

Per-User Configuration

This chapter describes a large-scale solution that uses features and commands described in other chapters of this manual, the *Dial Solutions Command Reference*, and the *Security Configuration Guide*.

Per-user configuration provides a flexible, scalable, easily maintained solution for customers with a large number of dial-in users. This solution can tie together the following dial-in features:

- Virtual template interfaces, generic interface configuration and router-specific configuration information stored in the form of a virtual template interface that can be applied (*cloned*) to a virtual access interface each time any user dials in. This is described in the “Configuring Virtual Template Interfaces” chapter of this manual.
- AAA per-user security and interface configuration information stored on a separate AAA server and sent by the AAA server to the access server or router in response to authorization requests during the PPP authentication phase. The per-user configuration information can add to or override the generic configuration on a virtual interface.
- Virtual profiles, which can use either or both of the two sources of information above for virtual interface configuration. When a user dials in, virtual profiles can apply the generic interface configuration and then apply the per-user configuration to create a unique virtual access interface for that user. This is described in the “Configuring Virtual Profiles” chapter of this manual.

This set of features is supported on all platforms that support Multilink PPP.

A virtual access interface created dynamically for any user dial-in session is deleted when the session ends. The resources used during the session are returned for other dial-in uses.

Background Information

This section provides background information, such as benefits, support, and operational processes.

Benefits

The per-user configuration feature provides these benefits:

- Maintenance ease for service providers with a large number of access servers and a very large number of dial-in users. Service providers do not need to update all their routers and access servers when user-specific information changes; instead, they can update one AAA server.
- Scalability. By separating generic virtual interface configuration on the router from the configuration for each individual, Internet service providers and other enterprises with large numbers of dial-in users can provide a uniquely configured interface for each individual user. In

addition, by separating the generic virtual interface configuration from the physical interfaces on the router, the number and types of physical interfaces on the router or access server are not intrinsic barriers to growth.

Prerequisites

When a specific user dials in to a router, the use of a per-user configuration from an AAA server requires that AAA is configured on the router and that a configuration for that user exists on the AAA server.

Terminology

The following new or uncommon terms are used in this chapter:

AV pair—A configuration parameter on an AAA server; part of the user configuration that the AAA server sends to the router, in response to user-specific authorization requests. The router interprets each AV pair as a Cisco IOS router configuration command and applies the AV pairs in order. In this chapter, the term *AV pair* refers to a configuration parameter on either a TACACS+ or a RADIUS server.

On a TACACS server, an AV pair can take forms such as these:

```
rte-fltr-out#0="router igrp 60"  
rte-fltr-out#3="permit 0.0.5.6 255.255.0.0"  
rte-fltr-out#4="permit any"
```

On a RADIUS server, an AV pair can take forms such as these:

```
cisco-avpair = "lcp:interface-config=ip address 1.1.1.1 255.255.255.255.0",  
cisco-avpair = "ip:rte-fltr-out#0=router igrp 60",  
cisco-avpair = "ip:rte-fltr-out#1=deny 171.0.0.0 0.255.255.255",
```

cloning—Creating and configuring a virtual access interface by applying a specific virtual template interface. The template is the source of the generic user information and router-dependent information. The result of cloning is a virtual access interface configured with all the commands in the template.

virtual access interface—Instance of a unique virtual interface that is created dynamically and exists temporarily. Virtual access interfaces can be created and configured differently by different applications, such as virtual profiles and virtual private dialup networks.

virtual profile—Unique virtual access interface created dynamically when certain users call in and torn down dynamically when the call disconnects. A specific user's virtual profile can be configured by a virtual template interface, user-specific interface configuration (stored on an AAA server), or both a virtual template interface and user-specific interface configuration from AAA.

Configuration of a virtual access interface begins with a virtual template interface (if any), followed by application of user-specific configuration for the particular user's dial-in session (if any).

virtual template interface—Generic configuration of an interface for a certain purpose or configuration common to certain users, plus router-dependent information. This takes the form of a list of Cisco IOS interface commands to be applied to a virtual access interface as needed. Several applications can apply virtual template interfaces, but generally each application uses a single template. Each virtual template interface is identified by number.

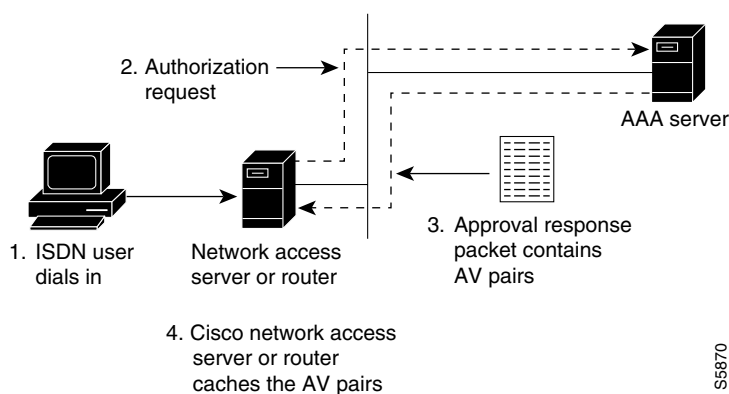
General Operational Processes

In general, the per-user configuration process on the Cisco router or network access server proceeds as follows:

- 1 The user dials in.
- 2 The authentication and authorization phases occur.
 - If AAA is configured, the router sends an authorization request to the AAA server.
 - If the AAA server has information (AV pairs or other configuration parameters) that defines a configuration for the specific user, the server includes it in the information in the approval response packet.
 - Figure 126 illustrates the request and response part of the process that happens when a user dials in, given that AAA is configured and that the AAA server has per-user configuration information for the dial-in user.
 - The router looks for AV pairs in the AAA approval response.
 - The router caches the configuration parameters.

Note TACACS servers treat authentication and authorization as two phases; RADIUS servers combine authentication and authorization into a single step. For more detailed information, refer to your server documentation.

Figure 126 Per-User Configuration Authentication and Authorization



- 3 A virtual access interface is created for this user.
 - The router finds the virtual template that is set up for virtual profiles, if any, and applies the commands to the virtual access interface.
 - The router looks for the AV pairs to apply to this virtual access interface to configure it for the dial-in user.
 - The AV pairs are sent to the Cisco IOS command-line parser, which interprets them as configuration commands, and applies them to configure this virtual access interface.

The result of this process is a virtual access interface configured uniquely for the dial-in user.

When this user's call ends, the virtual access interface is deleted and its resources are returned for other dial-in uses.

Note The use of virtual profiles can modify the process that occurs between the user dial-in and the use of AAA configuration information. For more information, see the “Configuring Virtual Profiles” chapter of this manual.

Operational Processes with IP Address Pooling

During IPCP address negotiation, if an IP pool name is specified for a user, the network access server checks whether the named pool is defined locally. If it is, no special action is required and the pool is consulted for an IP address.

If the required pool is not present (either in the local configuration or as a result of a previous download operation), then an authorization call to obtain it is made, using the special username:

```
pools-nas-name
```

where *nas-name* is the configured name of the network access server. In response, the AAA server downloads the configuration of the required pool.

This pool username can be changed using Cisco IOS configuration. For example:

```
aaa configuration config-name nas1-pools-definition.cisco.us
```

This command has the effect of changing the username that is used to download the pool definitions from the default name “pools-nas-name” to “nas1-pools-definition.cisco.com.”

On a TACACS+ server, the entries for an IP address pool and a user of the pool might be as follows:

```
user = nas1-pools {
  service = ppp protocol = ip {
    pool-def#1 = "aaa 1.0.0.1 1.0.0.3"
    pool-def#2 = "bbb 2.0.0.1 2.0.0.10"
    pool-def#3 = "ccc 3.0.0.1 3.0.0.20"
    pool-timeout=60
  }
}

user = georgia {
  login = cleartext lab
  service = ppp protocol = ip {
    addr-pool=bbb
  }
}
```

On a RADIUS server, the entries for the same IP address pool and user is as follows:

```
nas1-pools      Password = "cisco" User-Service-Type=Outbound-User
  cisco-avpair = "ip:pool-def#1=aaa 1.0.0.1 1.0.0.3",
  cisco-avpair = "ip:pool-def#2=bbb 2.0.0.1 2.0.0.10",
  cisco-avpair = "ip:pool-def#3=ccc 3.0.0.1 3.0.0.20",
  cisco-avpair = "ip:pool-timeout=60"

georgia Password = "lab"
  User-Service-Type = Framed-User,
  Framed-Protocol = PPP,
  cisco-avpair = "ip:addr-pool=bbb"
```

Note This entry specifies a User-Service-Type of Outbound-User. This attribute is supplied by the network access server to prevent ordinary logins from using the well-known username and password combination of nas1-pools/cisco.

Pools downloaded to a Cisco network access server are not retained in nonvolatile memory and automatically disappear whenever the access server or router restarts. Downloaded pools can also be made to time-out automatically by adding a suitable AV pair. See the bullets below and the pool-timeout attribute in Table 29. Downloaded pools are marked as *dynamic* in the output of the **show ip local pools** command.

To delete downloaded pools, you can do either of the following:

- Manually delete the definition from the network access server. For example, if “bbb” is the name of a downloaded pool, you can enter the Cisco IOS command **no ip local pool bbb**.

Deleting a pool definition does not interrupt service for current users. If a pool is deleted and then redefined to include a pool address that is currently allocated, the new pool understands and tracks the address as expected.

- Set an AV pair pool-timeout value; this is a more desirable solution.

The pool-timeout AV pair starts a timer when the pool is downloaded. Once the timer expires, the pools are deleted. The next reference to the pools again causes an authorization call to be made, and the pool definition is downloaded again.

This method allows definitions to be made and changed on the AAA server and propagated to network access servers.

Supported Attributes for AV Pairs

Table 29 provides a partial list of the Cisco-specific supported attributes for AV pairs that can be used for per-user virtual interface configuration. For complete lists of attributes of Cisco-specific, vendor-specific, and TACACS+ supported attributes, see the *Security Configuration Guide* and *Security Command Reference*.

Table 29 Partial List of Cisco-Specific Supported AV Pair Attributes

Attribute	Meaning
inacl#	An input access list definition. For IP, standard or extended access list syntax can be used, though you cannot mix them within a single list. For IPX, only extended syntax is recognized. The value of this attribute is the text that comprises the body of a named access list definition.
outacl# ¹	An output access list definition. For IP, standard or extended access list syntax can be used. For IPX, only extended syntax is recognized. The value of this attribute is the text that comprises the body of a named access list definition.
rte-fltr-in#	An input route filter. For IP, standard or extended access list syntax can be used, though you cannot mix them within a single list. For IPX, only extended syntax is recognized. The first line of this filter must specify a routing process. Subsequent lines comprise the body of a named access list.
rte-fltr-out#	An output route filter. For IP, standard or extended access list syntax can be used, though you cannot mix them within a single list. For IPX, only extended syntax is recognized. The first line of this filter must specify a routing process. Subsequent lines comprise the body of a named access list.
route# ²	Static routes, for IP and IPX. The value is text of the form <i>destination-address mask [gateway]</i>
sap#	IPX static SAPs. The value is text from the body of an ipx sap configuration command.

Table 29 Partial List of Cisco-Specific Supported AV Pair Attributes (Continued)

Attribute	Meaning
sap-fltr-in#	IPX input SAP filter. Only extended access list syntax is recognized. The value is text from the body of an extended IPX access list configuration command. (The Novell socket number for SAP filtering is 452.)
sap-fltr-out#	IPX output SAP filter. Only extended access list syntax is recognized. The value is text from the body of an extended IPX access list configuration command.
pool-def#	An IP pool definition. The value is text from the body of an ip local pool configuration command.
pool-timeout	An IP pool definition. The body is an integer representing a timeout, in minutes.

1. The “outacl” attribute still exists and retains its old meaning.
2. The “route” attribute, without a trailing #, is still recognized for backward compatibility with the TACACS+ protocol specification, but if multiple static routes are required in TACACS+, full “route#” names will have to be employed.

Table 30 provides examples for each attribute on an AAA TACACS+ server.

Table 30 TACACS+ Server AV Pair Examples for Each Attribute

Attribute	TACACS+ Server Examples
inacl#	<p>IP: inacl#3="permit ip any any precedence immediate" inacl#4="deny igrp 0.0.1.2 255.255.0.0 any"</p> <p>IPX: inacl#1="deny 3C01.0000.0000.0001" inacl#2="deny 4C01.0000.0000.0002"</p>
outacl#	<p>outacl#2="permit ip any any precedence immediate" outacl#3="deny igrp 0.0.9.10 255.255.0.0 any"</p>
rte-fltr-in#	<p>IP: rte-fltr-in#0="router igrp 60" rte-fltr-in#3="permit 0.0.3.4 255.255.0.0" rte-fltr-in#4="deny any"</p> <p>IPX: rte-fltr-in#1="deny 3C01.0000.0000.0001" rte-fltr-in#2="deny 4C01.0000.0000.0002"</p>
rte-fltr-out#	<p>rte-fltr-out#0="router igrp 60" rte-fltr-out#3="permit 0.0.5.6 255.255.0.0" rte-fltr-out#4="permit any"</p>
route#	<p>IP: route#1="3.0.0.0 255.0.0.0 1.2.3.4" route#2="4.0.0.0 255.0.0.0"</p> <p>IPX: route#1="4C000000 ff000000 30.12.3.4" route#2="5C000000 ff000000 30.12.3.5"</p>
sap#	<p>sap#1="4 CE1-LAB 1234.0000.0000.0001 451 4" sap#2="5 CE3-LAB 2345.0000.0000.0001 452 5"</p>
sap-fltr-in#	<p>sap-fltr-in#1="deny 6C01.0000.0000.0001" sap-fltr-in#2="permit -1"</p>
sap-fltr-out#	<p>sap-fltr-out#1="deny 6C01.0000.0000.0001" sap-fltr-out#2="permit -1"</p>

Attribute	TACACS+ Server Examples
pool-def#	pool-def#1 = "aaa 1.0.0.1 1.0.0.3" pool-def#2 = "bbb 2.0.0.1 2.0.0.10" pool-def#3 = "ccc 3.0.0.1 3.0.0.20"
pool-timeout	pool-timeout=60

Table 31 provides examples for each attribute on an AAA RADIUS server.

Table 31 RADIUS Server AV Pair Examples for Each Attribute

Attribute	RADIUS Server Examples
lcp:interface-config ¹	cisco-avpair = "lcp:interface-config=ip address 1.0.0.0 255.255.255.0",
inacl#	cisco-avpair = "ip:inacl#3=permit ip any any precedence immediate", cisco-avpair = "ip:inacl#4=deny igmp 0.0.1.2 255.255.0.0 any",
outacl#	cisco-avpair = "ip:outacl#2=permit ip any any precedence immediate", cisco-avpair = "ip:outacl#3=deny igmp 0.0.9.10 255.255.0.0 any",
rte-fltr-in#	IP: cisco-avpair = "ip:rte-fltr-in#0=router igmp 60", cisco-avpair = "ip:rte-fltr-in#3=permit 0.0.3.4 255.255.0.0", cisco-avpair = "ip:rte-fltr-in#4=deny any", IPX: cisco-avpair = "ipx:rte-fltr-in=deny 3C01.0000.0000.0001",
rte-fltr-out#	cisco-avpair = "ip:rte-fltr-out#0=router igmp 60", cisco-avpair = "ip:rte-fltr-out#3=permit 0.0.5.6 255.255.0.0", cisco-avpair = "ip:rte-fltr-out#4=permit any",
route#	IP: cisco-avpair = "ip:route=3.0.0.0 255.0.0.0 1.2.3.4", cisco-avpair = "ip:route=4.0.0.0 255.0.0.0", IPX: cisco-avpair = "ipx:route=4C000000 ff000000 30.12.3.4", cisco-avpair = "ipx:route=5C000000 ff000000 30.12.3.5"
sap#	cisco-avpair = "ipx:sap=4 CE1-LAB 1234.0000.0000.0001 451 4", cisco-avpair = "ipx:sap=5 CE3-LAB 2345.0000.0000.0001 452 5",
sap-fltr-in#	cisco-avpair = "ipx:sap-fltr-in=deny 6C01.0000.0000.0001", cisco-avpair = "ipx:sap-fltr-in=permit -1"
sap-fltr-out#	cisco-avpair = "ipx:sap-fltr-out=deny 6C01.0000.0000.0001", cisco-avpair = "ipx:sap-fltr-out=permit -1"
pool-def#	cisco-avpair = "ip:pool-def#1=aaa 1.0.0.1 1.0.0.3", cisco-avpair = "ip:pool-def#2=bbb 2.0.0.1 2.0.0.10", cisco-avpair = "ip:pool-def#3=ccc 3.0.0.1 3.0.0.20",
pool-timeout	cisco-avpair = "ip:pool-timeout=60"

1. This attribute is specific to RADIUS servers. It can be used to add Cisco IOS interface configuration commands to a specific user's configuration information.

AAA Server Configuration for a Per-User Configuration Solution

The configuration requirements and the structure of per-user configuration information is set by the specifications of each type of AAA server. Refer to your server documentation for more detailed information. The following sections about TACACS and RADIUS servers are specific to per-user configuration.

Configuring a Freeware TACACS Server for Per-User Configuration

On a TACACS server, the entry in the users file takes a standard form. In the freeware version of TACACS+, the following lines appear in order:

- “User =” followed by the username, a space and an open brace
- Authentication parameters
- Authorization parameters
- One or more AV pairs
- End brace on a line by itself

The general form of a freeware TACACS user entry is shown by the following general example:

```
user = username {
    authentication parameters go here
    authorization parameters go here
}
```

The freeware TACACS user entry form is also shown by the following examples for specific users:

```
user= tx
    Password= cleartext welcome
    Service= PPP protocol= ip {
        ip:route=10.0.0.0 255.0.0.0
        ip:route=11.0.0.0 255.0.0.0
        ip:route=12.0.0.0 255.0.0.0
        ip:inacl#5=deny 20.0.0.1
    }

user= frox
    Password= cleartext lab
    Service= PPP protocol= ip {
        ip:addr-pool=bbb
    }
```

For more requirements and detailed information, see your AAA server documentation.

Configuring a CiscoSecure TACACS Server for Per-User Configuration

The format of an entry in the users file in the AAA database is generally name = value. Some values allow additional subparameters to be specified and, in these cases, the subparameters are enclosed in braces ({}). Below is a simple example of an AAA database showing the default user, one group, two users who belong to the group, and one user who does not:

```
# Sample AA Database 1
unknown_user = {
    password = system #Use the system's password file (/etc/passwd)
}
group = staff {
    # password for staff who do not have their own
    password = des "sefjkAlM7zybE"
```

```
service = shell {
    # allow any commands with any attributes
    default cmd = permit
    default attribute = permit
}
}
user = joe { # joe uses the group password

    member = "staff"
}
user = pete { # pete has his own password
    member = "staff"
    password = des "alkd9Ujiqp2y"
}
user = anita {
    # Use the "default" user password mechanism defined above.
    service = shell {
        cmd = telnet { # allow telnet to any destination
        }
    }
}
}
```

For more information about the requirements and details of configuring the CiscoSecure server, see the *CiscoSecure UNIX Server User Guide*.

Configuring a RADIUS Server for Per-User Configuration

On a RADIUS server, the format of an entry in the users file includes the following lines in order:

- Username and password
- User service type
- Framed Protocol
- One or more AV pairs

Note All these AV pairs are vendor specific. To use them, RADIUS servers must support the use of vendor-specific AV pairs. Patches for some servers are available from the Cisco Consulting Engineering (CE) customer-support organization.

The structure of an AV pair for Cisco platforms starts with *cisco-avpair* followed by a space, an equal sign, and another space. The rest of the line is within double quotation marks and, for all lines but the last, ends with a comma. Inside the double quotation marks is a phrase indicating the supported attribute, another equal sign, and a Cisco IOS command. The following examples show two quite different partial user configurations on a RADIUS server.

```
tx      Password = "welcome"
        User-Service-Type = Framed-User,
        Framed-Protocol = PPP,
        cisco-avpair = "ip:route=10.0.0.0 255.0.0.0",
        cisco-avpair = "ip:route=11.0.0.0 255.0.0.0",
        cisco-avpair = "ip:route=12.0.0.0 255.0.0.0",
        cisco-avpair = "ip:inacl#5=deny 20.0.0.1"

frox    Password = "lab"
        User-Service-Type = Framed-User,
        Framed-Protocol = PPP,
        cisco-avpair = "ip:addr-pool=bbb"
```

Configuration Tasks

Per-user configuration information exists on AAA servers only and is configured there, as described in the “AAA Server Configuration for a Per-User Configuration Solution” section.

For more information about configuring an application that can tie AAA per-user configuration information to generic interface and router configuration, see the “Configuring Virtual Profiles” chapter of this manual. Virtual profiles are required for combining per-user configuration information and generic interface and router configuration information to create virtual access interfaces for individual ISDN B channels.

However, you can monitor and debug the per-user configuration settings on the router or access server that are set from an AAA server. Table 32 indicates some of the commands to use for each AV-pair attribute.

Table 32 Monitoring and Debugging Per-User Configuration Settings

Attribute	Show Commands	Debug Commands
inacl#	show ip access-list	debug aaa author
outacl#	show ip interface <i>interface</i> show ipx access-lists show ipx interface	debug aaa per-user
rte-fltr-in#	show ip access-lists	debug aaa author
rte-fltr-out#	show ip protocols	debug aaa per-user
route#	show ip route show ipx route	debug aaa author debug aaa per-user
sap#	show ipx servers	debug aaa author debug aaa per-user
sap-fltr-in#	show ipx access-lists	debug aaa author
sap-fltr-out#	show ipx interface	debug aaa per-user
pool-def#	show ip local pool <i>[name]</i>	—
pool-timeout		

Per-User Configuration Examples

This section provides TACACS+ and RADIUS variations of two comprehensive examples, which show router or access server configuration and AV pair configuration on an AAA server.

TACACS+ Freeware Examples

This section provides the TACACS+ freeware versions of the following examples:

- IP Access Lists and Static Routes Using Virtual Profiles over ISDN BRI Example
- IPX Per-User SAP Filters, Using IPXWAN and Virtual Profiles by a Synchronous Interface Example

IP Access Lists and Static Routes Using Virtual Profiles over ISDN BRI Example

This example provides configurations for the TACACS+ freeware daemon, the network access server, and the peer router named tx. On the TACACS+ AAA server, peer router tx has a configuration that includes static routes and IP access lists.

TACACS+ Freeware Daemon Configuration File

```
key = tac123
user = tx {
  global = cleartext welcome
  service = ppp protocol = ip {
    route#1="10.0.0.0 255.0.0.0"
    route#2="11.0.0.0 255.0.0.0"
    route#3="12.0.0.0 255.0.0.0"
    inacl#1="deny 20.0.0.1"
  }
}
```

Current Network Access Server Configuration

```
version 11.3
service timestamps debug datetime localtime
service udp-small-servers
service tcp-small-servers
!
hostname sf
!
aaa new-model
aaa authentication ppp default tacacs+
aaa authorization network tacacs+
enable secret 5 $1$koOn$/1QAYlov6JFAElxRCrL.o/
enable password lab
!
username tx password 7 15050E0007252621
ip host sf 172.21.114.132
ip domain-name cisco.com
ip name-server 171.69.2.132
ip name-server 198.92.30.32
isdn switch-type basic-5ess
interface Ethernet0
  ip address 172.21.114.132 255.255.255.224
  no ip mroute-cache
  media-type 10BaseT
!
interface Virtual-Template1
  ip unnumbered Ethernet0
  no cdp enable
!
!
interface BRI0
  ip unnumbered Ethernet0
  no ip mroute-cache
  encapsulation ppp
  no ip route-cache
  dialer idle-timeout 300
  dialer map ip 20.0.0.1 name tx broadcast 61482
  dialer-group 1
  no fair-queue
  ppp authentication chap
!
!
```

```
ip default-gateway 172.21.114.129
no ip classless
ip route 0.0.0.0 0.0.0.0 172.21.114.129
!
virtual-profile virtual-template 1
dialer-list 1 protocol ip permit
tacacs-server host 172.21.114.130
tacacs-server key tac123
```

Current Peer Configuration for tx

```
version 11.3
no service pad
!
hostname tx
!
enable secret 5 $1$m1WK$RsjborN1Z.XZuFqsrtSnp/
enable password lab
!
username sf password 7 051C03032243430C
ip host tx 172.21.114.134
ip domain-name cisco.com
ip name-server 171.69.2.132
ip name-server 198.92.30.32
isdn switch-type basic-5ess
!
interface Ethernet0
ip address 172.21.114.134 255.255.255.224
no ip route-cache
shutdown
!
interface BRI0
ip address 20.0.0.1 255.0.0.0
encapsulation ppp
dialer map ip 172.21.114.132 name sf broadcast 61483
dialer-group 1
no fair-queue
!
ip default-gateway 172.21.114.129
no ip classless
ip route 172.21.0.0 255.255.0.0 BRI0
dialer-list 1 protocol ip permit
!
line con 0
exec-timeout 0 0
line vty 0 4
password lab
login
end
```

IPX Per-User SAP Filters, Using IPXWAN and Virtual Profiles by a Synchronous Interface Example

This example provides configurations for the TACACS+ daemon and the peer router named ny. On the TACACS+ AAA server, user ny has a configuration that includes inbound and outbound SAP filters.

TACACS+ Freeware Daemon User's Configuration File

key = tac123

```

user = ny {
  global = cleartext welcome
  service = ppp protocol = ipx {
    sap="101 CYBER-01 40.0000.0000.0001 400 10"
    sap="202 CYBER-02 40.0000.0000.0001 401 10"
    sap="303 CYBER-03 40.0000.0000.0001 402 10"
    sap-fltr-out#0="deny 40 101"
    sap-fltr-out#1="deny 40 202"
    sap-fltr-out#2="permit -1"
    sap-fltr-in#0="permit 30 444"
    sap-fltr-in#1="deny -1"
  }
}

```

Current Remote Peer (ny) Configuration

```

version 11.3
!
hostname ny
!
enable password lab
!
username la password 7 140017070F0B272E
ip host NY 172.21.114.131
ip name-server 171.69.2.132
ip name-server 198.92.30.32
ipx routing 0000.0c47.090d
ipx internal-network 30
!
interface Ethernet0
  ip address 172.21.114.131 255.255.255.224
!
!
interface Serial1
  no ip address
  encapsulation ppp
  ipx ipxwan 0 unnumbered peer-ny
  clockrate 4000000
!
ipx sap 444 ZEON-4 30.0000.0000.0001 444 10
ipx sap 555 ZEON-5 30.0000.0000.0001 555 10
ipx sap 666 ZEON-6 30.0000.0000.0001 666 10
!
Current Network Access Server (la) Configuration
version 11.3
service timestamps debug uptime
!
hostname la
!
aaa new-model
aaa authentication ppp default tacacs+
aaa authorization network tacacs+
enable password lab
!

```

```
username ny password 7 044C0E0A0C2E414B
ip host LA 172.21.114.133
ip name-server 198.92.30.32
ip name-server 171.69.2.132
ipx routing 0000.0c47.12d3
ipx internal-network 40
!
interface Ethernet0
 ip address 172.21.114.133 255.255.255.224
!
interface Virtual-Template1
 no ip address
 ipx ipxwan 0 unnumbered nas-la
 no cdp enable
!
interface Serial1
 ip unnumbered Ethernet0
 encapsulation ppp
 ipx ipxwan 0 unnumbered nas-la
 ppp authentication chap
!
ipx sap 333 DEEP9 40.0000.0000.0001 999 10
!
virtual-profile virtual-template 1
tacacs-server host 172.21.114.130
tacacs-server key tac123
```

RADIUS Examples

This section provides the RADIUS versions of the following examples:

- IP Access Lists and Static Routes Using Virtual Profiles over ISDN BRI Example
- IPX Per-User SAP Filters, Using IPXWAN and Virtual Profiles by a Synchronous Interface Example

IP Access Lists and Static Routes Using Virtual Profiles over ISDN BRI Example

In this example, a remote peer (tx) dials in to a BRI on a Cisco network access server (sf), which requests user configuration information from an AAA server (radiusd).

RADIUS Users File

```
tx      Password = "welcome"
        User-Service-Type = Framed-User,
        Framed-Protocol = PPP,
        cisco-avpair = "ip:route=10.0.0.0 255.0.0.0",
        cisco-avpair = "ip:route=11.0.0.0 255.0.0.0",
        cisco-avpair = "ip:route=12.0.0.0 255.0.0.0",
        cisco-avpair = "ip:inacl#5=deny 20.0.0.1"
```

Current Network Access Server Configuration

```
version 11.3
service timestamps debug datetime localtime
service udp-small-servers
service tcp-small-servers
!
hostname sf
!
```

```

aaa new-model
aaa authentication ppp default radius
aaa authorization network radius
enable secret 5 $1$koOn$/1QAYlov6JFAElxRCrL.o/
enable password lab
!
username tx password 7 15050E0007252621
ip host sf 172.21.114.132
ip domain-name cisco.com
ip name-server 171.69.2.132
ip name-server 198.92.30.32
isdn switch-type basic-5ess
interface Ethernet0
 ip address 172.21.114.132 255.255.255.224
 no ip mroute-cache
 media-type 10BaseT
!
!
interface Virtual-Template1
 ip unnumbered Ethernet0
 no cdp enable
!
!
interface BRI0
 ip unnumbered Ethernet0
 no ip mroute-cache
 encapsulation ppp
 no ip route-cache
 dialer idle-timeout 300
 dialer map ip 20.0.0.1 name tx broadcast 61482
 dialer-group 1
 no fair-queue
 ppp authentication chap
!
ip default-gateway 172.21.114.129
no ip classless
ip route 0.0.0.0 0.0.0.0 172.21.114.129
!
virtual-profile vtemplate 1
dialer-list 1 protocol ip permit
radius-server host 172.21.114.130
radius-server key rad123

```

Current Peer Configuration for tx

```

version 11.3
no service pad
!
hostname tx
!
enable secret 5 $1$m1WK$RsjborN1Z.XZuFqsrtSnp/
enable password lab
!
username sf password 7 051C03032243430C
ip host tx 172.21.114.134
ip domain-name cisco.com
ip name-server 171.69.2.132
ip name-server 198.92.30.32
isdn switch-type basic-5ess
!
interface Ethernet0
 ip address 172.21.114.134 255.255.255.224
 no ip route-cache
 shutdown

```

```
!  
interface BRI0  
  ip address 20.0.0.1 255.0.0.0  
  encapsulation ppp  
  dialer map ip 172.21.114.132 name sf broadcast 61483  
  dialer-group 1  
  no fair-queue  
!  
ip default-gateway 172.21.114.129  
no ip classless  
ip route 172.21.0.0 255.255.0.0 BRI0  
dialer-list 1 protocol ip permit  
!  
line con 0  
  exec-timeout 0 0  
line vty 0 4  
  password lab  
  login  
!  
end
```

Outputs of Ping Command from tx

```
tx# ping 172.21.114.132  
  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 172.21.114.132, timeout is 2 seconds:  
U.U.U  
Success rate is 0 percent (0/5)  
  
(fails due to access list deny)
```

Radius Debug Output

```
radrecv: Request from host ac157284 code=1, id=46, length=67  
  Client-Id = 172.21.114.132  
  Client-Port-Id = 1112670208  
  User-Name = "tx"  
  CHAP-Password = "\037\317\213\326*\236)#+\266\243\255x\331\370v\334"  
  User-Service-Type = Framed-User  
  Framed-Protocol = PPP  
Sending Ack of id 46 to ac157284 (172.21.114.132)  
  User-Service-Type = Framed-User  
  Framed-Protocol = PPP  
[Vendor 9] cisco-avpair = "ip:route=10.0.0.0 255.0.0.0"  
[Vendor 9] cisco-avpair = "ip:route=11.0.0.0 255.0.0.0"  
[Vendor 9] cisco-avpair = "ip:route=12.0.0.0 255.0.0.0"  
[Vendor 9] cisco-avpair = "ip:inacl#5=deny 20.0.0.1"
```

Network Access Server (sf) Show and Debug Commands Output

```
sf# show debug  
  
General OS:  
  AAA Authorization debugging is on  
PPP:  
  PPP authentication debugging is on  
  Multilink activity debugging is on  
ISDN:  
  ISDN events debugging is on  
Dial on demand:  
  Dial on demand events debugging is on
```

```

VTEMPLATE:
  Virtual Template debugging is on

pr 4 08:30:09: ISDN BR0: received HOST_INCOMING_CALL
  Bearer Capability i = 0x080010
*Apr 4 08:30:09: -----
  Channel ID i = 0x0101
*Apr 4 08:30:09: IE out of order or end of 'private' IEs --
  Bearer Capability i = 0x8890
*Apr 4 08:30:09: Channel ID i = 0x89
*Apr 4 08:30:09: Called Party Number i = 0xC1, '61483'
*Apr 4 08:30:09: ISDN BR0: Event: Received a call from <unknown> on B1 at 64 Kb/s
*Apr 4 08:30:09: ISDN BR0: Event: Accepting the call
%LINK-3-UPDOWN: Interface BRI0:1, changed state to up
*Apr 4 08:30:09: ISDN BR0: received HOST_CONNECT
  Channel ID i = 0x0101
*Apr 4 08:30:09: -----
  Channel ID i = 0x89
*Apr 4 08:30:09: ISDN BR0: Event: Connected to <unknown> on B1 at 64 Kb/s
*Apr 4 08:30:09: PPP BRI0:1: Send CHAP challenge id=30 to remote
*Apr 4 08:30:10: PPP BRI0:1: CHAP response received from tx
*Apr 4 08:30:10: PPP BRI0:1: CHAP response id=30 received from tx
*Apr 4 08:30:10: AAA/AUTHOR/LCP: authorize LCP
*Apr 4 08:30:10: AAA/AUTHOR/LCP: BRI0:1: (0): user='tx'
*Apr 4 08:30:10: AAA/AUTHOR/LCP: BRI0:1: (0): send AV service=ppp
*Apr 4 08:30:10: AAA/AUTHOR/LCP: BRI0:1: (0): send AV protocol=lcp
*Apr 4 08:30:10: AAA/AUTHOR/LCP: BRI0:1: (2084553184): Method=RADIUS
*Apr 4 08:30:10: AAA/AUTHOR (2084553184): Post authorization status = PASS_ADD
*Apr 4 08:30:10: PPP BRI0:1: Send CHAP success id=30 to remote
*Apr 4 08:30:10: PPP BRI0:1: remote passed CHAP authentication.
*Apr 4 08:30:10: VTEMPLATE Reuse vaccess1, New Recycle queue size:0

*Apr 4 08:30:10: VTEMPLATE set default vaccess1 with no ip address

*Apr 4 08:30:10: Virtual-Access1 VTEMPLATE hardware address 0000.0c46.154a
*Apr 4 08:30:10: VTEMPLATE vaccess1 has a new cloneblk vtemplate, now it has vtemplate
*Apr 4 08:30:10: VTEMPLATE undo default settings vaccess1

*Apr 4 08:30:10: VTEMPLATE ***** CLONE VACCESS1 *****Apr 4
08:30:10: VTEMPLATE Clone from vtemplate1 to vaccess1
interface Virtual-Access1
no ip address
encap ppp
ip unnumbered ethernet 0
end

%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
*Apr 4 08:30:10: AAA/AUTHOR/LCP: authorize LCP
*Apr 4 08:30:10: AAA/AUTHOR/LCP: Virtual-Access1: (0): user='tx'
*Apr 4 08:30:10: AAA/AUTHOR/LCP: Virtual-Access1: (0): send AV service=ppp
*Apr 4 08:30:10: AAA/AUTHOR/LCP: Virtual-Access1: (0): send AV protocol=lcp
*Apr 4 08:30:10: AAA/AUTHOR/LCP: Virtual-Access1: (1338953760): Method=RADIUS
*Apr 4 08:30:10: AAA/AUTHOR (1338953760): Post authorization status = PASS_ADD
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): can we start IPCP?
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): user='tx'
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): send AV service=ppp
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): send AV protocol=ip
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (1716082074): Method=RADIUS
*Apr 4 08:30:10: AAA/AUTHOR (1716082074): Post authorization status = PASS_ADD
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: we can start IPCP (0x8021)
*Apr 4 08:30:10: MLP Bad link Virtual-Access1
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): can we start UNKNOWN?
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): user='tx'
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): send AV service=ppp
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (0): send AV protocol=unknown

```

Per-User Configuration Examples

```
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: (2526612868): Method=RADIUS
*Apr 4 08:30:10: AAA/AUTHOR (2526612868): Post authorization status = PASS_ADD
*Apr 4 08:30:10: AAA/AUTHOR/FSM: Virtual-Access1: we can start UNKNOWN (0x8207)
*Apr 4 08:30:10: MLP Bad link Virtual-Access1
*Apr 4 08:30:10: BRI0:1: Vaccess started from dialer_remote_name
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): can we start IPCP?
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): user='tx'
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): send AV service=ppp
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): send AV protocol=ip
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (3920403585): Method=RADIUS
*Apr 4 08:30:10: AAA/AUTHOR (3920403585): Post authorization status = PASS_ADD
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: we can start IPCP (0x8021)
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): can we start UNKNOWN?
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): user='tx'
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): send AV service=ppp
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (0): send AV protocol=unknown
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: (3439943223): Method=RADIUS
*Apr 4 08:30:10: AAA/AUTHOR (3439943223): Post authorization status = PASS_ADD
*Apr 4 08:30:10: AAA/AUTHOR/FSM: BRI0:1: we can start UNKNOWN (0x8207)
%LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: start: her address 20.0.0.1, we want
0.0.0.0
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: (0): user='tx'
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: (0): send AV servi*Apr 4 08:30:13:
AAA/AUTHOR/IPCP: Virtual-Access1: (0): send AV service=ppp
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: (0): send AV protocol=ip
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: (0): send AV addr*20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: (3215797579): Method=RADIUS
*Apr 4 08:30:13: AAA/AUTHOR (3215797579): Post authorization status = PASS_ADD
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV service=ppp
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV protocol=ip
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV addr*20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV route=10.0.0.0
255.0.0.0
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV route=11.0.0.0
255.0.0.0
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV route=12.0.0.0
255.0.0.0
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV inacl#5=deny 20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: authorization succeeded
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: done: her address 20.0.0.1, we want
20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR/IPCP: Virtual-Access1: authorization succeeded
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: parse_cmd 'ip route 10.0.0.0 255.0.0.0
20.0.0.1' ok (0)
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: enqueue peruser IP txt=no ip route 10.0.0.0
255.0.0.0 20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: parse_cmd 'ip route 11.0.0.0 255.0.0.0
20.0.0.1' ok (0)
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: enqueue peruser IP txt=no ip route 11.0.0.0
255.0.0.0 20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: parse_cmd 'ip route 12.0.0.0 255.0.0.0
20.0.0.1' ok (0)
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: enqueue peruser IP txt=no ip route 12.0.0.0
255.0.0.0 20.0.0.1
*Apr 4 08:30:13: AAA/AUTHOR: parse 'ip access-list standard Virtual-Access1#0' ok (0)
*Apr 4 08:30:13: AAA/AUTHOR: parse 'deny 20.0.0.1' ok (0)
*Apr 4 08:30:13: AAA/AUTHOR: Virtual-Access1: enqueue peruser IP txt=no ip access-list
standard Virtual-Access1#0
*Apr 4 08:30:13: VTEMPLATE vaccess1 has a new cloneblk AAA, now it has vtemplate/AAA
*Apr 4 08:30:13: VTEMPLATE ***** CLONE VACCESS1 *****

*Apr 4 08:30:13: VTEMPLATE Clone from AAA to vaccess1
interface Virtual-Access1
```

```
ip access-group Virtual-Access1#0 in
end

*Apr  4 08:30:13: AAA/AUTHOR: Virtual-Access1: vaccess parse `interface Virtual-Access1
ip access-group Virtual-Access1#0 in
` ok (0)
*Apr  4 08:30:13: AAA/AUTHOR/FSM: Check for unauthorized mandatory AV's
*Apr  4 08:30:13: AAA/AUTHOR/FSM: Processing AV service=ppp
*Apr  4 08:30:13: AAA/AUTHOR/FSM: Processing AV protocol=unknown
*Apr  4 08:30:13: AAA/AUTHOR/FSM: succeeded
%ISDN-6-CONNECT: Interface BRI0:1 is now connected to  tx
sf#

sf# show ip access-lists

Standard IP access list Virtual-Access1#0 (per-user)
deny  20.0.0.1

sf# show ip route

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR

Gateway of last resort is 172.21.114.129 to network 0.0.0.0

U    10.0.0.0/8 [1/0] via 20.0.0.1
U    11.0.0.0/8 [1/0] via 20.0.0.1
U    12.0.0.0/8 [1/0] via 20.0.0.1
     20.0.0.0/8 is subnetted, 1 subnets
C      20.0.0.1 is directly connected, Virtual-Access1
     172.21.0.0/16 is subnetted, 1 subnets
C      172.21.114.128 is directly connected, Ethernet0
S*   0.0.0.0/0 [1/0] via 172.21.114.129

sf# show interfaces virtual-access 1

Virtual-Access1 is up, line protocol is up
Hardware is Virtual Access interface
Interface is unnumbered. Using address of Ethernet0 (172.21.114.132)
MTU 1500 bytes, BW 64 Kbit, DLY 100000 usec, rely 255/255, load 1/255
Encapsulation PPP, loopback not set, keepalive set (10 sec)
DTR is pulsed for 5 seconds on reset
LCP Open, multilink Closed
Open: IPCP, CDP
Last input 5d04h, output never, output hang never
Last clearing of "show interface" counters 00:06:42
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
   76 packets input, 3658 bytes, 0 no buffer
   Received 0 broadcasts, 0 runts, 0 giants
   0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  141 packets output, 2909 bytes, 0 underruns
   0 output errors, 0 collisions, 0 interface resets
   0 output buffer failures, 0 output buffers swapped out
   0 carrier transitions
```

```
sf# show ip interface virtual-access 1
```

```
Virtual-Access1 is up, line protocol is up
Interface is unnumbered. Using address of Ethernet0 (172.21.114.132)
Broadcast address is 255.255.255.255
Peer address is 20.0.0.1
MTU is 1500 bytes
Helper address is not set
Directed broadcast forwarding is enabled
Outgoing access list is not set
Inbound access list is Virtual-Access1#0
Proxy ARP is enabled
Security level is default
Split horizon is enabled
ICMP redirects are always sent
ICMP unreachable are always sent
ICMP mask replies are never sent
IP fast switching is disabled
```

```
sf# debug ip packet
```

```
IP packet debugging is on
sf#
*Apr 4 08:30:42: IP: s=172.21.114.129 (Ethernet0), d=255.255.255.255, len 186, rcvd 2
*Apr 4 08:30:42: IP: s=20.0.0.1 (Virtual-Access1), d=172.21.114.132, len 104, a*Apr 4
08:30:42: IP: s=20.0.0.1 (Virtual-Access1), d=172.21.114.132, len 104, access denied
*Apr 4 08:30:42: IP: s=172.21.114.132 (local), d=20.0.0.1 (Virtual-Access1), len 4,
sending
*Apr 4 08:30:42: IP: s=20.0.0.1 (Virtual-Access1), d=172.21.114.132, len 104, access
denied
*Apr 4 08:30:44: IP: s=20.0.0.1 (Virtual-Access1), d=172.21.114.132, len 104, access
denied
*Apr 4 08:30:44: IP: s=172.21.114.132 (local), d=20.0.0.1 (Virtual-Access1), len 16,
sending
*Apr 4 08:30:44: IP: s=20.0.0.1 (Virtual-Access1), d=172.21.114.132, len 104, access
denied
```

IPX Per-User SAP Filters, Using IPXWAN and Virtual Profiles by a Synchronous Interface Example

In this example, a remote peer (ny) dials in to a synchronous interface on a Cisco network access server (1a), which requests user configuration information from an AAA server (radiusd).

RADIUS Users File

```
ny Password = "welcome"
User-Service-Type = Framed-User,
Framed-Protocol = PPP,
cisco-avpair = "ipx:sap=101 CYBER-01 40.0000.0000.0001 400 10",
cisco-avpair = "ipx:sap=202 CYBER-02 40.0000.0000.0001 401 10",
cisco-avpair = "ipx:sap=303 CYBER-03 40.0000.0000.0001 402 10",
cisco-avpair = "ipx:sap-fltr-out#20=deny 40 101",
cisco-avpair = "ipx:sap-fltr-out#21=deny 40 202",
cisco-avpair = "ipx:sap-fltr-out#22=permit -1",
cisco-avpair = "ipx:sap-fltr-in#23=permit 30 444",
cisco-avpair = "ipx:sap-fltr-in#23=deny -1"
```

Current Remote Peer (ny) Configuration

```
version 11.3
!
hostname ny
!
enable password lab
!
username la password 7 140017070F0B272E
ip host NY 172.21.114.131
ip name-server 171.69.2.132
ip name-server 198.92.30.32
ipx routing 0000.0c47.090d
ipx internal-network 30
!
interface Ethernet0
 ip address 172.21.114.131 255.255.255.224
!
!
interface Serial1
 no ip address
 encapsulation ppp
 ipx ipxwan 0 unnumbered peer-ny
 clockrate 4000000
!
ipx sap 444 ZEON-4 30.0000.0000.0001 444 10
ipx sap 555 ZEON-5 30.0000.0000.0001 555 10
ipx sap 666 ZEON-6 30.0000.0000.0001 666 10
!
Current Network Access Server (la) Configuration
version 11.3
service timestamps debug uptime
!
hostname la
!
aaa new-model
aaa authentication ppp default radius
aaa authorization network radius
enable password lab
!
username ny password 7 044C0E0A0C2E414B
ip host LA 172.21.114.133
ip name-server 198.92.30.32
ip name-server 171.69.2.132
ipx routing 0000.0c47.12d3
ipx internal-network 40
!
interface Ethernet0
 ip address 172.21.114.133 255.255.255.224
!
interface Virtual-Template1
 no ip address
 ipx ipxwan 0 unnumbered nas-la
 no cdp enable
!
interface Serial1
 ip unnumbered Ethernet0
 encapsulation ppp
 ipx ipxwan 0 unnumbered nas-la
 ppp authentication chap
!
ipx sap 333 DEEP9 40.0000.0000.0001 999 10
!
```

```
virtual-profile vtemplate 1
radius-server host 172.21.114.130
radius-server key rad123
```

RADIUS Debug Output

```
radrecv: Request from host ac157285 code=1, id=23, length=67
  Client-Id = 172.21.114.133
  Client-Port-Id = 1399128065
  User-Name = "ny"
  CHAP-Password = "%(\012I$\262\352\031\276\024\302\277\225\347z\274"
  User-Service-Type = Framed-User
  Framed-Protocol = PPP
Sending Ack of id 23 to ac157285 (172.21.114.133)
  User-Service-Type = Framed-User
  Framed-Protocol = PPP
[Vendor 9] cisco-avpair = "ipx:sap=101 CYBER-01 40.0000.0000.0001 400 10"
[Vendor 9] cisco-avpair = "ipx:sap=202 CYBER-02 40.0000.0000.0001 401 10"
[Vendor 9] cisco-avpair = "ipx:sap=303 CYBER-03 40.0000.0000.0001 402 10"
[Vendor 9] cisco-avpair = "ipx:sap-fltr-out#20=deny 40 101"
[Vendor 9] cisco-avpair = "ipx:sap-fltr-out#21=deny 40 202"
[Vendor 9] cisco-avpair = "ipx:sap-fltr-out#22=permit -1"
[Vendor 9] cisco-avpair = "ipx:sap-fltr-in#23=permit 30 444"
[Vendor 9] cisco-avpair = "ipx:sap-fltr-in#23=deny -1"
```

Network Access Server Show Command Output

```
la# show ipx servers

Codes: S - Static, P - Periodic, E - EIGRP, N - NLSP, H - Holddown, + = detail
5 Total IPX Servers
```

Table ordering is based on routing and server info

Type	Name	Net	Address	Port	Route	Hops	Itf
s	101 CYBER-01	40.0000.0000.0001		0400	conn	10	Int
s	202 CYBER-02	40.0000.0000.0001		0401	conn	10	Int
s	303 CYBER-03	40.0000.0000.0001		0402	conn	10	Int
S	333 DEEP9	40.0000.0000.0001		0999	conn	10	Int
P	444 ZEON-4	30.0000.0000.0001		0444	7/01	11	Vi1

```
ny# show ipx servers
```

```
Codes: S - Static, P - Periodic, E - EIGRP, N - NLSP, H - Holddown, + = detail
5 Total IPX Servers
```

Table ordering is based on routing and server info

Type	Name	Net	Address	Port	Route	Hops	Itf
P	303 CYBER-03	40.0000.0000.0001		0402	7/01	11	Se1
P	333 DEEP9	40.0000.0000.0001		0999	7/01	11	Se1
S	444 ZEON-4	30.0000.0000.0001		0444	conn	10	Int
S	555 ZEON-5	30.0000.0000.0001		0555	conn	10	Int
S	666 ZEON-6	30.0000.0000.0001		0666	conn	10	Int

```
la# show ipx access-lists

IPX sap access list Virtual-Access1#2
  permit 30 444
  deny FFFFFFFF
IPX sap access list Virtual-Access1#3
  deny 40 101
  deny 40 202
  permit FFFFFFFF
```

