IP communication should be encrypted and protected, and which need not. (These access lists are not the same as normal access list, the normal ones only decide which can pass, and which should be prohibited on the interface).

The encryption access list specified by the IPSec encryption map has four major functions:

- Judging the outgoing messages to be protected by IPSec (permit is protected)
- When beginning negotiating IPSec security association, indication data stream will be protected by the new security association (indicated by a single permit)
- Processed into the message, with the aim to filter and discard those communications should be protected by IPSec but not covered by it.
- When processing IKE negotiation initiated by corresponding port of IPSec, decide whether to accept the application of IPSec security association of it.

If the user wants to let one certain message uses a combination protected by IPSec (For example, just make authentication), the other uses another combination protected by IPSec, (For example, authentication and encryption) then two different encryption access lists are required to be created to define two different types of communication. These two different access lists are used in different encryption maps, and these encryption maps adopt different IPSec policies.

Afterwards, when applying set of encryption map to the interface, these encryption access lists are accordingly related with these interfaces. Use the following command to create encryption access list under global configuration state:

Command	Purpose
ip access-list extended *name*	Defines which IP communications should be encryption protection.
permit *protocol source source-mask destination destination-mask*	Then uses “permit” and “deny” command to configure accessing rules.

4.3.3 Encryption Access List Techniques

Using the key word permit will enable all IP communications to meet the requirements to receive encryption protection from depicted policy of corresponding encryption map. Using the key word deny will prohibit the communication to receive encryption protection from certain encryption map (namely, it does not permit the specified policy of this encryption map to be applied on this communication). If all the encryption maps on the interface deny certain communication, this communication will not be provided with encryption protection.

After defining the corresponding encryption map, and applying the set of encryption map on the interface, the specified encryption access list will be applied on this interface. Incoming and outgoing messages will be judged by the same IPSec access list. Thus, the access list can be directly used for outgoing messages, and reversely used for incoming messages. So only one rule should be defined in the access list, when applying this rule, IPSec will match the data messages with reverse source and destination addresses.

If several rules are set for a certain encryption access list, only the first rule works.

4.3.4 Using Key Word in the Encryption Access List

When establishing encryption access list, using key word **any** may bring about problems. That's why key word **any** is not recommended to use for specifying source or destination address.

When IPSec interface is required to transfer multicast communication, the key word **any** is not recommended to be used in the permit sentence; it may result in the failure of multicast. In the sentence, the keyword **any** is not suitable to use especially because all the outgoing messages are therefore under protection (and all the messages under protection will be sent to the corresponding port specified in the corresponding encryption map), and will request all the incoming messages to be under protection.

4.3.5 Definition of Transform Set

The transform set is a combination of specified security protocol and algorithm. In the process of IPSec security association negotiation, the two peers negotiate to use one specified transform set to protect specified data stream.

The user can define several transform sets, and then configure one or more of these transform sets in a encryption map.

When IKE negotiates for security association, the two peers will search for the transform set with unity on both sides, after finding that, the security association will use this transform set to protect specified data stream.

For manual established security association, there is no negotiation process between the two sides, so both sides should specify the same transform set.

If an alteration is made to the definition of transform set, then the change will only be applied to the encryption map which configured the transform set. The alteration will not be applied to the existed security association, but it will be applied in the later negotiation of establishing new security association. If the user wants to activate all these new settings immediately, "clear crypto sa "command may be used to partly or totally delete the security association database.

Using the following commands to define and transform set under global configuration state:

Command	Purpose
crypto ipsec transform-set *transform-set-name*	To define a transform set, when executing this command, it will enter into the encrypted exchange configuration state.
transform-type *transform1 [transform2[transform3]]*	Setting transform type.
mode [**tunnel** | **transport**]	Change the mode of transform set. The mode setting is only applicable to communications with the same source and destination addresses of IPSec; yet not used for all other communications (all other communications will be processed only under tunnel mode).
exit	Exit encrypted transform configuration state.

The following list is the applicable transform combination.

Selecting transform for transform set: applicable transform combination					
Select one type from AH transform		Select one from ESP encrypted transform		Select one from ESP authentication transform	
Transform	Description	Transform	Description	Transform	Description
ah-md5-hmac	AH authentication algorithm with MD5 (HMAC variable)	esp-des	ESP encrypted algorithm using DES	esp-md5-hmac	ESP authentication algorithm with MD5 (HMAC variable)
ah-sha-hmac	AH authentication algorithm with SHA (HMAC variable)	esp-3des	ESP encrypted algorithm using 3DES.	esp-sha-hmac	ESP authentication algorithm with SHA (HMAC variable)

4.3.6 Establishing Encryption Map

1. Regarding encryption map

The encryption map established for IPSec includes the following parts:

- Which of the communications should receive IPSec protection (encryption access list)
- Data stream intervals that will receive one protection from set of security association set
- Corresponding port address used for IPSec communication
- Current port address used for IPSec communication
- Which IPSec security policies apply to these communications (select from one or more transform set)
- Security associate is established manually or through IKE
- Other parameters may be used to define IPSec security association

Encryption maps with the same encryption map name (but different mapping serial number) form a encryption map set. Then applies the encryption map set on the interface; in this way, all IP messages passing this interface will be applied to the encryption map set on the interface to make judgment. If one encryption map matches an outgoing message that should receive protection, and encryption map specified to use IKE, it will carry out security association negotiation with corresponding port according to the included parameters of this encryption map. If the encryption map uses security association set manually established, then the security association should be established when processing configuration. If the negotiation is initiated from locally, then the specified policy in the encryption map will be used to establish and send to the IPSec corresponding port. If the negotiation is initiated from the corresponding port of the IPSec, then the local router will check the policy provided by the corresponding port, and then decide whether to accept or to decline the application from corresponding port.

In order to smoothly process IPSec communications between the two peers of IPSec, the encryption map on the two peers should include mutually compatible configuration sentence.

When two peers try to establish security association, both peers should have at least one encryption map which is compatible with one encryption map of the corresponding port. The compatibility of the two encryption maps should at least meet the following conditions:

- The encryption map should include compatible encryption access list.
- The encryption map of both sides should specify the corresponding peer address.
- The encryption map should have at least one identical transform set.

2. How many encryption maps should be established?

One interface may use only one encryption map set. The encryption map set includes combination of IPSec/ipsec-isakmp， or IPSec/ipsec-manual. If the user wants to apply the same policy on many interfaces, or to let many interfaces share the same encryption map set.

If the user wants to establish many encryption maps for the specified interface, seq-num parameter which uses mapping will be used to arrange sequence; the lower the value of seq-num is, the higher the priority is. First use mapping of high priority to judge communication on the interface with this encryption map set. If one situation of the followings exists, the user should establish many encryption maps for one port:

Different data messages will be processed by different IPSec corresponding peers.

If the user wants to apply different IPSec security policies to different types of data messages; for example, he/she wants to apply authentication to the communications in one setup subnet, yet apply both authentication and encryption in the other setup subnet. Under this situation, different types of communications should be defined in two different access list, and should establish a separate encryption map for each encryption access list.

If the user established security association manually, and wanted to specify many access lists, he/she should establish different access lists (each includes one permit), and specify a separate encryption map set for each access list.

3. Establishing manual mode encryption map

The use of manual security association is the result pre-arranged by the local router and the administrator of IPSec corresponding peer. Both sides may want to use manual security association at the beginning, and then use security association based on IKE, or the remote system may not support IKE. If the security association is not established with IKE, no negotiation process of security association will exist, so, in order to smoothly process messages for IPSec, the configuration of two peers should be the same.

The router may support both manual establishment and IKE establishment of security association.

In order to establish encryption map with manual mode, use the following commands under global configuration state:

Command	Purpose

crypto map *map-name seq-num* **ipsec-manual**	Establishing or modifying encryption map, execute this command and enter into encryption map configuration state.
match address *access-list-name*	Setting IPSec access list. This access list decides which messages can be protected by IPSec, which cannot be protected under the IPSec security defined by this encryption map.
set peer *ip-address*	Setting IPSec corresponding port address. The messages protected by IPSec will be forwarded to this address.
set transform-set *transform-set-name*	Setting transform set(regarding ipsec-manual encryption map, only transform set can be defined. For ipsec-isakmp, no more than six encryption maps can be defined).
set security-association inbound ah *spi hex-key-data* **set security-association outbound ah** *spi hex-key-data*	If the specified transform set includes AH protocol, then this command should be used to set AH security parameter index (SPIs) and key for incoming and outgoing messages. This command manually specified that AH security association will be used to protect messages.
set security-association inbound esp *spi* [**cipher** *hex-key-data*][**authenticator** *hex-key-data*] **set security-association outbound esp** *spi* [**cipher** *hex-key-data*] [**authenticator** *hex-key-data*]	If the specified transform set includes ESP protocol, then this command should be used to set ESP security parameter index (SPIs) and key for incoming and outgoing messages. If it includes ESP encryption algorithm, encryption key must be provided. This command manually specified that ESP security association will be used to protect communication.
exit	Exit encryption map configuration state, and return to global configuration state

Repeat these steps to establish other encryption maps.

4. Establishing encryption map with IKE

In order to establish security association encryption map via IKE, use the following commands under global configuration state:

Command	Purpose
crypto map *map-name seq-num* **ipsec-isakmp**	Establishing or modifying encryption map, execute this command and entering into encryption map configuration state
match address *access-list-name*	Setting IPSec access list. This access list decides which messages can be protected by IPSec, which cannot be protected under the IPSec security defined by this encryption map.
set peer *ip-address*	Setting IPSec corresponding port address. The messages protected by IPSec will be forwarded to this address.
set transform-set *transform-set-name1* [*transform-set-name2…transform-set-name6*]	Setting transform set，no more than six encryption maps can be defined（the first has the highest priority）
set security-association lifetime seconds *seconds*	(Optional) setting the life cycle of encryption map negotiated security association

set security-association lifetime kilobytes *kilobytes*	
set pfs [**group1** | **group2**]	(optional) The specified IPSec is applying for perfect forward security when using this encryption map to apply for new security association, and requires the application replied from the IPSec corresponding port to have PFS request.
exit	Exit encryption map configuration state, and return to global configuration state

Repeat these steps to establish other encryption maps.

4.3.7 The application of encryption map set to the interface

In order to let the configured encryption map work, the user should apply the encryption map to the interface. The router uses this encryption map set to judge all the messages passing through this interface, and applies specified policies to different messages under protection in the security association negotiation process.

In order to apply the encryption map set to the interface, use the following commands under interface configuration state.

Command	Purpose
crypto map *map-name*	Applying the encryption map set to the interface.

A same encryption map set may be used for many interfaces.

4.4 Examples for IPSec Configuration

The following are examples of IPSec configuration using IKE negotiated security association.

Regarding IKE, please refer to document “Configuring IKE”.

1. Defining encryption access lists

```
ip access-list extended aaa
permit ip 130.130.0.0 255.255.0.0 131.131.0.0 255.255.0.0
```

2. Defining transform set

```
crypto ipsec transform-set one
transform-type esp-des esp-sha-hmac
```

3. Setting IPSec access list and transform set with encryption map, and defining the destination of encrypted messages (the corresponding port of IPSec)

```
crypto map toShanghai 100 ipsec-isakmp
match address aaa
set transform-set one
set peer 192.2.2.1
```

4. Applying encryption map to the interface

```
interface Serial1/1
ip address 192.2.2.2
crypto map toShanghai
```

Chapter 5 Configuring IKE Security Protocol

5.1 Overview

This chapter discusses how to configure Internet Key Exchange (IKE) protocol. IKE is a key management protocol standard, used with IPSec standard. IPSec may not use IKE, but IKE enhances the function of IPSec through the provision of extra functions, flexibility and easier configuration of IPSec. IKE is a mixed protocol, this protocol realized Oakley key exchange and Skeme key exchange within the framework of internet security association and key management protocol (ISAKMP).(ISAKMP, Oakley and Skeme are the security protocols for IKE to be realized).

IKE can automatically process IPSec security association (SA), no need to pre-configure manually yet able to carry out IPSec security communication. IKE provides the following benefits:

- IKE avoids manual setting of all IPSec security parameters in the encryption map of two sides of communication.
- IKE permits to define life cycle of IPSec security association.
- IKE permits to change encryption key during the process of IPSec conversation.
- IKE permits IPSec to provide anti-replay service.
- IKE permits dynamic authentication between the peers.

5.1.1 Supported Standards

The router realizes the following standards:

- IPSec------IPSec is an open standard system which provides data encryption, data integrity and data authentication services between the peers. IPSec provides these security services at IP level, uses IKE to negotiate protocol and algorithm, and generates encryption and authentication key for IPSec. IPSec can be used to protect one or more data streams between a pair of host computers, a pair of security gateways or a security gateway and a host computer.

- IKE----This mixed protocol realizes Oakley and Skeme key exchange protocol in the ISAKMP framework, and can be used together with other protocols, and its initial aim of realization is to be used together with IPSec protocol. IKE provides the authentication on the two peers on IPSec, negotiate IPSec key and IPSec security association.

- ISAKMP------This protocol framework defines the format of payload to realize negotiation of the mechanism of key exchange protocol and security association.

- Oakley------A key exchange protocol which defines how to acquire authentication key materials.

- Skeme----A key exchange protocol which defines how to acquire authentication key materials and carry out quick key updating.

IKE realizing techniques include the following contents:

- DES------Data Encryption Standard (DES).

- 3DES----3DES is used to encrypt packet data.

- Diffie-Hellman----A key encrypted protocol, allows two groups to establish shared secrets on non-secured communication channel. Diffie-Hellman uses and establishes conversation key with IKE. The router supports 768 bit and 1024 bit Diffie-Hellman group.

- MD5 (HMAC variable) -----MD5 is a kind of hash algorithm. HMAC is an encrypted hash variable used to make authentication towards data.

- SHA (HMAC variable) ------SHA is a kind of hash algorithm. HMAC is a encrypted hash variable used to make authentication towards data.

5.2 IKE Configuration Tast List

Executes the tasks in the following sections when configuring IKE. The tasks of the first three sections is a must; others are optional, these all depend on the parameters configured.

- Guarantee the compatibility of access list with IKE

- Establish IKE policy

- The setting of pre-share key

- The clearing of IKE connection (Optional)

- IKE diagnosis (Optional)

Regarding examples of IKE configuration, please refer to the section“Examples of IKE configuration” at the end of this section.

5.3 IKE Configuration Tast

5.3.1 Guarantee the Compatibility of the Access List with IKE

IKE negotiates to use UDP on port 500. It guarantees that the communication on UDP port 500 is not prohibited on the interface used by IKE and IPSec. Under some circumstances, an additional rule is required in the access list to clearly permit packetsfrom UDP 500.

5.3.2 Establishing the IKE Policy

IKE policy is required to establish on the two peers of IPSec. IKE policy defines the security parameter combination used during the IKE negotiation.

In order to establish IKE policy, please take notice of the following questions:

- The reasons to establish policy
- Parameters defined in the policy
- How to complete a matching policy with IKE
- Setting value of policy parameters
- Establishing policy
- Extra configuration required for IKE policy

1. Reasons to establish policy

IKE negotiation should be protected, so each IKE negotiation starts from achieving public IKE policy. This policy describes which security parameters can be used to protect proceeding IKE negotiation.

After the two peers achieve a public IKE policy, the security parameter of this policy has ISAKMP security association tag established on each peer, during negotiation these security associations can be applied to all the proceeding IKE communications.

Many policies with different priorities should be established on both peers, in order to guarantee that at least one policy can match the remote policy.

2. Parameters defined in the policy

Defines 5 parameters in each IKE policy:

Parameter	Accepted value	Key word	Default value
Encrypted algorithm	56 bit DES-CBC 168 bit 3DES-CBC	des 3des	56 bit DES-CBC
Hash algorithm	SHA-1	sha	SHA-1

	MD5	md5	
Authentication method	Pre-shared key RSA signature RSA real-time encryption	pre-share rsa-sig rsa-encr	Pre-share key
Diffie-Hellman group tag	768 bit Diffie-Hellman 1024 bit Diffie-Hellman	1 2	768 bit Diffie-Hellman
The life cycle of security association	Can be defined as any period of time between 60 and 86400 seconds	-	86400 seconds (one day)

3. How to complete a matching policy with IKE

When IKE negotiation starts, IKE is searching for an IKE policy of the same on both peers. The peer that initiates negotiation sends all the policies to the remote peer, but the remote peer will try to find a matching policy. The remote peer searches for matching items via matching between the policy with the highest priority and the received policy. The remote peer checks every policy according to the order of priority (highest priority first) until finding a matching policy.

It makes matching choice when policies from the both peers include identical value of encrypted, hash, authentication and Diffied-Hellman parameter, and the life cycle defined by the remote peer policy is less or equal to the compared policy life cycle.

If no acceptable matching policy is found, IKE denies negotiation, and will not establish IPSec.

If a matching policy is found, IKE will complete the negotiation, and establish the IPSec security association.

Notes:

Extra configuration of the parameter is needed according to the different authentication methods defined by the policy.

4. Setting value of policy parameters

There are two choices for encrypted algorithm: 56 bit DES-CBC and 168-bit 3DES-CBC. 3DES-CB C is more secure.

There are two choices for hash algorithm: SHA-1 and MD5. MD5 has less abstract information and is quicker than SHA-1. One of the attacks towards MD5 is proved to be successful (but very difficult); however, the HMAC variable used by IKE can block this attack.

There are three choices for authentication method: currently, it only supports pre share key.

There are two choices for Diffie-Hellman group: 768 bit or 1024 bit Diffie-Hellman. 1024bit Diffie-Hellman is harder to attack but requires more CPU time.

The life cycle of the security association can be set as any value between 60-86400 seconds. The shorter the life cycle is, the more secure the IKE negotiation is. For longer life cycle, the latter IPSec security association can be set up quicker. For more information of this parameter and how to use it, please refer to the command description of lifetime (IKE policy) command.

5. Establishing policy

The user may establish many IKE policies, each corresponds to different parameters combination. For each policy established, a sole priority should be assigned (1-10000, 1 stands for the highest priority).

Multi-policy can be set on each peer-but at least one of these policies must contain encryption, hash, authentication and Diffie-Hellman parameter value which is identical with one policy on remote terminal.

If no policy is set, the router adopts a default policy which is always set to the lowest priority and contains the default value of every parameter.

In order to set one policy, use the following commands under the global configuration mode:

Command	Purpose
crypto isakmp policy *priority*	Establishing IKE policy (each solely tagged with priority) (this command enables you to enter into ISAKMP policy configuration state)
encryption {**des**|**3des**}	Defining encrypted algorithm
hash {**sha** | **md5**}	Defining hash algorithm
authentication { **pre-share**|**rsa-sig**|**rsa-encr**}	Defining authentication method
group {**1** | **2**}	Defining Diffie-Hellman group
lifetime *seconds*	Defining life cycle of IKE security association
exit	Exit ISAKMP policy configuration state
show crypto isakmp policy	(Optional) Showing all existed IKE policy (Using this command under management state)

If no value is assigned to the parameter, default value applies.

Notes:

When sending “show running” command, the default policy and the default value of configured policy are not shown in the configuration. Use “show crypto isakmp policy” command to check the default value of default policy and the configured policy.

6. Extra configuration required for IKE policy

Pre-share key authentication method: if pre-share key is assigned as the authentication method in the policy, pre-shared key should be equipped accordingly.

5.3.3 Setting Pre-Share Key

The pre-share key provided by the NOTICE is shared on both peers. A same key should be defined on both peers.

Use the following command under global configuration state if the user wants to reconfigure pre-share key:

Command	Purpose
crypto isakmp key *keystring peer-address*	At local peer: defining shared secret used together with the remote peer.
crypto isakmp key *keystring peer-address*	At remote peer: defining shared secret used together with the local peer.

5.3.4 Clearing the IKE Connection (Optional)

If required, the user can clear the existing IKE connection. In order to clear IKE connection, the user can use the following commands in EXEC mode:

Command	Purpose
show crypto isakmp sa	Showing existed isakmp SA.
clear crypto isakmp [**map** *map-name* | **peer** *ip-address*]	Clearing isakmp connection.

5.3.5 IKE Diagnosis (Optional)

In order to get help from IKE diagnosis, the user can use the following commands under management state:

Command	Purpose
show crypto isakmp policy	Shows parameters of IKE policy.
show crypto isakmp sa	Shows all current IKE security associations.
debug crypto isakmp	Shows debug information concerning IKE.

After completing IKE configuration, the user can configure IPSec. Please refer to “Configuring IPSEC”

5.4 Examples of IKE Configuration

1. Example1

This example established two IKE policies, policy 10 with the highest priority and policy 20 with the second highest priority, default priority is set as the lowest. Meanwhile, the policy establishes a pre-share key to be used together with the remote policy with IP address of 192.168.1.3.

```
crypto isakmp policy 10
encryption des
hash md5
```

```
authentication pre-share
group 2
lifetime 5000
crypto isakmp policy 20
authentication pre-share
lifetime 10000
crypto isakmp key 1234567890 192.168.1.3
```

2. Example2

In the above example, encryption des in policy 10 will not appear in the written configuration, because it is the default value of encryption algorithm parameter.

If the user sends “show crypto isakmp policy” command with this configuration, the router may produce output in the following form:

```
Protection suite of priority 10
    encryption algorithm:     DES   - Data Encryption Standard (56 bit keys).
    hash algorithm:           Message Digest 5
    authentication method:    Pre-Shared Key
    Diffie-Hellman group:     #1 (768 bit)
    lifetime:                 5000 seconds
Protection suite of priority 20
    encryption algorithm:     DES   - Data Encryption Standard (56 bit keys).
    hash algorithm:           Secure Hash Standard
    authentication method:    Pre-Shared Key
    Diffie-Hellman group:     #1 (768 bit)
    lifetime:                 10000 seconds
Default protection suite
    encryption algorithm:     DES   - Data Encryption Standard (56 bit keys).
    hash algorithm:           Secure Hash Standard
    authentication method:    Pre-Shared Key
    Diffie-Hellman group:     #1 (768 bit)
    lifetime:                 86400 seconds
```