Configuring Virtual Private Networks

This chapter describes how to configure, verify, maintain, and troubleshoot a Virtual Private Network (VPN). It includes the following main sections:

To identify the hardware platform or software image information associated with a feature, use the Feature Navigator on Cisco.com to search for information about the feature, or refer to the software release notes for a specific release. For more information, see the "Identifying Supported Platforms" section in the "Using Cisco IOS Software" chapter.

For a complete description of the commands mentioned in this chapter, refer to the *Cisco IOS Dial Technologies Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

VPN Technology Overview

A VPN carries private data over a public network and extends remote access to users over a shared infrastructure. VPNs maintain the same security and management policies as a private network. They are the most cost-effective method of establishing a point-to-point connection between remote users and a central network.

A benefit of VPNs or, more appropriately, access VPNs, is the way they delegate responsibilities for the network. The customer outsources the responsibility for the information technology (IT) infrastructure to an Internet service provider (ISP) that maintains the modems that the remote users dial in to (called modem pools), the access servers, and the internetworking expertise. The customer is then only responsible for authenticating its users and maintaining its network.

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Instead of connecting directly to the network by using the expensive Public Switched Telephone Network (PSTN), access VPN users need only use the PSTN to connect to the ISP local point of presence (POP). The ISP then uses the Internet to forward users from the POP to the customer network. Forwarding a user call over the Internet provides dramatic cost savings for the customer. Access VPNs use Layer 2 tunneling technologies to create a virtual point-to-point connection between users and the customer network. These tunneling technologies provide the same direct connectivity as the expensive PSTN by using the Internet. This means that users anywhere in the world have the same connectivity as they would at the customer headquarters.

VPNs allow separate and autonomous protocol domains to share common access infrastructure including modems, access servers, and ISDN routers. VPNs use the following tunneling protocols to tunnel link-level frames:

- Layer 2 Forwarding (L2F)
- Layer 2 Tunneling Protocol (L2TP)
- Point-to-Point Tunneling Protocol (PPTP)

Using one of these protocols, an ISP or other access service can create a virtual tunnel to link customer remote sites or remote users with corporate home networks. In particular, a network access server (NAS) at the ISP POP exchanges PPP messages with the remote users and communicates by L2F, L2TP, or PPTP requests and responses with the customer tunnel server to set up tunnels.

L2F, L2TP, and PPTP pass protocol-level packets through the virtual tunnel between endpoints of a point-to-point connection.

Frames from the remote users are accepted by the ISP POP, stripped of any linked framing or transparency bytes, encapsulated in L2F, L2TP or PPTP, and forwarded over the appropriate tunnel. The customer tunnel server accepts these frames, strips the Layer 2 encapsulation, and processes the incoming frames for the appropriate interface.

Cisco routers fast switch VPN traffic. In stack group environments in which some VPN traffic is offloaded to a powerful router, fast switching provides improved scalability.

VPDN MIB

The VPDN MIB offers a mechanism to track failures of user calls in a VPN system, allowing Simple Network Management Protocol (SNMP) retrieval of user call failure information, on a per-user basis.

Refer to the Cisco VPDN Management MIB for a list of supported objects for the VPDN MIB.

VPN Hardware Terminology

As new tunneling protocols have been developed for VPNs, new terminology has been created to describe the hardware involved in VPNs. Fundamentally, two routers are needed for a VPN:

- Network access server (NAS)—It receives incoming calls for dial-in VPNs and places outgoing calls for dial-out VPNs. Typically it is maintained by an ISP that wishes to provide VPN services to its customers.
- Tunnel server—It terminates dial-in VPNs and initiates dial-out VPNs. Typically it is maintained by the ISP customer and is the contact point for the customer network.

For the sake of clarity, we will use these generic terms, and not the technology-specific terms. Table 29 lists the generic terms ant the technology-specific terms that are often used for these devices.

Table 29 VPN Hardware Terminology

Generic Term	L2F Term	L2TP Term	PPTP Term
Tunnel Server	Home Gateway	L2TP Network Server (LNS)	PPTP Network Server (PNS)
Network Access Server (NAS)	NAS	L2TP Access Concentrator (LAC)	PPTP Access Concentrator (PAC)

In dial-in scenarios, users dial in to the NAS, and the NAS forwards the call to the tunnel server using a VPN tunnel.

In dial-out scenarios, the tunnel server initiates a VPN tunnel to the NAS, and the NAS dials out to the clients.

VPN Architectures

VPNs are designed on the basis of one of two architectural options:

- Client-Initiated VPNs
- NAS-Initiated VPNs

Client-Initiated VPNs

Users establish a tunnel across the ISP shared network to the customer network without an intermediate NAS participating in the tunnel negotiation and establishment. The customer manages the client software that initiates the tunnel. The main advantage of client-initiated VPNs is that they secure the connection between the client and the ISP. However, client-initiated VPNs are not as scalable and are more complex than NAS-initiated VPNs.

Client-initiated VPNs are also referred to as voluntary tunneling.

NAS-Initiated VPNs

Users dial in to the ISP NAS, which establishes a tunnel to the private network. NAS-initiated VPNs are more robust than client-initiated VPNs and do not require the client to maintain the tunnel-creating software. NAS-initiated VPNs do not encrypt the connection between the client and the ISP, but this is not a concern for most customers because the PSTN is much more secure than the Internet.

NAS-initiated VPNs are also referred to as compulsory tunneling.

Note In Cisco's VPN implementation, PPTP tunnels are client-initiated while L2F and L2TP tunnels are NAS-initiated.

PPTP Dial-In with MPPE Encryption

PPTP is a network protocol that enables the secure transfer of data from a remote client to a private enterprise server by creating a VPN across TCP/IP-based data networks. PPTP supports on-demand, multiprotocol, virtual private networking over public networks, such as the Internet.

Cisco supports client-initiated VPNs using PPTP. Therefore only the client and the tunnel server need to be configured. The client first establishes basic connectivity by dialing in to an ISP. Once the client has established a PPP session, it initiates a PPTP tunnel to the tunnel server. The tunnel server is configured to terminate PPTP tunnels and clone virtual-access interfaces from virtual templates.

Microsoft Point-to-Point Encryption (MPPE) is an outcropping technology that can be used to encrypt PPTP VPNs. It encrypts the entire session from the client to the tunnel server.

This section describes the following aspects of PPTP and MPPE:

PPTP Tunnel Negotiation

The following describes the protocol negotiation events that establish a PPTP tunnel:

1. The client dials in to the ISP and establishes a PPP session.
2. The client establishes a TCP connection with the tunnel server.
3. The tunnel server accepts the TCP connection.
4. The client sends a PPTP SCCRQ message to the tunnel server.
5. The tunnel server establishes a new PPTP tunnel and replies with an SCCRP message.
6. The client initiates the session by sending an OCRQ message to the tunnel server.
7. The tunnel server creates a virtual-access interface.
8. The tunnel server replies with an OCRP message.

Flow Control Alarm

The flow control alarm is a new function that indicates if PPTP detects congestion or lost packets. When a flow control alarm goes off, PPTP reduces volatility and additional control traffic by establishing an accompanying stateful MPPE session.

For more information, see the **pptp flow-control static-rtt** command and the output from the **show vpdn session** command in the "Verifying a Client-Initiated VPN" section.

MPPE Overview

MPPE is an encryption technology developed by Microsoft to encrypt point-to-point links. These PPP connections can be over a dialup line or over a VPN tunnel. MPPE works as a subfeature of Microsoft Point-to-Point Compression (MPPC).

MPPC is a scheme used to compress PPP packets between Cisco and Microsoft client devices. The MPPC algorithm is designed to optimize bandwidth utilization in order to support multiple simultaneous connections.

MPPE is negotiated using bits in the MPPC option within the Compression Control Protocol (CCP) MPPC configuration option (CCP configuration option number 18).

MPPE uses the RC4 algorithm with either 40- or 128-bit keys. All keys are derived from the cleartext authentication password of the user. RC4 is stream cipher; therefore, the sizes of the encrypted and decrypted frames are the same size as the original frame. The Cisco implementation of MPPE is fully interoperable with that of Microsoft and uses all available options, including historyless mode. Historyless mode can increase throughput in lossy environments such as VPNs, because neither side needs to send CCP Resets Requests to synchronize encryption contexts when packets are lost.

MPPE Encryption Types

Two modes of MPPE encryption are offered:

- Stateful MPPE Encryption
- Stateless MPPE Encryption

Stateful MPPE Encryption

Stateful encryption provides the best performance but may be adversely affected by networks that experience substantial packet loss. If you choose stateful encryption, you should also configure flow control to minimize the detrimental effects of this lossiness.

Because of the way that the RC4 tables are reinitialized during stateful synchronization, it is possible that two packets may be encrypted using the same key. For this reason, stateful encryption may not be appropriate for lossy network environments (such as Layer 2 tunnels on the Internet).

Stateless MPPE Encryption

Stateless encryption provides a lower level of performance, but will be more reliable in a lossy network environment.

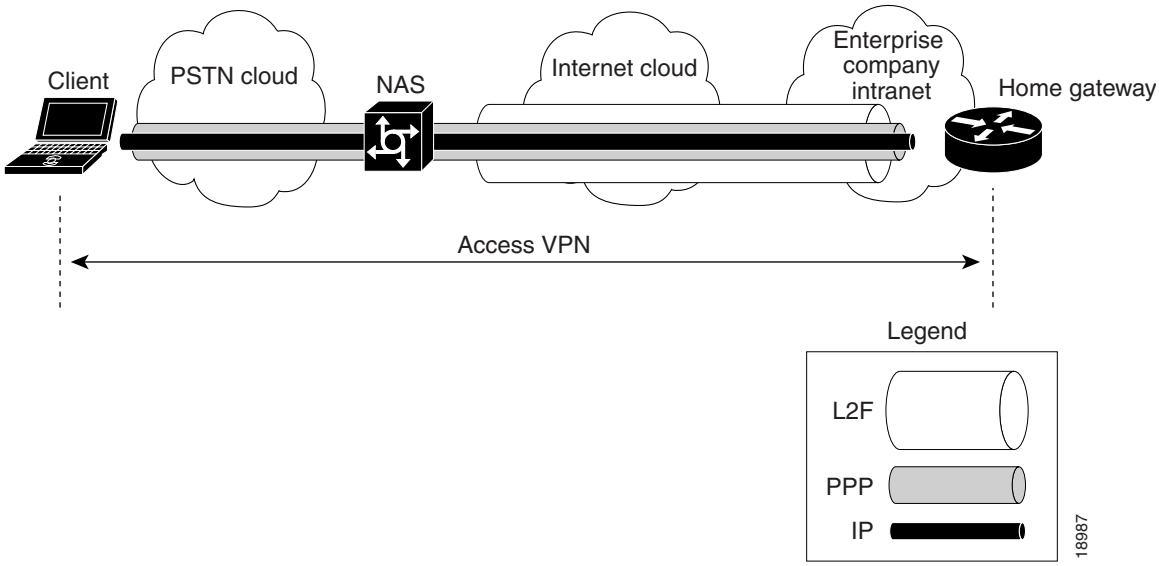
Caution If you choose stateless encryption, you *should not* configure flow control.

L2F Dial-In

VPNs use L2F or L2TP tunnels to tunnel the link layer of high-level protocols (for example, PPP frames or asynchronous High-Level Data Link Control (HDLC)). ISPs configure their NASs to receive calls from users and to forward the calls to the customer tunnel server. Usually, the ISP maintains only information about the tunnel server—the tunnel endpoint. The customer maintains the tunnel server users' IP addresses, routing, and other user database functions. Administration between the ISP and the tunnel server is reduced to IP connectivity.

Figure 71 shows the PPP link that runs between a client (the user hardware and software) and the tunnel server. The NAS and tunnel server establish an L2F tunnel that the NAS uses to forward the PPP link to the tunnel server. The VPN then extends from the client to the tunnel server. The L2F tunnel creates a virtual point-to-point connection between the client and the tunnel server.

Figure 71 End-to-End Access VPN Protocol Flow: L2F, PPP, and IP



The following sections give a functional description of the sequence of events that establish a VPN using L2F as the tunneling protocol:

- Protocol Negotiation Sequence
- L2F Tunnel Authentication Process

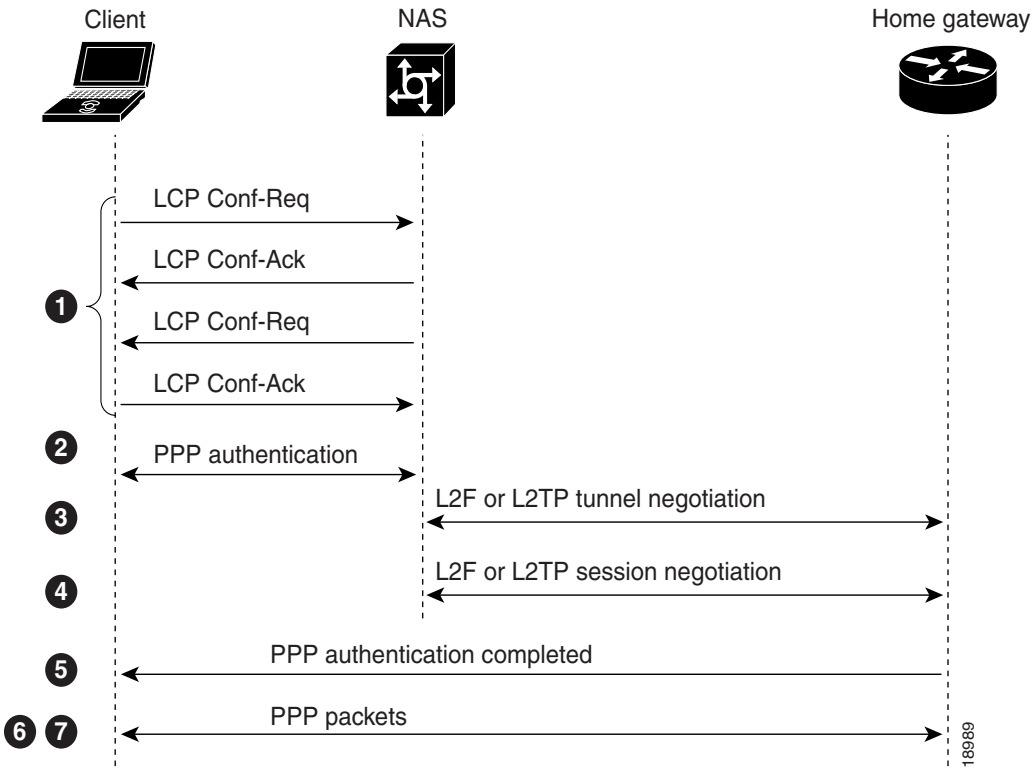
The "Protocol Negotiation Sequence" section provides an overview of the negotiation events that take place as the VPN is established. The "L2F Tunnel Authentication Process" section provides a detailed description of how the NAS and tunnel server establish the L2F tunnel.

Protocol Negotiation Sequence

A user who wants to connect to the customer tunnel server first establishes a PPP connection to the ISP NAS. The NAS then establishes an L2F tunnel with the tunnel server. Finally, the tunnel server authenticates the client username and password and establishes the PPP connection with the client.

Figure 72 shows the sequence of protocol negotiation events between the ISP NAS and the customer tunnel server.

Figure 72 Protocol Negotiation Events Between Access VPN Devices



The following explains the sequence of events shown in Figure 72:

1. The user client and the NAS conduct a standard PPP Link Control Protocol (LCP) negotiation.
2. The NAS begins PPP authentication by sending a Challenge Handshake Authentication Protocol (CHAP) challenge to the client.
3. The client replies with a CHAP response.
4. When the NAS receives the CHAP response, either the phone number that the user dialed in from (when using Dialed Number Information Service-based authentication) or the user domain name (when using authentication based on domain name) matches a configuration on either the NAS or its AAA server.

   This configuration instructs the NAS to create a VPN to forward the PPP session to the tunnel server by using an L2F tunnel.

   Because this is the first L2F session with the tunnel server, the NAS and the tunnel server exchange L2F_CONF packets, which prepare them to create the tunnel. Then they exchange L2F_OPEN packets, which open the L2F tunnel.
5. Once the L2F tunnel is open, the NAS and tunnel server exchange L2F session packets. The NAS sends an L2F_OPEN (Mid) packet to the tunnel server that includes the client information from the LCP negotiation, the CHAP challenge, and the CHAP response.

   The tunnel server forces this information on to a virtual access interface that it has created for the client and responds to the NAS with an L2F_OPEN (Mid) packet.
6. The tunnel server authenticates the CHAP challenge and response (using either local or remote AAA) and sends a CHAP Auth-OK packet to the client. This completes the three-way CHAP authentication.

7. When the client receives the CHAP Auth-OK packet, it can send PPP encapsulated packets to the tunnel server.

   The client and the tunnel server can now exchange I/O PPP encapsulated packets. The NAS acts as a transparent PPP frame forwarder.

   Subsequent PPP incoming sessions (designated for the same tunnel server) do not repeat the L2F tunnel negotiation because the L2F tunnel is already open.

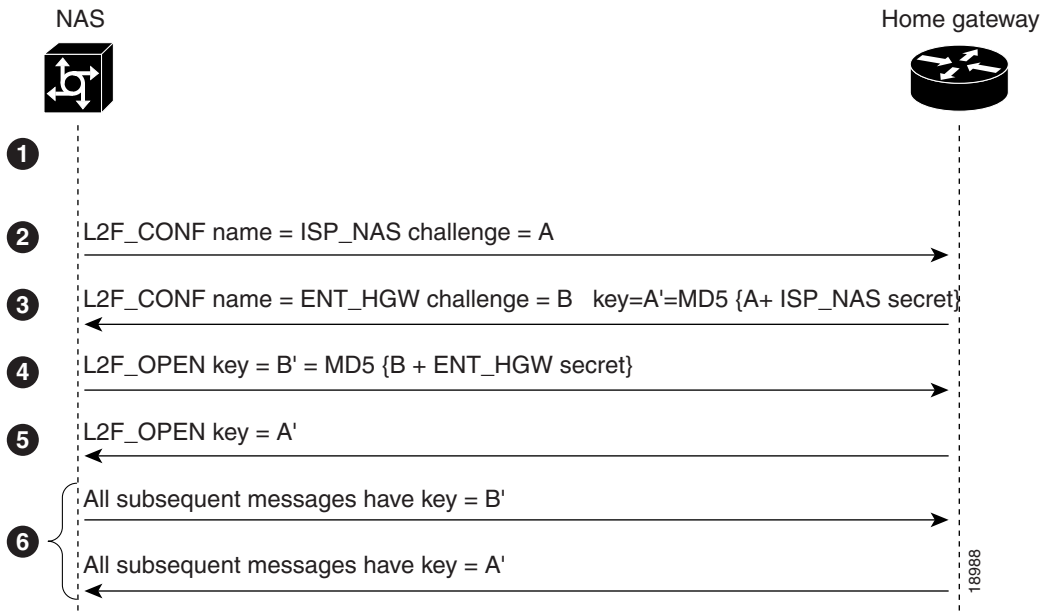
L2F Tunnel Authentication Process

When the NAS receives a call from a client that is to be tunneled to a tunnel server, it first sends a challenge to the tunnel server. The tunnel server then sends a combined challenge and response to the NAS. Finally, the NAS responds to the tunnel server challenge, and the two devices open the L2F tunnel.

Before the NAS and tunnel server can authenticate the tunnel, they must have a common "tunnel secret." A tunnel secret is a common shared secret that is configured on both the NAS and the tunnel server. For more information on tunnel secrets, see the "Configuring VPN Tunnel Authentication Using the L2TP Tunnel Password" section later in this chapter. By combining the tunnel secret with random value algorithms, which are used to encrypt the tunnel secret, the NAS and tunnel server authenticate each other and establish the L2F tunnel.

Figure 73 shows the tunnel authentication process.

Figure 73 L2F Tunnel Authentication Process



The following explains the sequence of events shown in Figure 73:

1. Before the NAS and tunnel server open an L2F tunnel, both devices must have a common tunnel secret in their configurations.
2. The NAS sends an L2F_CONF packet that contains the NAS name and a random challenge value, A.
3. After the tunnel server receives the L2F_CONF packet, it sends an L2F_CONF packet back to the NAS with the tunnel server name and a random challenge value, B. This message also includes a key containing A' (the MD5 of the NAS secret and the value A).
4. When the NAS receives the L2F_CONF packet, it compares the key A' with the MD5 of the NAS secret and the value A. If the key and value match, the NAS sends an L2F_OPEN packet to the tunnel server with a key containing B' (the Message Digest 5 (MD5) of the tunnel server secret and the value B).
5. When the tunnel server receives the L2F_OPEN packet, it compares the key B' with the MD5 of the tunnel server secret and the value B. If the key and value match, the tunnel server sends an L2F_OPEN packet to the NAS with the key A'.
6. All subsequent messages from the NAS include key = B'; all subsequent messages from the tunnel server include key = A'.

Once the tunnel server authenticates the client, the access VPN is established. The L2F tunnel creates a virtual point-to-point connection between the client and the tunnel server. The NAS acts as a transparent packet forwarder.

When subsequent clients dial in to the NAS, the NAS and tunnel server need not repeat the L2F tunnel negotiation because the L2F tunnel is already open.

L2TP Dial-In

L2TP is an emerging Internet Engineering Task Force (IETF) standard that combines the best features of two existing tunneling protocols: Cisco L2F (L2F) and Microsoft Point-to-Point Tunneling Protocol (PPTP).

L2TP offers the same full-range spectrum of features as L2F, but offers additional functionality. An L2TP-capable tunnel server will work with an existing L2F network access server and will concurrently support upgraded components running L2TP. Tunnel servers do not require reconfiguration each time an individual NAS is upgraded from L2F to L2TP. Table 30 offers a comparison of L2F and L2TP feature components.

Table 30 L2F and L2TP Feature Comparison

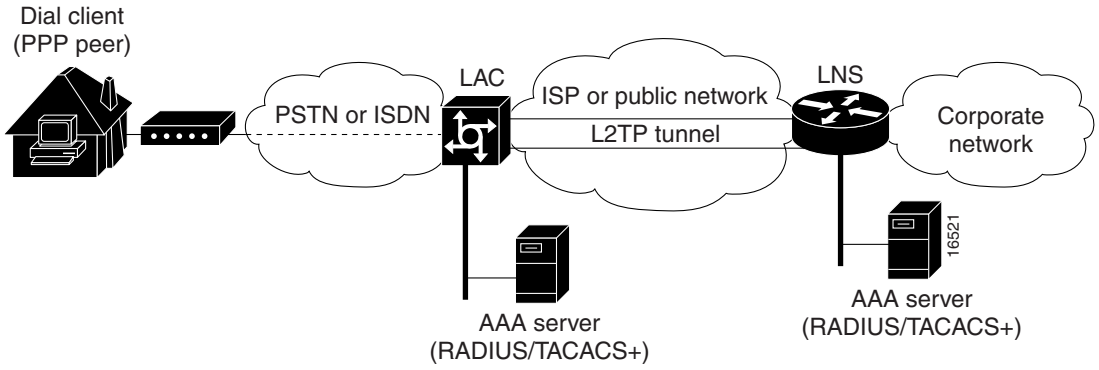
Function	L2F	L2TP
Flow Control	No	Yes
AVP hiding	No	Yes
Tunnel server load sharing	Yes	Yes
Tunnel server stacking/multihop support	Yes	Yes
Tunnel server primary and secondary backup	Yes	Yes
DNS name support	Yes	Yes
Domain name flexibility	Yes	Yes

Table 30 L2F and L2TP Feature Comparison (continued)

Function	L2F	L2TP
Idle and absolute timeout	Yes	Yes
Multilink PPP support	Yes	Yes
Multichassis Multilink PPP support	Yes	Yes
Security	• All security benefits of PPP, including multiple per-user authentication options (CHAP, MS-CHAP, PAP). • Tunnel authentication mandatory.	• All security benefits of PPP, including multiple per-user authentication options (CHAP, MS-CHAP, PAP). • Tunnel authentication optional.

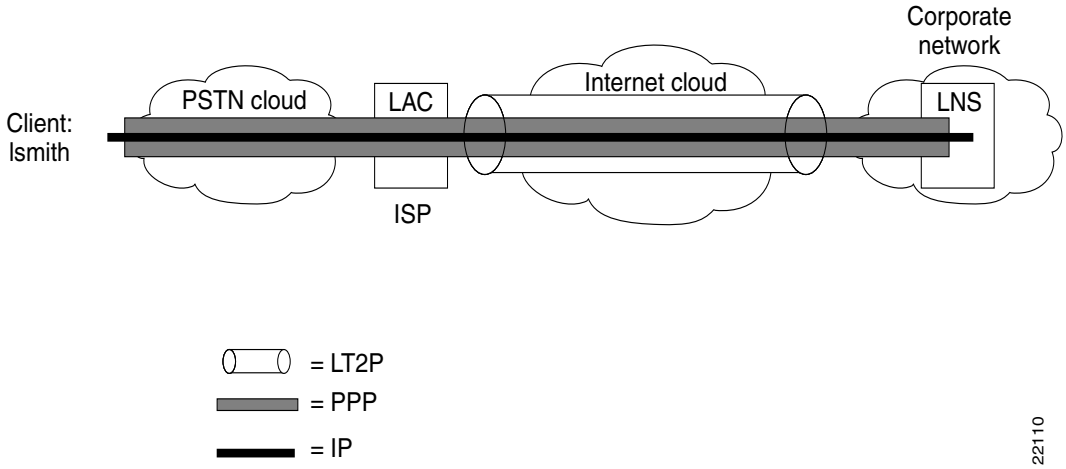
Traditional dialup networking services support only registered IP addresses, which limits the types of applications that are implemented over VPNs. L2TP supports multiple protocols and unregistered and privately administered IP addresses over the Internet. This allows the existing access infrastructure, such as the Internet, modems, access servers, and ISDN terminal adapters (TAs), to be used. It also allows customers to outsource dial-out support, thus reducing overhead for hardware maintenance costs and 800 number fees, and allows them to concentrate corporate gateway resources. Figure 74 shows the L2TP architecture in a typical dialup environment.

Figure 74 L2TP Architecture



The following sections supply additional detail about the interworkings and Cisco implementation of L2TP. Using L2TP tunneling, an Internet service provider (ISP) or other access service can create a virtual tunnel to link customer remote sites or remote users with corporate home networks. The NAS located at the POP of the ISP exchanges PPP messages with remote users and communicates by way of L2TP requests and responses with the customer tunnel server to set up tunnels. L2TP passes protocol-level packets through the virtual tunnel between endpoints of a point-to-point connection. Frames from remote users are accepted by the POP of the ISP, stripped of any linked framing or transparency bytes, encapsulated in L2TP and forwarded over the appropriate tunnel. The customer tunnel server accepts these L2TP frames, strips the L2TP encapsulation, and processes the incoming frames for the appropriate interface. Figure 75 shows the L2TP tunnel detail and how user “lsmith” connects to the tunnel server to access the designated corporate intranet.

Figure 75 L2TP Tunnel Structure



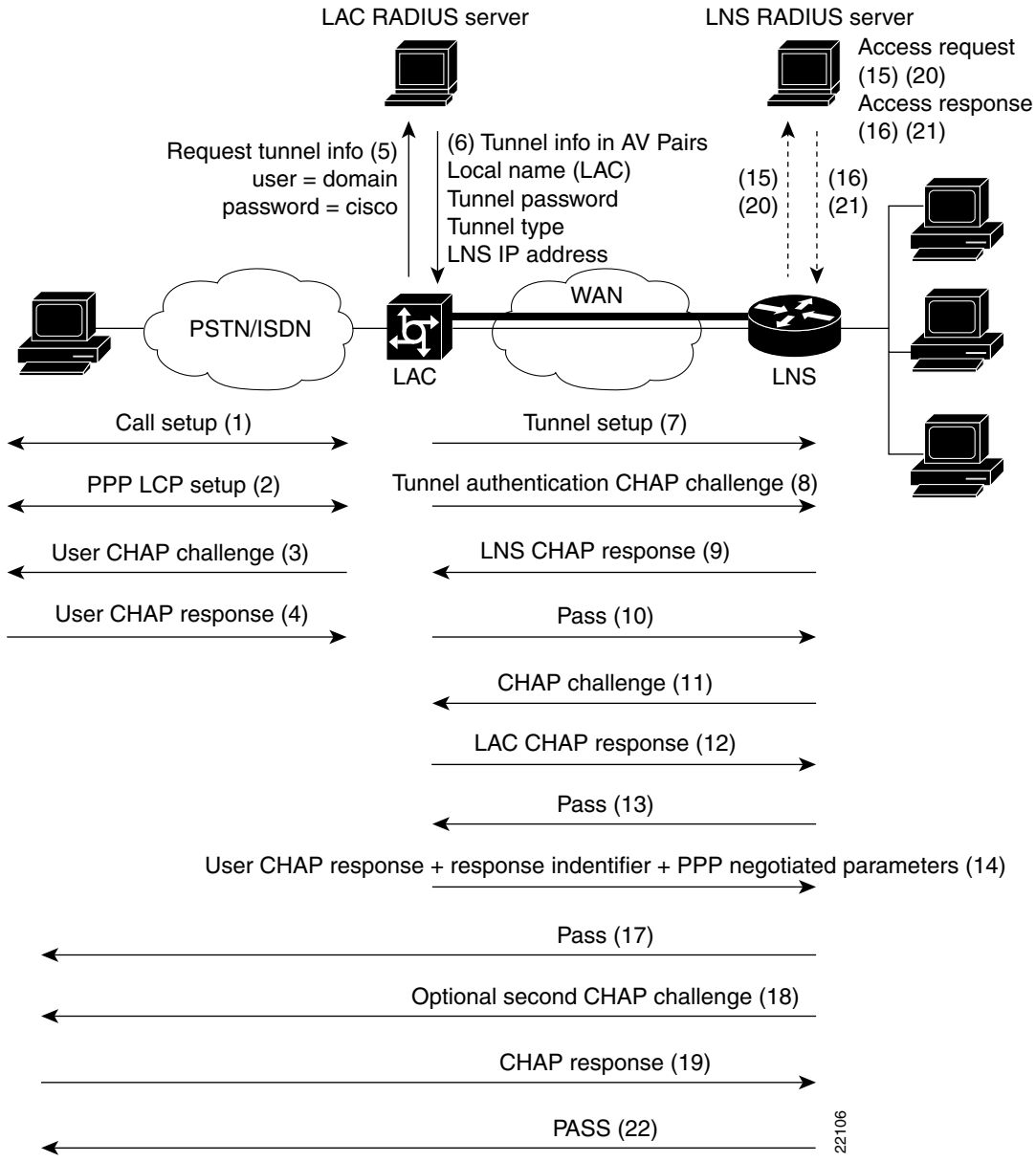
Incoming Call Sequence

The following describes the events required to establish a VPN connection between a remote user, a NAS at the ISP POP, and the tunnel server at the home LAN using an L2TP tunnel:

1. The remote user initiates a PPP connection to the ISP, using the analog telephone system or ISDN.
2. The ISP network NAS accepts the connection at the POP, and the PPP link is established.
3. After the end user and NAS negotiate LCP, the NAS partially authenticates the end user with CHAP or PAP. The username, domain name, or Dialed Number Information Service (DNIS) is used to determine whether the user is a VPN client. If the user is not a VPN client, authentication continues, and the client will access the Internet or other contacted service. If the username is a VPN client, the mapping will name a specific endpoint (the tunnel server).
4. The tunnel endpoints, the NAS, and the tunnel server authenticate each other before any sessions are attempted within a tunnel. Alternatively, the tunnel server can accept tunnel creation without any tunnel authentication of the NAS.
5. Once the tunnel exists, an L2TP session is created for the end user.
6. The NAS will propagate the LCP negotiated options and the partially authenticated CHAP/PAP information to the tunnel server. The tunnel server will funnel the negotiated options and authentication information directly to the virtual access interface. If the options configured on the virtual template interface do not match the negotiated options with the NAS, the connection will fail, and a disconnect will be sent to the NAS.

The result is that the exchange process appears to be between the dialup client and the remote tunnel server exclusively, as if no intermediary device (the NAS) is involved. Figure 76 offers a pictorial account of the L2TP incoming call sequence with its own corresponding sequence numbers. Note that the sequence numbers in Figure 76 are not related to the sequence numbers described in the previous table.

Figure 76 L2TP Incoming Call Flow



VPN Tunnel Authentication Search Order

When a call to a NAS is to be tunneled to a tunnel server, the NAS must identify the tunnel server to which the call is to be forwarded. You can configure the router to authenticate users and also to select the outgoing tunnel on the basis of the following criteria:

- The user domain name
- The DNIS information in the incoming calls
- Both the domain name and the DNIS information

VPN Tunnel Lookup Based on Domain Name

When a NAS is configured to forward VPN calls on the basis of the user domain name, the user must use a username of the form *username@domain*. The NAS then compares the user domain name to the domain names it is configured to search for. When the NAS finds a match, it forwards the user call to the proper tunnel server.

VPN Tunnel Lookup Based on DNIS Information

When a NAS is configured to forward VPN calls on the basis of the user DNIS information, the NAS identifies the user DNIS information, which is provided on ISDN lines, and then forwards the call to the proper tunnel server.

The ability to select a tunnel on the basis of DNIS information provides additional flexibility to network service providers that offer VPN services and to the corporations that use the services. Instead of having to use only the domain name for tunnel selection, tunnel selection can be based on the dialed number.

With this feature, a corporation—which might have only one domain name—can provide multiple specific phone numbers for users to dial in to the NAS at the service provider POP. The service provider can select the tunnel to the appropriate services or portion of the corporate network on the basis of the dialed number.

VPN Tunnel Lookup Based on Both Domain Name and DNIS Information

When a service provider has multiple AAA servers configured, VPN tunnel authorization searches based on domain name can be time consuming and might cause the client session to time out.

To provide more flexibility, service providers can now configure the NAS to perform tunnel authorization searches by domain name only, by DNIS only, or by both in a specified order.

NAS AAA Tunnel Definition Lookup

Authentication, authorization, and accounting (AAA) tunnel definition lookup allows the NAS to look up tunnel definitions using keywords. Two new Cisco AV pairs are added to support NAS tunnel definition lookup: tunnel type and l2tp-tunnel-password. These AV pairs are configured on the RADIUS server. Descriptions of the values are as follows:

- tunnel type—Indicates that the tunnel type is either L2F or L2TP. This is an optional AV pair and if not defined, reverts to L2F, the default value. If you want to configure an L2TP tunnel, you must use the L2TP AV pair value. This command is case sensitive.
- l2tp-tunnel-password—This value is the secret (password) used for L2TP tunnel authentication and L2TP AV pair hiding. This is an optional AV pair value; however, if it is not defined, the secret will default to the password associated with the local name on the NAS local username-password database. This AV pair is analogous to the **l2tp local secret** command.

For example:

```
request dialin l2tp ip 172.21.9.13 domain hoser.com
l2tp local name dustie
l2tp local secret partner
```

is equivalent to the following RADIUS server configuration:

```
acme.com Password = "cisco"
cisco-avpair = "vpdn: tunnel-id=dustie",
```

```
cisco-avpair = "vpdn: tunnel-type=l2tp",
cisco-avpair = "vpdn: l2tp-tunnel-password=partner',
cisco-avpair = "vpdn: ip-addresses=172.21.9.13"
```

Note The password for the domain must be "cisco." This is hard-coded in Cisco IOS software.

L2TP Dial-Out

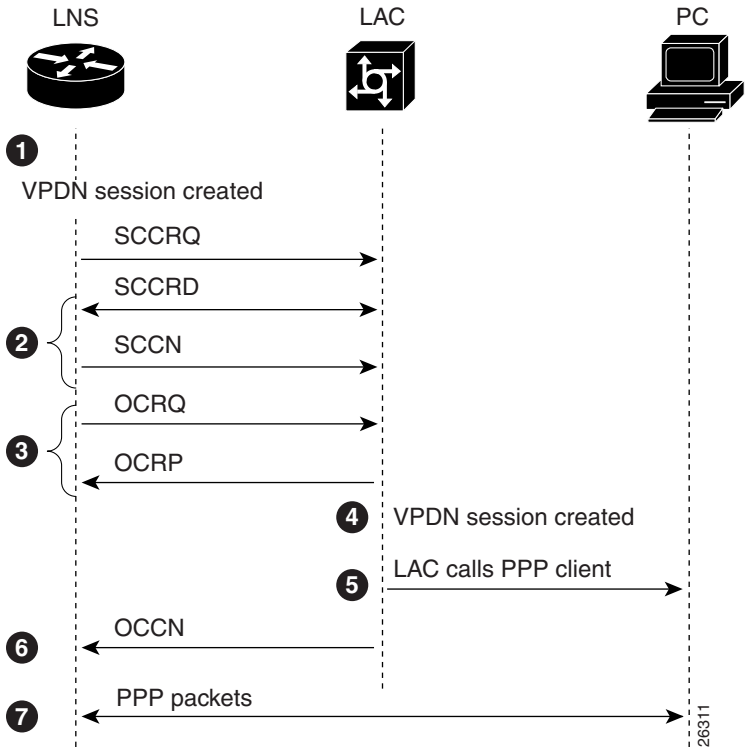
The L2TP dial-out feature enables tunnel servers to tunnel dial-out VPN calls using L2TP as the tunneling protocol. This feature enables a centralized network to efficiently and inexpensively establish a virtual point-to-point connection with any number of remote offices.

Note Cisco routers can carry both dial-in and dial-out calls in the same L2TP tunnels.

L2TP dial-out involves two devices: a tunnel server and a NAS. When the tunnel server wants to perform L2TP dial-out, it negotiates an L2TP tunnel with the NAS. The NAS then places a PPP call to the client(s) that the tunnel server wants to dial out to.

Figure 77 shows a typical L2TP dial-out scenario.

Figure 77 L2TP Dial-Out Process



The following explains the sequence of events described in Figure 77:

1. The tunnel server receives Layer 3 packets, which are to be dialed out, and forwards them to its dialer interface (either a dialer profile or dial-on-demand routing [DDR]).

The dialer issues a dial call request to the VPN group, and the tunnel server creates a virtual access interface. If the dialer is a dialer profile, this interface becomes a member of the dial pool. If the dialer is DDR, the interface becomes a member of the rotary group.

The VPN group creates a VPN session for this connection and sets it in the pending state.

2. The tunnel server and NAS establish an L2TP tunnel (unless a tunnel is already open).
3. The tunnel server sends an Outgoing Call ReQuest (OCRQ) packet to the NAS, which checks if it has a dial resource available.

   If the resource is available, the NAS responds to the tunnel server with an Outgoing Call RePly (OCRP) packet. If the resource is not available, the NAS responds with a Call Disconnect Notification (CDN) packet, and the session is terminated.
4. If the NAS has an available resource, it creates a VPN session and sets it in the pending state.
5. The NAS then initiates a call to the PPP client. When the NAS call connects to the PPP client, the NAS binds the call interface to the appropriate VPN session.
6. The NAS sends an Outgoing Call CoNnected (OCCN) packet to the tunnel server. The tunnel server binds the call to the appropriate VPN session and then brings the virtual access interface up.
7. The dialer on the tunnel server and the PPP client can now exchange PPP packets. The NAS acts as a transparent packet forwarder.

If the dialer interface is a DDR and a virtual profile is configured, the PPP endpoint is the tunnel server virtual-access interface, not the dialer. All Layer 3 routes point to this interface instead of the dialer.

Note Large-scale dial-out, Bandwidth Allocation Protocol (BAP), and Dialer Watch are not supported. All configuration must be local on the router.

VPN Configuration Modes Overview

Cisco VPN is configured using the VPN group configuration mode. VPN groups can now support the following:

- One or both of the following tunnel server VPN subgroup configuration modes
  - Accept-dialin
  - Request-dialout
- One or both of the following NAS VPN subgroup configuration modes
  - Request-dialin
  - Accept-dialout
- One of the four VPN subgroup configuration modes

A VPN group can act as either a tunnel server or a NAS, but not both. But individual routers can have both tunnel server VPN groups and NAS VPN groups.

Table 31 list four VPDN group configuration commands that correspond to the configuration modes listed above. These command modes are accessed from VPN group mode; therefore, they are generically referred to as VPN subgroups.

Table 31 New VPN Group Command Modes

Command	Command Mode Prompt	Type of Service
accept-dialin	`router(config-vpdn-acc-in)#`	tunnel server
request-dialout	`router(config-vpdn-req-ou)#`	tunnel server
request-dialin	`router(config-vpdn-req-in)#`	NAS
accept-dialout	`router(config-vpdn-acc-ou)#`	NAS

The keywords and arguments for the previous **accept-dialin** and **request-dialin** VPDN group configuration commands are now independent commands. The previous syntax is still supported, but when you display the configuration, the commands will appear in the new format.

For example, to configure a NAS to request dial-in, you could use the old command, as follows:

```
request-dialin l2tp ip 10.1.2.3 domain jgb.com
```

However when you view the configuration, the keywords and arguments are displayed in the new format with individual commands:

```
request dialin
 protocol l2tp
 domain jgb.com
initiate-to ip 10.1.2.3
```

Similarly, the **accept-dialout** and **request-dialout** commands have subgroup commands that are used to specify information such as the tunneling protocol and dialer resource.

Table 32 lists the new VPN subgroup commands and which command modes they apply to:

Table 32 VPN Subgroup Commands

Command	VPN Subgroups
default	all subgroups
dialer	accept-dialout
dnis	request-dialin
domain	request-dialin
pool-member	request-dialout
protocol	all subgroups
rotary-group	request-dialout
virtual-template	accept-dialin

The other VPN group commands are dependent on which VPN subgroups exist on the VPN group.

Table 33 lists the VPN group commands and which subgroups you need to enable in order for them to be configurable.

Table 33 VPN Group Commands

Command	VPN Subgroups
accept-dialin	tunnel server VPN group[1]
accept-dialout	NAS VPN group[2]
authen before-forward	request-dialin
default	any subgroup
force-local-chap	accept-dialin
initiate-to	request-dialin or request-dialout
lcp renegotiation	accept-dialin
local name	any subgroup
multilink	request-dialin
request-dialin	NAS VPN Group[2]
request-dialout	tunnel server VPN Group[1]
source-ip	any subgroup
terminate-from	accept-dialin or accept-dialout

1. Tunnel server VPN groups can be configured for accept-dialin and/or request-dialout.
2. NAS VPN groups can be configured for accept-dialout and/or request-dialin.

Prerequisites for VPNs

Before configuring a VPN, you must complete the prerequisites described in Table 34. These prerequisites are discussed in the sections that follow.

Table 34 VPN Prerequisites

Prerequisite	Client-Initiated Dial-In	NAS-Initiated Dial-In	Dial-Out
Configuring the LAN Interface	Required	Required	Required
Configuring AAA	Optional	Required	Required
Specifying the IP Address Pool and BOOTP Servers on the Tunnel Server	Required	Required	N/A
Commissioning the T1 Controllers on the NAS	N/A	Required	N/A
Configuring the Serial Channels for Modem Calls on the NAS	N/A	Required	N/A
Configuring the Modems and Asynchronous Lines on the NAS	N/A	Required	N/A
Configuring the Group-Asynchronous Interface on the NAS	N/A	Required	N/A

Table 34 VPN Prerequisites

Prerequisite	Client-Initiated Dial-In	NAS-Initiated Dial-In	Dial-Out
Configuring the Dialer on a NAS	N/A	N/A	Required
Configuring the Dialer on a Tunnel Server	N/A	N/A	Required

Configuring the LAN Interface

To assign an IP address to the interface that will be carrying the VPN traffic and that brings up the interface, use the following commands on both the NAS and the tunnel server beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# interface interface-type number`	Enters interface configuration mode.
Step 2	`Router(config-if)# ip address ip-address subnet-mask`	Configures the IP address and subnet mask on the interface.
Step 3	`Router(config-if)# no shutdown`	Changes the state of the interface from administratively down to up.

Configuring AAA

To enable AAA, use the following commands on both the NAS and the tunnel server in global configuration mode. If you use RADIUS or TACACS+ for AAA, you also need to point the router to the AAA server using either the **radius-server host** or the **tacacs-server host** command.

Refer to the *Cisco IOS Security Configuration Guide*, Release 12.2, for a complete list of commands and configurable options for security and AAA implementation.

For information on configuring remote AAA servers, refer to the CiscoSecure ACS documentation at: http://www.cisco.com/univercd/cc/td/doc/product/access/acs_soft/index.htm.

	Command	Purpose
Step 1	`Router(config)# aaa new-model`	Enables the AAA access control system.
Step 2	`Router(config)# aaa authentication login default {local | radius | tacacs}`	Enables AAA authentication at login and uses the local username database for authentication.[1]
Step 3	`Router(config)# aaa authentication ppp default {local | radius | tacacs}`	Configures the AAA authentication method that is used for PPP and VPN connections.[1]
Step 4	`Router(config)# aaa authorization network default {local | radius | tacacs}`	Configures the AAA authorization method that is used for network-related service requests.[1]
Step 5	`Router(config)# aaa accounting network default start-stop {radius | tacacs}`	(Optional) Enables AAA accounting that sends a stop accounting notice at the end of the requested user process.[1]

	Command	Purpose
Step 6	`Router(config)# vpdn aaa override-server {aaa-server-ip-address | aaa-server-name}`	(Optional) Specifies the AAA servers to be used for VPDN tunnel authorization. If this command is not configured, the default AAA server configured for network authorization is used for VPDN authorization.
Step 7	`Router(config)# vpdn aaa attribute [{nas-ip-address vpdn-nas}| (nas-port vpdn-nas}]`	(Optional) Enables the reporting of AAA attributes from the HGW to the configured RADIUS or TACACS+ AAA server. This command is applicable only on the tunnel server and is disabled by default.
Step 8	`Router(config)# vpdn aaa untagged`	(Optional) Enables the application of untagged attribute values to all attribute sets for VPDN tunnels, unless a value for that attribute is already specified in the attribute set. This command is enabled by default, therefore configuration of this command is required only if the command has been previously disabled.
Step 9	`Router(config)# radius-server host ip-address [auth-port number] [acct-port number]`	Specifies the RADIUS server IP address and optionally the ports to be used for authentication and accounting requests.
	`Router(config)# radius-server key cisco`	Sets the authentication key and encryption key for all RADIUS communication. **Note** The RADIUS key must be "cisco." This is hard-coded in Cisco IOS software.
	or `Router(config)# tacacs-server host ip-address [port integer] [key string]`	Specifies the TACACS+ server IP address and optionally the port to be used, and an authentication and encryption key.

1. If you specify more than one method, AAA will query the servers or databases in the order that they are entered.

Specifying the IP Address Pool and BOOTP Servers on the Tunnel Server

To specify the IP addresses and the BOOTP servers that will be assigned to VPN clients, use the following commands on the tunnel server in global configuration mode.

The IP address pool is the addresses that the tunnel server assigns to clients. You must configure an IP address pool. You can also provide BOOTP servers. Domain Name System (DNS) servers translate host names to IP addresses. WINS servers, which are specified using the **async-bootp nbns-server** command, provide dynamic NetBIOS names that Windows devices use to communicate without IP addresses.

	Command	Purpose
Step 1	`HGW(config)# ip local pool default first-ip-address last-ip-address`	Configures the default local pool of IP address that will be used by clients.
Step 2	`HGW(config)# async-bootp dns-server ip-address1 [additional-ip-address]`	(Optional) Returns the configured addresses of DNS in response to BOOTP requests.
Step 3	`HGW(config)# async-bootp nbns-server ip-address1 [additional-ip-address]`	(Optional) Returns the configured addresses of Windows NT servers in response to BOOTP requests.

Commissioning the T1 Controllers on the NAS

To define the ISDN switch type and commission the T1 controllers to allow modem calls to come into the NAS, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`NAS(config)# isdn switch-type switch-type`	Enters the telco switch type. An ISDN switch type that is specified in global configuration mode is automatically propagated into the individual serial interfaces (for example, serial interface 0:23, 1:23, 2:23, and 3:23).
Step 2	`NAS(config)# controller t1 0`	Accesses controller configuration mode for the first T1 controller, which is number 0. The controller ports are numbered 0 through 3 on the quad T1/PRI card.
Step 3	`NAS(config-controller)# framing framing-type`	Enters the T1 framing type.
Step 4	`NAS(config-controller)# linecode linecode`	Enters the T1 line-code type.

	Command	Purpose
Step 5	`NAS(config-controller)# clock source line primary`	Configures the access server to get its primary clocking from the T1 line assigned to controller 0. Line clocking comes from the remote switch.
Step 6	`NAS(config-controller)# pri-group timeslots` *range*	Assigns the T1 time slots as ISDN PRI channels. After you enter this command, a D-channel serial interface is instantly created (for example, S0:23), along with individual B-channel serial interfaces (S0:0, S0:1, and so on). The D-channel interface functions like a dialer for the B channels using the controller. If this was an E1 interface, the PRI group range would be 1 to 31. The D-channel serial interfaces would be S0:15, S1:15, S2:15, and S3:15.

Configuring the Serial Channels for Modem Calls on the NAS

To configure the D channels (the signaling channels) to allow incoming voice calls to be routed to the integrated MICA technologies modems and to control the behavior of the individual B channels, use the following commands on the NAS beginning in global configuration mode:

	Command	Purpose
Step 1	`NAS(config)# interface serial 0:23`	Accesses configuration mode for the D-channel serial interface that corresponds to controller T1 0. The behavior of serial 0:0 through serial 0:22 is controlled by the configuration instructions provided for serial 0:23. This concept is also true for the other remaining D-channel configurations.
Step 2	`NAS(config-if)# isdn incoming-voice modem`	Enables analog modem voice calls that come in through the B channels to be connected to the integrated modems.
Step 3	`NAS(config-if)# exit`	Returns to global configuration mode.

Step 4	`NAS(config)# interface serial 1:23`
	`NAS(config-if)# isdn incoming-voice modem`
	`NAS(config-if)# exit`
	`NAS(config)# interface serial 2:23`
	`NAS(config-if)# isdn incoming-voice modem`
	`NAS(config-if)# exit`
	`NAS(config)# interface serial 3:23`
	`NAS(config-if)# isdn incoming-voice modem`
	`NAS(config-if)# exit`
		Configures the three remaining D channels with the same ISDN incoming-voice modem setting.

Configuring the Modems and Asynchronous Lines on the NAS

To define a range of modem lines and to enable PPP clients to dial in, bypass the EXEC facility, and automatically start PPP, use the following commands on the NAS beginning in global configuration mode.

Configure the modems and lines after the ISDN channels are operational. Each modem corresponds with a dedicated asynchronous line inside the NAS. The modem speed of 115200 bps and hardware flow control are default values for integrated modems.

	Command	Purpose
Step 1	`NAS(config)# line line-number [ending-line-number]`	Enters the modem line or range of modem lines (by entering an *ending-line-number*) that you want to configure.
Step 2	`NAS(config-line)# autoselect ppp`	Enables PPP clients to dial in, bypass the EXEC facility, and automatically start PPP on the lines.
Step 3	`NAS(config-line)# autoselect during-login`	Displays the username:password prompt as the modems connect. **Note** These two autoselect commands enable EXEC (shell) and PPP services on the same lines.
Step 4	`NAS(config-line)# modem inout`	Supports incoming and outgoing modem calls.

Configuring the Group-Asynchronous Interface on the NAS

To create a group-asynchronous interface and project protocol characteristics to the asynchronous interfaces, use the following commands on the NAS beginning in global configuration mode.

The group-async interface is a template that controls the configuration of the specified asynchronous interfaces inside the NAS. Asynchronous interfaces are lines running in PPP mode. An asynchronous interface uses the same number as its corresponding line. Configuring all the asynchronous interfaces as an asynchronous group saves you time by reducing the number of configuration steps.

	Command	Purpose
Step 1	`NAS(config)# interface group-async number`	Creates the group-asynchronous interface.
Step 2	`NAS(config-if)# ip unnumbered interface-type number`	Uses the IP address defined on the specified interface.
Step 3	`NAS(config-if)# encapsulation ppp`	Enables PPP.
Step 4	`NAS(config-if)# async mode interactive`	Configures interactive mode on the asynchronous interfaces. Interactive mode means that clients can dial in to the NAS and get a router prompt or PPP session. Dedicated mode means that only PPP sessions can be established on the NAS. Clients cannot dial in and get an EXEC (shell) session.

	Command	Purpose
Step 5	`NAS(config-if)# ppp authentication {chap | pap | chap pap | pap chap}`	Configures the authentication to be used on the interface during LCP negotiation. When both authentication methods are specified, the NAS first authenticates with the first method entered. If the first method is rejected by the client, the second authentication method is used.
Step 6	`NAS(config-if)# group-range range`	Specifies the range of asynchronous interfaces to include in the group, which is usually equal to the number of modems in the access server.

Configuring the Dialer on a NAS

To configure the dialer on a NAS for L2TP dial-out, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`NAS(config)# interface dialer number`	Defines a dialer rotary group.
Step 2	`NAS(config-if)# ip unnumbered interface-type number`	Configures the dialer to use the interface IP address.
Step 3	`NAS(config-if)# encapsulation ppp`	Enables PPP encapsulation.
Step 4	`NAS(config-if)# dialer in-band`	Enables DDR on the dialer.
Step 5	`NAS(config-if)# dialer aaa`	Enables the dialer to use the AAA server to locate profiles for dialing information.
Step 6	`NAS(config-if)# dialer-group group-number`	Assigns the dialer to the specified dialer group.
Step 7	`NAS(config-if)# ppp authentication chap`	Specifies that CHAP authentication will be used.

Configuring the Dialer on a Tunnel Server

To configure the dialer on an a tunnel server for L2TP dial-out, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`LNS(config)# interface dialer number`	Defines a dialer rotary group.
Step 2	`LNS(config-if)# ip address ip-address subnet-mask`	Specifies an IP address for the group.
Step 3	`LNS(config-if)# encapsulation ppp`	Enables PPP encapsulation.
Step 4	`LNS(config-if)# dialer remote-name peer-name`	Specifies the name used to authenticate the remote router that is being dialed.
Step 5	`LNS(config-if)# dialer string dialer-number`	Specifies the number that is dialed.
Step 6	`LNS(config-if)# dialer vpdn`	Enables dial-out.
Step 7	`LNS(config-if)# dialer pool pool-number`	Specifies the dialer pool.

	Command	Purpose
Step 8	`LNS(config-if)# dialer-group group-number`	Assigns the dialer to the specified dialer group.
Step 9	`LNS(config-if)# ppp authentication chap`	Specifies that CHAP authentication will be used.

How to Configure a VPN

Configuration for both dial-in and dial-out VPNs is described in the following sections:

- Enabling a VPN
- Configuring VPN Tunnel Authentication Using the Host Name or Local Name
- Configuring VPN Tunnel Authentication Using the L2TP Tunnel Password
- Configuring Client-Initiated Dial-In VPN
- Configuring NAS-Initiated Dial-In VPN
- Configuring Dial-Out VPN
- Configuring Advanced VPN Features

See the section "Configuration Examples for VPN" later in this chapter for examples of how you can implement VPN in your network.

Enabling a VPN

To enable a VPN tunnel, use the following command in global configuration mode:

Command	Purpose
`Router(config)# vpdn[1] enable`	Enables VPN.

1. The Cisco IOS command syntax uses the more specific term VPDN (virtual private dialup network) instead of VPN.

To disable a VPN tunnel, use the **clear vpdn tunnel** command in EXEC mode. The **no vpdn enable** command does not automatically disable a VPN tunnel.

Configuring VPN Tunnel Authentication Configuration

VPN tunnel authentication enables routers to authenticate the other tunnel endpoint before establishing a VPN tunnel. It is required for L2F tunnels and optional for L2TP tunnels.

Disabling VPN Tunnel Authentication for L2TP Tunnels

To disable VPN tunnel authentication for L2TP tunnels, use the following commands beginning in global configuration mode:

Command	Purpose
`ISP_NAS(config)# vpdn-group group` `ISP_NAS(config-vpdn)# no l2tp tunnel authentication`	Disables VPN tunnel authentication for the specified VPN group. The VPN group will not challenge any router that attempts to open an L2TP tunnel.

Note Before you can configure any **l2tp** VPN group command, you must specify L2TP as the protocol for a VPN subgroup within the VPN group. For more information, see the "Configuring NAS-Initiated Dial-In VPN" and "Configuring Dial-Out VPN" sections later in this chapter.

VPN tunnel authentication can be performed in the following ways:

- Using local AAA on both the NAS and the tunnel server
- Using RADIUS on the NAS and local AAA on the tunnel server
- Using TACACS+ on the NAS and local AAA on the tunnel server

This section discusses local tunnel authentication. For information on RADIUS and TACACS+, refer to the "NAS AAA Tunnel Definition Lookup" section earlier in this chapter and the *Cisco IOS Security Configuration Guide*, Release 12.2.

VPN tunnel authentication requires that a single shared secret—called the *tunnel secret*—be configured on both the NAS and tunnel server. There are two methods for configuring the tunnel secret:

- Configuring VPN Tunnel Authentication Using the Host Name or Local Name

  The tunnel secret is configured as a password by using the **username** command.
- Configuring VPN Tunnel Authentication Using the L2TP Tunnel Password

  The tunnel secret is configured by using the **l2tp tunnel password** command.

Configuring VPN Tunnel Authentication Using the Host Name or Local Name

To configure VPN tunnel authentication using the **hostname** or **local name** commands, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`ISP_NAS(config)# hostname host-name`	Configures the router host name. By default, the router uses the host name as the tunnel name in VPN tunnel authentication.

	or
	`ISP_NAS(config)# vpdn-group group`
	`ISP_NAS(config-vpdn)# local name tunnel-name`
		or
		(Optional) Configures the local name for the VPN group. When negotiating VPN tunnel authentication for this VPN group, the router will use the local name as the tunnel name.

Step 2	`ISP_NAS(config)# username tunnel-name password tunnel-secret`	Configures the other router's tunnel name and the tunnel secret as a user name and password combination. **Note** The tunnel secret must be the same on both routers. Each router must have the other router's tunnel name (specified by either the **hostname** or **local name** command) configured as a username with the tunnel secret as the password.

Configuring VPN Tunnel Authentication Using the L2TP Tunnel Password

To configure VPN tunnel authentication using the **l2tp tunnel password** command, use the following commands beginning in global configuration:

	Command	Purpose
Step 1	`ISP_NAS(config)# vpdn-group group` `ISP_NAS(config-vpdn)# l2tp tunnel password tunnel-secret`	Configures the tunnel secret that will be used for VPN tunnel authentication for this VPN group and enters VPDN configuration mode.
Step 2	`ISP_NAS(config-vpdn)# local name tunnel-name` `ISP_NAS(config-vpdn)# exit`	(Optional) Configures the tunnel name of the router.
	`ISP_NAS(config)# username tunnel-name password tunnel-secret`	(Optional) Configures the other router's tunnel name and the tunnel secret as a user name. If the other router uses the **l2tp tunnel password** command to configure the tunnel secret, these commands are not necessary. **Note** The tunnel secret must be the same on both routers.

For sample VPN tunnel authentication configurations, see the "VPN Tunnel Authentication Examples" section later in this chapter.

Configuring Client-Initiated Dial-In VPN

For client-initiated dial-in VPNs, complete the following tasks:

- Configuring a Tunnel Server to Accept Dial-In (Required)
- Configuring MPPE on the ISA Card (Optional)
- Tuning PPTP (Optional)

When configuring PPTP and MPPE, you should consider the following restrictions:

- Only Cisco Express Forwarding (CEF) and process switching are supported. Regular fast switching is not supported.
- PPTP does not support multilink.
- VPDN multihop is not supported.
- Because all PPTP signaling is over TCP, TCP configurations will affect PPTP performance in large-scale environments.
- MPPE is not supported with TACACS.
- Windows clients must use MS-CHAP authentication in order for MPPE to work.
- If you are performing mutual authentication with MS-CHAP and MPPE, both sides of the tunnel must use the same password.
- To use MPPE with AAA, you must use a RADIUS server that supports the Microsoft Vendor specific attribute for MPPE-KEYS. CiscoSecure NT supports MPPE beginning with release 2.6. CiscoSecure UNIX does not support MPPE.

Configuring a Tunnel Server to Accept PPTP Tunnels

To configure a tunnel to accept tunneled PPP connections from a client, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`PNS(config)# vpdn-group 1`	Creates vpdn group 1.
Step 2	PNS(config-vpdn)# **accept-dialin**	Enables the tunnel server to accept dial-in requests.
Step 3	PNS(config-vpdn-acc-in)# **protocol pptp**	Specifies that the tunneling protocol will be PPTP.
Step 4	PNS(config-vpdn-acc-in)# **virtual-template** *template-number*	Specifies the number of the virtual template that will be used to clone the virtual-access interface.
Step 5	PNS(config-vpdn-acc-in)# **exit**	Exit to higher command mode.
Step 6	PNS(config-vpdn)# **local name** *localname*	(Optional) Specifies that the tunnel server will identify itself with this local name. If no local name is specified, the tunnel server will identify itself with its host name.

Configuring MPPE on the ISA Card

To offload MPPE encryption from the tunnel server processor to the ISA card, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`PNS(config)# controller isa slot/port`	Enters controller configuration mode on the ISA card.
Step 2	PNS(config-controller)# **encryption mppe**	Enables MPPE encryption

Tuning PPTP

To tune PPTP, use one or more of the following commands in VPDN configuration mode:

Command	Purpose
`PNS(config-vpdn)# pptp flow-control receive-window packets`	Specifies how many packets the client can send before it must wait for the acknowledgment from the tunnel server.
`PNS(config-vpdn)# pptp flow-control static-rtt milliseconds`	Specifies the timeout interval of the tunnel server between sending a packet to the client and receiving a response.
`PNS(config-vpdn)# pptp tunnel echo seconds`	Specifies the period of idle time on the tunnel that will trigger an echo message from the tunnel server to the client.

Configuring NAS-Initiated Dial-In VPN

The following tasks must be completed for NAS-initiated dial-in VPNs:

- Configuring a NAS to Request Dial-In (Required)
- Configuring a Tunnel Server to Accept Dial-In (Required)
- Creating the Virtual Template on the Network Server (Required)

Configuring a NAS to Request Dial-In

The NAS is a device that is typically (although not always) located at a service provider POP; initial configuration and ongoing management are done by the service provider.

To configure a NAS to accept PPP calls and tunnel them to a tunnel server, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`NAS(config)# vpdn-group 1`	Creates VPN group 1.
Step 2	`NAS(config-vpdn)# request-dialin`	Enables the NAS to request L2F or L2TP dial-in requests.
Step 3	`NAS(config-vpdn-req-in)# protocol [l2f | l2tp | any]`	Specifies which tunneling protocol is to be used.

	Command	Purpose

Step 4	`NAS(config-vpdn-req-in)# domain domain-name`
	
	or
	
	`NAS(config-vpdn-req-in)# dnis dnis-number`
		Specifies the domain name of the users that are to be tunneled.
		
		Specifies the DNIS number of users that are to be tunneled.
		
		You can configure multiple domain names and/or DNIS numbers for an individual request-dialin subgroup.

Step 5	`NAS(config-vpdn-req-in)# exit` `NAS(config-vpdn)# initiate-to ip ip-address`	Specifies the IP address that the NAS will establish the tunnel with. This is the IP address of the tunnel server.
Step 6	`NAS(config-vpdn)# vpdn search-order {domain | dnis | domain dnis | dnis domain}`	(Optional) Specifies the method that is used to determine if a dial-in call should be tunneled. If both keywords are entered, the NAS will search the criteria in the order they are entered.

Configuring a Tunnel Server to Accept Dial-In

To configure a tunnel server to accept tunneled PPP connections from a NAS, use the following commands beginning in global configuration mode.

The tunnel server is the termination point for a VPN tunnel. The tunnel server initiates outgoing calls to and receives incoming calls from the NAS.

	Command	Purpose
Step 1	`LNS(config)# vpdn-group 1`	Creates VPN group 1.
Step 2	`LNS(config-vpdn)# accept-dialin`	Enables the tunnel server to accept dial-in requests.
Step 3	`LNS(config-vpdn-acc-in)# protocol [l2f | l2tp | any]`	Specifies which tunneling protocol is to be used.
Step 4	`LNS(config-vpdn-acc-in)# virtual-template number`	Specifies the number of the virtual template that will be used to clone the virtual access interface.
Step 5	`LNS(config-vpdn-acc-in)# exit` `LNS(config-vpdn)# terminate-from hostname hostname`	Accepts tunnels that have this host name configured as a local name.

See the section "Tunnel Server Comprehensive Dial-in Configuration Example" later in this chapter for a configuration example.

Creating the Virtual Template on the Network Server

At this point, you can configure the virtual template interface with configuration parameters you want applied to virtual access interfaces. A virtual template interface is a logical entity configured for a serial interface. The virtual template interface is not tied to any physical interface and is applied dynamically, as needed. Virtual access interfaces are *cloned* from a virtual template interface, used on demand, and then freed when no longer needed.

To create and configure a virtual template interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`HGW(config)# interface virtual-template number`	Create the virtual template that is used to clone virtual access interfaces.
Step 2	`HGW(config-if)# ip unnumbered interface-type number`	Specifies that the virtual access interfaces use the specified interface IP address.
Step 3	`HGW(config-if)# ppp authentication {chap | pap | chap pap | pap chap}`	Enables CHAP authentication using the local username database.
Step 4	`HGW(config-if)# peer default ip address pool pool`	Returns an IP address from the default pool to the client.
Step 5	`HGW(config-if)# encapsulation ppp`	Enables PPP encapsulation.

Optionally, you can configure other commands for the virtual template interface. For more information about configuring virtual template interfaces, refer to the "Configuring Virtual Template Interfaces" chapter in this publication.

Configuring Dial-Out VPN

The following tasks must be completed for dial-out VPNs:

- Configuring a Tunnel Server to Request Dial-Out (Required)
- Configuring a NAS to Accept Dial-Out (Required)

Configuring a Tunnel Server to Request Dial-Out

To configure a tunnel server to request dial-out tunneled PPP connections to a NAS, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`LNS(config)# vpdn-group 1`	Creates VPN group 1.
Step 2	`LNS(config-vpdn)# request-dialout`	Enables the tunnel server to send L2TP dial-out requests.
Step 3	`LNS(config-vpdn-req-ou)# protocol l2tp`	Specifies L2TP as the tunneling protocol. **Note** L2TP is the only protocol that supports dial-out.

Step 4	`LNS(config-vpdn-req-ou)# pool-member pool-number`
	or
	`LNS(config-vpdn-req-ou)# rotary-group group-number`
		Specifies the dialer profile pool that will be used to dial out.
		Specifies the dialer rotary group that will be used to dial out.
		You can configure only one dialer profile pool or dialer rotary group. Attempting to configure a second dialer resource will remove the first from the configuration.

	Command	Purpose
Step 5	`LNS(config-vpdn-req-ou)# exit` `LNS(config-vpdn)# initiate-to ip ip-address`	Specifies the IP address that will be dialed out. This is the IP address of the NAS.
Step 6	`LNS(config-vpdn)# local name hostname`	Specifies that the L2TP tunnel will identify itself with this host name.

Configuring a NAS to Accept Dial-Out

To configure a NAS to accept tunneled dial-out connections from a tunnel server, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`NAS(config)# vpdn-group 1`	Creates VPN group 1.
Step 2	`NAS(config-vpdn)# accept-dialout`	Enables the NAS to accept L2TP dial-out requests.
Step 3	`NAS(config-vpdn-acc-ou)# protocol l2tp`	Specifies L2TP as the tunneling protocol. **Note** L2TP is the only protocol that supports dial-out.
Step 4	`NAS(config-vpdn-acc-ou)# dialer dialer-interface`	Specifies the dialer that is used to dial out to the client.
Step 5	`NAS(config-vpdn-acc-ou)# exit` `NAS(config-vpdn)# terminate-from hostname hostname`	Accepts L2TP tunnels that have this host name configured as a local name.

Configuring Advanced VPN Features

The following optional tasks provide advanced VPN features:

- Configuring Advanced Remote AAA Features
- Configuring Per-User VPN
- Configuring Preservation of IP ToS Field
- Shutting Down a VPN Tunnel
- Limiting the Number of Allowed Simultaneous VPN Sessions
- Enabling Soft Shutdown of VPN Tunnels
- Configuring Event Logging
- Setting the History Table Size

Configuring Advanced Remote AAA Features

This section describes the following two advanced remote AAA features for VPNs:

- Tunnel Server Load Balancing on the NAS AAA Server
- DNS Name Support

Tunnel Server Load Balancing on the NAS AAA Server

NAS AAA servers can forward users of the same domain name or DNIS to more than one tunnel server. The NAS AAA server can be configured to balance the load of calls equally among the tunnel servers, or it can designate different priority levels to the tunnel servers.

To configure load balancing on a NAS RADIUS server, configure multiple IP addresses in the vpdn:ip-addresses attribute value (AV) pair. The IP addresses can be separated by either spaces or by commas. The following example shows a profile that will equally balance the load between three tunnel servers.

```
user = terrapin.com{
profile_id = 29
profile_cycle = 7
radius=Cisco {
check_items= {
2=cisco
}
reply_attributes= {
9,1="vpdn:l2tp-tunnel-password=cisco123"
9,1="vpdn:tunnel-type=l2tp"
9,1="vpdn:ip-addresses=172.16.171.11 172.16.171.12 172.16.171.13"
9,1="vpdn:tunnel-id=tunnel"
}
}
}
```

To specify different priorities for the tunnel servers, separate the IP addresses with a slash. The following AV pair instructs the RADIUS server to equally balance calls between 172.16.171.11 and 172.16.171.12. If both of those tunnel servers are unavailable, the RADIUS server will tunnel calls to 172.16.171.13.

```
9,1="vpdn:ip-addresses=172.16.171.11 172.16.171.12/172.16.171.13"
```

DNS Name Support

NAS AAA servers can resolve DNS names and translate them into IP addresses. The server will first look up the name in its name cache. If the name is not in the name cache, the server will resolve the name by using a DNS server. The following AV pair instructs the RADIUS server to resolve the DNS name "terrapin" and tunnel calls to the appropriate IP addresses:

```
9,1="vpdn:ip-addresses=terrapin"
```

For detailed information about remote AAA configuration, refer to the CiscoSecure ACS documentation at http://www.cisco.com/univercd/cc/td/doc/product/access/acs_soft/index.htm.

Configuring Per-User VPN

In a VPN that uses remote AAA, when a user dials in, the access server that receives the call forwards information about the user to its remote AAA server. With basic VPN, the access server sends only the user domain name (when performing authentication based on domain name) or the telephone number the user dialed in from (when performing authentication based on DNIS).

Per-user VPN configuration sends the entire structured username to the AAA server the first time the router contacts the AAA server. This enables Cisco IOS software to customize tunnel attributes for individual users who use a common domain name or DNIS.

Without VPN per-user configuration, Cisco IOS software sends only the domain name or DNIS to determine VPN tunnel attribute information. Then, if no VPN tunnel attributes are returned, Cisco IOS software sends the entire username string.

Note Per-user VPN configuration supports only RADIUS as the AAA protocol.

To configure per-user VPN, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# vpdn-group group-number`	Enters VPN group configuration mode.
Step 2	`Router(config-vpdn)# authen before-forward`	Specifies that the entire structured username be sent to the AAA server the first time the router contacts the AAA server.

Configuring Preservation of IP ToS Field

When L2TP data packets are created, they have a type of service (ToS) field of zero, which indicates normal service. This ignores the ToS field of the encapsulated IP packets that are being tunneled.

To preserve quality of service (QoS) for tunneled packets by copying the ToS field of the IP packets' onto the L2TP data packets when they are created at the tunnel server virtual access interface, use the following commands beginning in global configuration mode:

	Command	Purpose
Step 1	`LNS(config)# vpdn-group 1`	Creates VPN group 1.

Step 2	`LNS(config-vpdn)# accept-dialin`
	or
	`LNS(config-vpdn)# request-dialout`
		Enables the tunnel server to accept dial-in requests.
		
		Enables the tunnel server to send L2TP dial-out requests.

Step 3	`LNS(config-vpdn-acc-in)# protocol l2tp`
	or
	`LNS(config-vpdn-req-ou)# protocol l2tp`
		Specifies L2TP as the tunneling protocol.
		Note L2TP is the only protocol that supports dial-out and IP ToS preservation.

Step 4	`LNS(config-vpdn-req-ou)# exit`	Returns to VPDN group configuration mode.
Step 5	`LNS(config-vpdn)# ip tos reflect`	Preserves the ToS field of the encapsulated IP packets.

Note The tunneled link must carry IP for the ToS field to be preserved. The encapsulated payload of Multilink PPP (MLP) connections is not IP, therefore this task has no effect when MLP is tunneled.

Note Proxy PPP dial-in is not supported.

Shutting Down a VPN Tunnel

To shut down a VPN tunnel, use the following command in privileged EXEC mode:

Command	Purpose
`Router# clear vpdn tunnel {l2f nas-name hgw-name | l2tp [remote-name] [local-name]}`	Shuts down a specific tunnel and all the sessions within the tunnel.

Limiting the Number of Allowed Simultaneous VPN Sessions

To set a limit for the maximum number of allowed simultaneous VPN sessions, use the following command in global configuration mode:

Command	Purpose
`Router(config)# vpdn session-limit sessions`	Limits the number of simultaneous VPN sessions on the router to the number specified with the *sessions* argument.

To verify that the **vpdn session-limit** command is working properly, perform the following steps:

Note If you use a Telnet session to connect to the NAS, enable the **terminal monitor** command, which ensures that your EXEC session is receiving the logging and debug output from the NAS.

Step 1 Enter the **vpdn session-limit 1** global configuration command on either the NAS or tunnel server.

Step 2 Establish a VPN session by dialing in to the NAS using an allowed username and password.

Step 3 Attempt to establish another VPN session by dialing in to the NAS using another allowed username and password.

Step 4 A Syslog message similar to the following should appear on the console of the router:

```
00:11:17:%VPDN-6-MAX_SESS_EXCD:L2F HGW great_went has exceeded configured local
session-limit and rejected user wilson@soam.com
```

Step 5 Enter the **show vpdn history failure** command on the router. If you see output similar to the following, the session limit was successful:

```
User:wilson@soam.com
 NAS:cliford_ball, IP address = 172.25.52.8, CLID = 2
 Gateway:great_went, IP address = 172.25.52.7, CLID = 13
 Log time:00:04:21, Error repeat count:1
 Failure type:Exceeded configured VPDN maximum session limit.
 Failure reason:
```

Enabling Soft Shutdown of VPN Tunnels

To prevent new sessions from being established on a VPN tunnel without disturbing the service of existing sessions, use the following command in global configuration mode:

Command	Purpose
`Router(config)# vpdn softshut`[1]	Prevents new sessions from being established on a VPN tunnel without disturbing existing sessions.

1. When the **vpdn softshut** command is enabled, Multichassis Multilink PPP (MMP) L2F tunnels can still be created and established.

When the **vpdn softshut** command is enabled on a NAS, the potential session will be authorized before it is refused. This authorization ensures that accurate accounting records can be kept.

When the **vpdn softshut** command is enabled on a tunnel server, the reason for the session refusal will be returned to the NAS. This information is recorded in the VPN history failure table.

To verify that the **vpdn softshut** command is working properly, perform the following steps:

Step 1 Establish a VPN session by dialing in to the NAS using an allowed username and password.

Step 2 Enter the **vpdn softshut** global configuration command on either the NAS or the tunnel server.

Step 3 Verify that the original session is still active by entering the **show vpdn** command:

```
ENT_HGW# show vpdn

% No active L2TP tunnels

L2F Tunnel and Session

 NAS CLID HGW CLID NAS Name         HGW Name         State
 36       1        cliford_ball    great_went       open
                   172.25.52.8     172.25.52.7

 CLID   MID    Username                    Intf   State
 36     1      mockingbird@gamehendge.com Vi1    open
```

Step 4 Attempt to establish another VPN session by dialing in to the NAS using another allowed username and password.

Step 5 A Syslog message similar to the following should appear on the console of the soft shutdown router:

```
00:11:17:%VPDN-6-SOFTSHUT:L2F HGW great_went has turned on softshut and rejected user
wilson@soam.com
```

Step 6 Enter the **show vpdn history failure** command on the soft shutdown router. If you see output similar to the following, the soft shutdown was successful:

```
User:wilson@soam.com
NAS:cliford_ball, IP address = 172.25.52.8, CLID = 2
Gateway:great_went, IP address = 172.25.52.7, CLID = 13
Log time:00:04:21, Error repeat count:1
Failure type:VPDN softshut has been activated.
Failure reason:
```

Configuring Event Logging

The Syslog mechanism provides generic and failure event logging. Generic logging is a mixture of type error, warning, notification, and information logging for VPN. Logging can be done locally or at a remote tunnel destination. Both generic and failure event logging is enabled by default; therefore, if you wish to disable VPN failure events you must specifically configure the router or access server to do so. In order to disable the router to log VPN generic or history events, use the following commands in global configuration mode:

Command	Purpose
`Router(config)# vpdn logging [local | remote]`	Enables generic event logging, locally or at a remote endpoint.
`Router(config)# vpdn history failure`	Enables the logging of failure events to the failure history table. **Note** By default, VPN failure history logging is enabled.

Setting the History Table Size

You may set the failure history table to a specific number of entries based on the amount of data you wish to track. To set the failure history table, use the following commands in global configuration mode:

Command	Purpose
`Router(config)# vpdn history failure table-size` *entries*	(Optional) Sets the failure history table depth.

Verifying VPN Sessions

The following sections detail the procedures used for verifying VPN sessions:

- Verifying a Client-Initiated VPN
- Verifying a NAS-Initiated VPN

Verifying a Client-Initiated VPN

To verify that a PPTP network functions properly, complete the following verification steps:

Step 1 From the client, dial in to the ISP and establish a PPP session.

Step 2 From the client, dial in to the tunnel server.

Step 3 From the client, ping the tunnel server. From the client desktop:

a. Click **Start**.

b. Select **Run**.

c. Enter **ping** *tunnel-server-ip-address.*

d. Click **OK**.

e. Look at the terminal screen and verify that the tunnel server is sending ping reply packets to the client.

Step 4 From the tunnel server, enter the **show vpdn** command and verify that the client has established a PPTP session.

```
PNS# show vpdn

% No active L2TP tunnels

% No active L2F tunnels
```

```
PPTP Tunnel and Session Information (Total tunnels=1 sessions=1)

LocID RemID Remote Name     State    Remote Address  Port  Sessions
13    13    10.1.2.41       estabd   10.1.2.41       1136  1

LocID RemID TunID Intf    Username      State   Last Chg
13    0     13    Vi3                   estabd  000030
```

Step 5 For more detailed information, enter the **show vpdn session all** or **show vpdn session window** commands. The last line of output from the **show vpdn session all** command indicates the current status of the flow control alarm.

```
PNS# show vpdn session all

% No active L2TP tunnels

% No active L2F tunnels

PPTP Session Information (Total tunnels=1 sessions=1)

Call id 13 is up on tunnel id 13
Remote tunnel name is 10.1.2.41
 Internet Address is 10.1.2.41
 Session username is unknown, state is estabd
 Time since change 000106, interface Vi3
 Remote call id is 0
 10 packets sent, 10 received, 332 bytes sent, 448 received
 Ss 11, Sr 10, Remote Nr 10, peer RWS 16
 0 out of order packets
 Flow alarm is clear.
```

The last line of output from the **show vpdn session window** command indicates the current status of the flow control alarm (under the heading “Congestion”) and the number of flow control alarms that have gone off during the session (under the heading “Alarms”).

```
PNS# show vpdn session window

% No active L2TP tunnels
% No active L2F tunnels
PPTP Session Information (Total tunnels=1 sessions=1)

LocID RemID TunID ZLB-tx  ZLB-rx  Congestion Alarms   Peer-RWS
13    0     13    0       1       clear      0        16
```

Step 6 For information on the virtual-access interface, enter the **show ppp mppe virtual-access** *number* command:

```
PNS# show ppp mppe virtual-access3

Interface Virtual-Access3 (current connection)
  Hardware (ISA5/1, flow_id=13) encryption, 40 bit encryption, Stateless mode
  packets encrypted = 0        packets decrypted  = 1
  sent CCP resets   = 0        receive CCP resets = 0
  next tx coherency = 0        next rx coherency  = 0
  tx key changes    = 0        rx key changes     = 0
  rx pkt dropped    = 0        rx out of order pkt= 0
  rx missed packets = 0
```

To update the key change information, reissue the **show ppp mppe virtual-access3** command.

```
PNS# show ppp mppe virtual-access3

Interface Virtual-Access3 (current connection)
```

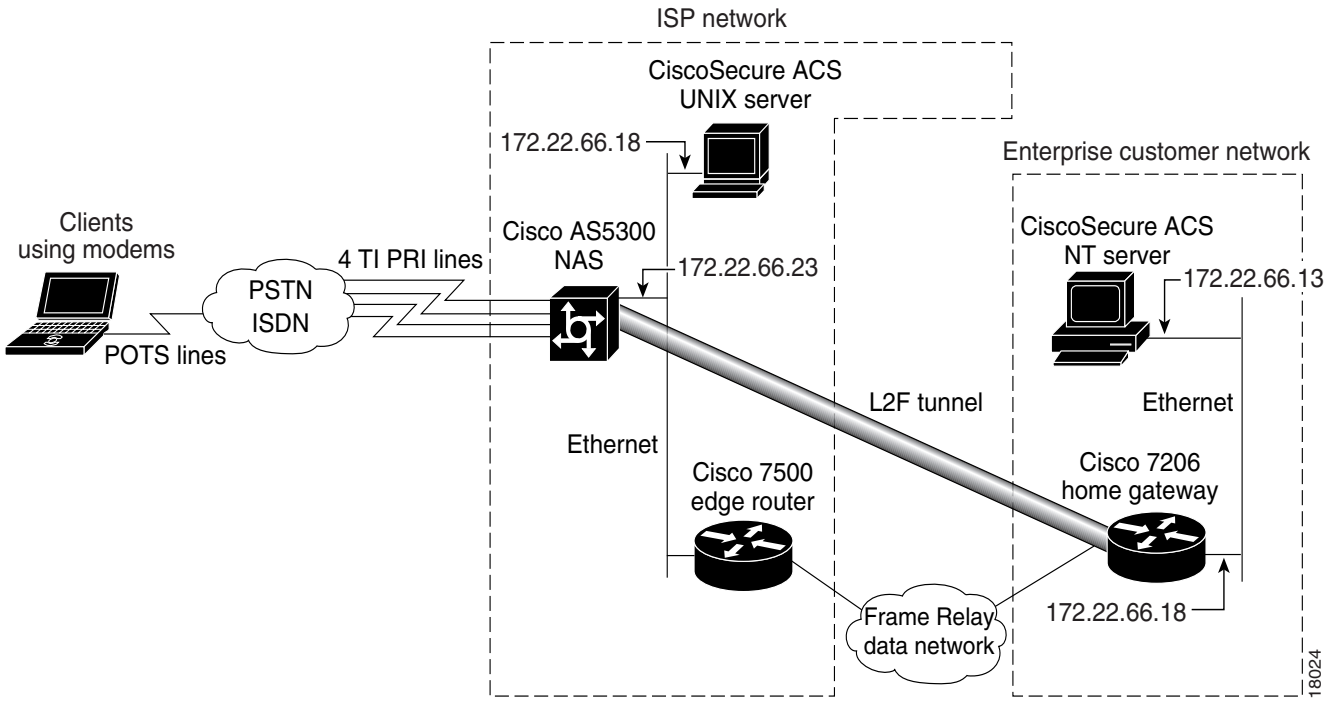
```
Hardware (ISA5/1, flow_id=13) encryption, 40 bit encryption, Stateless mode
packets encrypted = 0        packets decrypted  = 1
sent CCP resets   = 0        receive CCP resets = 0
next tx coherency = 0        next rx coherency  = 0
tx key changes    = 0        rx key changes     = 1
rx pkt dropped    = 0        rx out of order pkt= 0
rx missed packets = 0
```

Verifying a NAS-Initiated VPN

This section describes how to verify that an L2F dial-in scenario functions as shown in Figure 78. To verify connectivity, complete the following verification steps:

- Step 1: Dialing In to the NAS
- Step 2: Pinging the Tunnel Server
- Step 3: Displaying Active Call Statistics on the Tunnel Server
- Step 4: Pinging the Client
- Step 5: Verifying That the Virtual-Access Interface Is Up and That LCP Is Open
- Step 6: Viewing Active L2F Tunnel Statistics

Figure 78 L2F Dial-In Topology Using Remote AAA



Step 1 From the client, dial in to the NAS by using the PRI telephone number assigned to the NAS T1 trunks. Sometimes this telephone number is called the hunt group number.

As the call comes in to the NAS, a LINK-3-UPDOWN message automatically appears on the NAS terminal screen. In the following example, the call comes in to the NAS on asynchronous interface 14. The asynchronous interface is up.

```
*Jan  1 21:22:18.410: %LINK-3-UPDOWN: Interface Async14, changed state to up
```

Note No **debug** commands are turned on to display this log message. Start troubleshooting the NAS if you do not see this message 30 seconds after the client first sends the call.

Step 2 From the client, ping the tunnel server. From the client Windows 95 desktop, perform the following steps:

a. Click **Start**.

b. Select **Run**.

c. Enter the **ping** *ip-address* command, where the IP address is the tunnel server address.

d. Click **OK**.

e. Look at the terminal screen and verify that the tunnel server is sending ping reply packets to the client.

Step 3 From the tunnel server, enter the **show caller** command and the **show caller user** *name* command to verify that the client received an IP address. The following example shows that Jeremy is using interface virtual-access 1 and IP address 172.30.2.1. The network administrator jane-admin is using console 0.

```
ENT_HGW# show caller
  Line         User                Service       Active
  con 0        jane-admin          TTY           00:00:25
  Vi1          jeremy@hgw.com      PPP   L2F     00:01:28

ENT_HGW# show caller user jeremy@hgw.com

  User: jeremy@hgw.com, line Vi1, service PPP L2F, active 00:01:35
  PPP: LCP Open, CHAP (<- AAA), IPCP
  IP: Local 172.22.66.25, remote 172.30.2.1
  VPDN: NAS ISP_NAS, MID 1, MID open
        HGW  ENT_HGW, NAS CLID 36, HGW CLID 1, tunnel open
  Counts: 105 packets input, 8979 bytes, 0 no buffer
          0 input errors, 0 CRC, 0 frame, 0 overrun
          18 packets output, 295 bytes, 0 underruns
          0 output errors, 0 collisions, 0 interface resets
```

Step 4 From the tunnel server, ping Jeremy's PC at IP address 172.30.2.1:

```
ENT_HGW# ping 172.30.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 128/132/152 ms
```

Step 5 From the tunnel server, enter the **show interface virtual-access 1** command to verify that the interface is up, that LCP is open, and that no errors are reported:

```
ENT_HGW# show interface virtual-access 1
Virtual-Access1 is up, line protocol is up
  Hardware is Virtual Access interface
  Interface is unnumbered. Using address of FastEthernet0/0 (172.22.66.25)
  MTU 1500 bytes, BW 115 Kbit, DLY 100000 usec,
     reliablility 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set, keepalive set (10 sec)
  DTR is pulsed for 5 seconds on reset
```

```
  LCP Open
  Open: IPCP
  Last input 00:00:02, output never, output hang never
  Last clearing of "show interface" counters 3d00h
  Queueing strategy: fifo
  Output queue 1/40, 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
     114 packets input, 9563 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     27 packets output, 864 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
     0 carrier transitions
```

Step 6 From the tunnel server, display active tunnel statistics by entering the **show vpdn** command and the **show vpdn tunnel all** command:

```
ENT_HGW# show vpdn

% No active L2TP tunnels

L2F Tunnel and Session

 NAS CLID HGW CLID NAS Name        HGW Name        State
 36       1        ISP_NAS         ENT_HGW         open
                   172.22.66.23    172.22.66.25

 CLID   MID    Username                  Intf   State
 36     1      jeremy@hgw.com            Vi1    open

ENT_HGW# show vpdn tunnel all

% No active L2TP tunnels

L2F Tunnel
NAS name: ISP_NAS
NAS CLID: 36
NAS IP address 172.22.66.23
Gateway name: ENT_HGW
Gateway CLID: 1
Gateway IP address 172.22.66.25
State: open
Packets out: 52
Bytes out: 1799
Packets in: 100
Bytes in: 7143
```

Monitoring and Maintaining VPNs

To display useful information for monitoring and maintaining VPN sessions, use the following commands in privileged EXEC mode:

Command	Purpose
`Router# clear vpdn tunnel [pptp | l2f | l2tp] network-access-server gateway-name`	Shuts down a specific tunnel and all the sessions within the tunnel.
`Router# show interface virtual access number`	Displays information about the virtual access interface, LCP, protocol states, and interface statistics. The status of the virtual access interface should be: `Virtual-Access3 is up, line protocol is up`
`Router# show vpdn`	Displays a summary of all active VPN tunnels.
`Router# show vpdn domain`	Displays all VPN domains and DNIS groups configured on the NAS.
`Router# show vpdn group [name | name domain | name endpoint]`	Displays a summary of the relationships among VPDN groups and customer/VPDN profiles. When you include the name of the VPDN group, the output displays information on domain/DNIS, tunnel endpoint, session limits, group priority, active sessions, group status, and reserved sessions.
`Router# show vpdn history failure`	Displays information about VPN user failures.
`Router# show vpdn multilink`	Displays VPN multilink information.
`Router# show vpdn session [all | packets | sequence | state | timers | window] [interface | tunnel | username]`	Displays VPN session information including interface, tunnel, username, packets, status, and window statistics.
`Router# show vpdn tunnel [all | packets | state | summary | transport] [id | local-name | remote-name]`	Displays VPN tunnel information including tunnel protocol, ID, local and remote tunnel names, packets sent and received, tunnel, and transport status.

Troubleshooting VPNs

Troubleshooting components in VPN is not always straightforward because there are multiple technologies and OSI layers involved. To display detailed messages about VPN and VPN-related events, use the following commands in EXEC mode:

Command	Purpose
`Router# debug aaa authentication`	Displays information on AAA authentication.
`Router# debug aaa authorization`	Displays information on AAA authorization.
`Router# debug ppp chap`	Displays CHAP packet exchanges.
`Router# debug ppp mppe`	Displays debug messages for MPPE events.
`Router# debug ppp negotiation`	Displays information about packets sent during PPP startup and detailed PPP negotiation options.
`Router# debug vpdn error`	Displays errors that prevent a tunnel from being established or errors that cause an established tunnel to be closed.
`Router# debug vpdn event`	Displays messages about events that are part of normal tunnel establishment or shutdown.

Command	Purpose
`Router# debug vpdn l2tp-sequencing`	Displays message about L2TP tunnel sequencing.
`Router# debug vpdn l2x-data`	Display messages about L2F and L2TP data information.
`Router# debug vpdn l2x-errors`	Displays L2F and L2TP protocol errors that prevent L2F and L2TP establishment or prevent normal operation.

`Router# debug vpdn l2x-packets`
or
`Router# debug vpdn packet`
	Displays each protocol packet exchanged. This option may result in a large number of debug messages and should generally be used only on a debug chassis with a single active session.

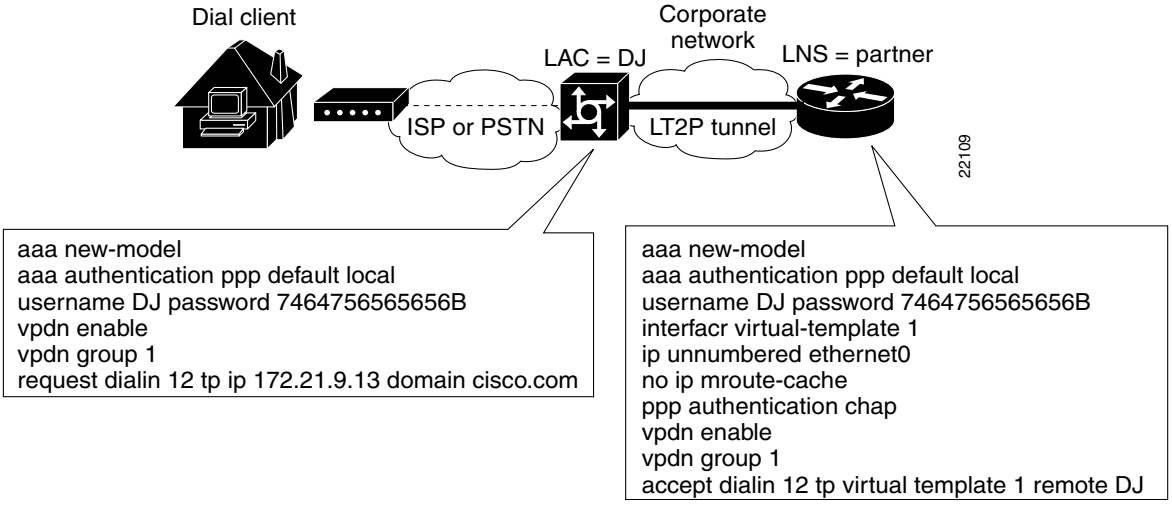
Successful Debug Examples

The following sections provide examples of debug output from successful VPN sessions:

- L2TP Dial-In Debug Output on NAS Example
- L2TP Dial-In Debug Output on a Tunnel Server Example
- L2TP Dial-Out Debug Output on a NAS Example
- L2TP Dial-Out Debug Output on a Tunnel Server Example

Figure 79 shows the topology used for the L2TP dial-in debug examples.

Figure 79 Topology Diagram for L2TP Dial-In Debug Example



L2TP Dial-In Debug Output on NAS Example

The following is debug output from a successful L2TP dial-in session on a NAS for the topology shown in Figure 79:

```
DJ# debug vpdn event

VPDN events debugging is on
```

```
DJ# debug vpdn l2x-events

L2X protocol events debugging is on

DJ# show debugging

VPN:
  L2X protocol events debugging is on
  VPDN events debugging is on
DJ#
20:47:33: %LINK-3-UPDOWN: Interface Async7, changed state to up
20:47:35: As7 VPDN: Looking for tunnel -- hoser.com --
20:47:35: As7 VPDN: Get tunnel info for hoser.com with NAS DJ, IP 172.21.9.13
20:47:35: As7 VPDN: Forward to address 172.21.9.13
20:47:35: As7 VPDN: Forwarding...
20:47:35: As7 VPDN: Bind interface direction=1
20:47:35: Tnl/Cl 8/1 L2TP: Session FS enabled
20:47:35: Tnl/Cl 8/1 L2TP: Session state change from idle to wait-for-tunnel
20:47:35: As7 8/1 L2TP: Create session
20:47:35: Tnl 8 L2TP: SM State idle
20:47:35: Tnl 8 L2TP: Tunnel state change from idle to wait-ctl-reply
20:47:35: Tnl 8 L2TP: SM State wait-ctl-reply
20:47:35: As7 VPDN: kath@hoser.com is forwarded
20:47:35: Tnl 8 L2TP: Got a challenge from remote peer, DJ
20:47:35: Tnl 8 L2TP: Got a response from remote peer, DJ
20:47:35: Tnl 8 L2TP: Tunnel Authentication success
20:47:35: Tnl 8 L2TP: Tunnel state change from wait-ctl-reply to established
20:47:35: Tnl 8 L2TP: SM State established
20:47:35: As7 8/1 L2TP: Session state change from wait-for-tunnel to wait-reply
20:47:35: As7 8/1 L2TP: Session state change from wait-reply to established
20:47:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async7, changed state to up
```

L2TP Dial-In Debug Output on a Tunnel Server Example

The following is debug output from a successful L2TP dial-in session on a tunnel server for the topology shown in Figure 79:

```
tunnel# debug vpdn l2x-events

L2X protocol events debugging is on

20:19:17: L2TP: I SCCRQ from DJ tnl 8
20:19:17: L2X: Never heard of DJ
20:19:17: Tnl 7 L2TP: New tunnel created for remote DJ, address 172.21.9.4
20:19:17: Tnl 7 L2TP: Got a challenge in SCCRQ, DJ
20:19:17: Tnl 7 L2TP: Tunnel state change from idle to wait-ctl-reply
20:19:17: Tnl 7 L2TP: Got a Challenge Response in SCCCN from DJ
20:19:17: Tnl 7 L2TP: Tunnel Authentication success
20:19:17: Tnl 7 L2TP: Tunnel state change from wait-ctl-reply to established
20:19:17: Tnl 7 L2TP: SM State established
20:19:17: Tnl/Cl 7/1 L2TP: Session FS enabled
20:19:17: Tnl/Cl 7/1 L2TP: Session state change from idle to wait-for-tunnel
20:19:17: Tnl/Cl 7/1 L2TP: New session created
20:19:17: Tnl/Cl 7/1 L2TP: O ICRP to DJ 8/1
20:19:17: Tnl/Cl 7/1 L2TP: Session state change from wait-for-tunnel to wait-connect
20:19:17: Tnl/Cl 7/1 L2TP: Session state change from wait-connect to established
20:19:17: Vi1 VPDN: Virtual interface created for kath@hoser.com
20:19:17: Vi1 VPDN: Set to Async interface
20:19:17: Vi1 VPDN: Clone from Vtemplate 1 filterPPP=0 blocking
20:19:18: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
20:19:18: Vi1 VPDN: Bind interface direction=2
```

```
20:19:18: Vi1 VPDN: PPP LCP accepting rcv CONFACK
20:19:19: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state
to up
```

L2TP Dial-Out Debug Output on a NAS Example

The following is sample output from the **debug dialer events** and **show debugging** EXEC commands for a successful dial-out session on a NAS:

```
NAS# debug dialer events

Dial on demand events debugging is on

NAS# show debugging

Dial on demand:
  Dial on demand events debugging is on
VPN:
  L2X protocol events debugging is on
  VPDN events debugging is on
NAS#
*Mar  1 00:05:26.155:%SYS-5-CONFIG_I:Configured from console by console
*Mar  1 00:05:26.899:%SYS-5-CONFIG_I:Configured from console by console
*Mar  1 00:05:36.195:L2TP:I SCCRQ from lns_l2x0 tnl 1
*Mar  1 00:05:36.199:Tnl 1 L2TP:New tunnel created for remote lns_l2x0, address
10.40.1.150
*Mar  1 00:05:36.203:Tnl 1 L2TP:Got a challenge in SCCRQ, lns_l2x0
*Mar  1 00:05:36.207:Tnl 1 L2TP:O SCCRP  to lns_l2x0 tnlid 1
*Mar  1 00:05:36.215:Tnl 1 L2TP:Tunnel state change from idle to wait-ctl-reply
*Mar  1 00:05:36.231:Tnl 1 L2TP:I SCCCN from lns_l2x0 tnl 1
*Mar  1 00:05:36.235:Tnl 1 L2TP:Got a Challenge Response in SCCCN from lns_l2x0
*Mar  1 00:05:36.239:Tnl 1 L2TP:Tunnel Authentication success
*Mar  1 00:05:36.239:Tnl 1 L2TP:Tunnel state change from wait-ctl-reply to established
*Mar  1 00:05:36.243:Tnl 1 L2TP:SM State established
*Mar  1 00:05:36.251:Tnl 1 L2TP:I OCRQ from lns_l2x0 tnl 1
*Mar  1 00:05:36.255:Tnl/Cl 1/1 L2TP:Session sequencing disabled
*Mar  1 00:05:36.259:Tnl/Cl 1/1 L2TP:Session FS enabled
*Mar  1 00:05:36.259:Tnl/Cl 1/1 L2TP:New session created
*Mar  1 00:05:36.263:12C:Same state, 0
*Mar  1 00:05:36.267:DSES 12C:Session create
*Mar  1 00:05:36.271:L2TP:Send OCRP
*Mar  1 00:05:36.275:Tnl/Cl 1/1 L2TP:Session state change from idle to wait-cs-answer
*Mar  1 00:05:36.279:DSES 0x12C:Building dialer map
*Mar  1 00:05:36.283:Dialout 0x12C:Next hop name is 71014
*Mar  1 00:05:36.287:Serial0:23 DDR:rotor dialout [priority]
*Mar  1 00:05:36.291:Serial0:23 DDR:Dialing cause dialer session 0x12C
*Mar  1 00:05:36.291:Serial0:23 DDR:Attempting to dial 71014
*Mar  1 00:05:36.479:%LINK-3-UPDOWN:Interface Serial0:22, changed state to up
*Mar  1 00:05:36.519:isdn_call_connect:Calling lineaction of Serial0:22
*Mar  1 00:05:36.519:Dialer0:Session free, 12C
*Mar  1 00:05:36.523::0 packets unqueued and discarded
*Mar  1 00:05:36.527:Se0:22 VPDN:Bind interface direction=1
*Mar  1 00:05:36.531:Se0:22 1/1 L2TP:Session state change from wait-cs-answer to
established
*Mar  1 00:05:36.531:L2TP:Send OCCN
*Mar  1 00:05:36.539:Se0:22 VPDN:bound to vpdn session
*Mar  1 00:05:36.555:Se0:22 1/1 L2TP:O FS failed
*Mar  1 00:05:36.555:Se0:22 1/1 L2TP:O FS failed
*Mar  1 00:05:42.515:%ISDN-6-CONNECT:Interface Serial0:22 is now connected to 71014
```

L2TP Dial-Out Debug Output on a Tunnel Server Example

The following is sample debug output from the **debug vpdn event**, **debug vpdn error**, **debug ppp chap**, **debug ppp negotiation**, and **debug dialer events** commands for a successful dial-out session on a tunnel server:

```
LNS# debug dialer events

Dial on demand events debugging is on

LNS# debug ppp negotiation

PPP protocol negotiation debugging is on

LNS# debug ppp chap

PPP authentication debugging is on

LNS# show debugging

Dial on demand:
  Dial on demand events debugging is on
PPP:
  PPP authentication debugging is on
  PPP protocol negotiation debugging is on
VPN:
  VPDN events debugging is on
  VPDN errors debugging is on
LNS#
*Apr 22 19:48:32.419:%SYS-5-CONFIG_I:Configured from console by console
*Apr 22 19:48:32.743:%SYS-5-CONFIG_I:Configured from console by console
*Apr 22 19:48:33.243:Di0 DDR:dialer_fsm_idle()
*Apr 22 19:48:33.271:Vi1 PPP:Phase is DOWN, Setup
*Apr 22 19:48:33.279:Vi1 PPP:Phase is DOWN, Setup
*Apr 22 19:48:33.279:Virtual-Access1 DDR:Dialing cause ip (s=10.60.1.160, d=10.10.1.110)
*Apr 22 19:48:33.279:Virtual-Access1 DDR:Attempting to dial 71014
*Apr 22 19:48:33.279:Tnl/Cl 1/1 L2TP:Session sequencing disabled
*Apr 22 19:48:33.279:Tnl/Cl 1/1 L2TP:Session FS enabled
*Apr 22 19:48:33.283:Tnl/Cl 1/1 L2TP:Session state change from idle to wait-for-tunnel
*Apr 22 19:48:33.283:Tnl/Cl 1/1 L2TP:Create dialout session
*Apr 22 19:48:33.283:Tnl 1 L2TP:SM State idle
*Apr 22 19:48:33.283:Tnl 1 L2TP:O SCCRQ
*Apr 22 19:48:33.283:Tnl 1 L2TP:Tunnel state change from idle to wait-ctl-reply
*Apr 22 19:48:33.283:Tnl 1 L2TP:SM State wait-ctl-reply
*Apr 22 19:48:33.283:Vi1 VPDN:Bind interface direction=2
*Apr 22 19:48:33.307:Tnl 1 L2TP:I SCCRP from lac_l2x0
*Apr 22 19:48:33.307:Tnl 1 L2TP:Got a challenge from remote peer, lac_l2x0
*Apr 22 19:48:33.307:Tnl 1 L2TP:Got a response from remote peer, lac_l2x0
*Apr 22 19:48:33.311:Tnl 1 L2TP:Tunnel Authentication success
*Apr 22 19:48:33.311:Tnl 1 L2TP:Tunnel state change from wait-ctl-reply to established
*Apr 22 19:48:33.311:Tnl 1 L2TP:O SCCCN  to lac_l2x0 tnlid 1
*Apr 22 19:48:33.311:Tnl 1 L2TP:SM State established
*Apr 22 19:48:33.311:L2TP:O OCRQ
*Apr 22 19:48:33.311:Vi1 1/1 L2TP:Session state change from wait-for-tunnel to wait-reply
*Apr 22 19:48:33.367:Vi1 1/1 L2TP:I OCRP from lac_l2x0 tnl 1, cl 0
*Apr 22 19:48:33.367:Vi1 1/1 L2TP:Session state change from wait-reply to wait-connect
*Apr 22 19:48:33.631:Vi1 1/1 L2TP:I OCCN from lac_l2x0 tnl 1, cl 1
*Apr 22 19:48:33.631:Vi1 1/1 L2TP:Session state change from wait-connect to established
*Apr 22 19:48:33.631:Vi1 VPDN:Connection is up, start LCP negotiation now
*Apr 22 19:48:33.631:%LINK-3-UPDOWN:Interface Virtual-Access1, changed state to up
*Apr 22 19:48:33.631:Vi1 DDR:dialer_statechange(), state=4Dialer statechange to up
Virtual-Access1
*Apr 22 19:48:33.631:Vi1 DDR:dialer_out_call_connected()
```

```
*Apr 22 19:48:33.631:Vi1 DDR:dialer_bind_profile() to Di0
*Apr 22 19:48:33.631:%DIALER-6-BIND:Interface Virtual-Access1 bound to profile
Dialer0Dialer call has been placed Virtual-Access1
*Apr 22 19:48:33.635:Vi1 PPP:Treating connection as a callout
*Apr 22 19:48:33.635:Vi1 PPP:Phase is ESTABLISHING, Active Open
*Apr 22 19:48:33.635:Vi1 LCP:O CONFREQ [Closed] id 1 len 15
*Apr 22 19:48:33.635:Vi1 LCP:   AuthProto CHAP (0x0305C22305)
*Apr 22 19:48:33.635:Vi1 LCP:   MagicNumber 0x50E7EC2A (0x050650E7EC2A)
*Apr 22 19:48:33.663:Vi1 LCP:I CONFREQ [REQsent] id 1 len 15
*Apr 22 19:48:33.663:Vi1 LCP:   AuthProto CHAP (0x0305C22305)
*Apr 22 19:48:33.663:Vi1 LCP:   MagicNumber 0x10820474 (0x050610820474)
*Apr 22 19:48:33.663:Vi1 LCP:O CONFACK [REQsent] id 1 len 15
*Apr 22 19:48:33.663:Vi1 LCP:   AuthProto CHAP (0x0305C22305)
*Apr 22 19:48:33.663:Vi1 LCP:   MagicNumber 0x10820474 (0x050610820474)
*Apr 22 19:48:33.663:Vi1 LCP:I CONFACK [ACKsent] id 1 len 15
*Apr 22 19:48:33.663:Vi1 LCP:   AuthProto CHAP (0x0305C22305)
*Apr 22 19:48:33.663:Vi1 LCP:   MagicNumber 0x50E7EC2A (0x050650E7EC2A)
*Apr 22 19:48:33.663:Vi1 LCP:State is Open
*Apr 22 19:48:33.663:Vi1 PPP:Phase is AUTHENTICATING, by both
*Apr 22 19:48:33.663:Vi1 CHAP:Using alternate hostname lns0
*Apr 22 19:48:33.663:Vi1 CHAP:O CHALLENGE id 1 len 25 from "lns0"
*Apr 22 19:48:33.679:Vi1 CHAP:I CHALLENGE id 1 len 35 from "user0@foo.com0"
*Apr 22 19:48:33.679:Vi1 AUTH:Started process 0 pid 92
*Apr 22 19:48:33.679:Vi1 CHAP:Using alternate hostname lns0
*Apr 22 19:48:33.683:Vi1 CHAP:O RESPONSE id 1 len 25 from "lns0"
*Apr 22 19:48:33.695:Vi1 CHAP:I SUCCESS id 1 len 4
*Apr 22 19:48:33.699:Vi1 CHAP:I RESPONSE id 1 len 35 from "user0@foo.com0"
*Apr 22 19:48:33.699:Vi1 CHAP:O SUCCESS id 1 len 4
*Apr 22 19:48:33.699:Vi1 DDR:dialer_remote_name() for user0@foo.com0
*Apr 22 19:48:33.699:Vi1 PPP:Phase is UP
*Apr 22 19:48:33.703:Vi1 IPCP:O CONFREQ [Closed] id 1 len 10
*Apr 22 19:48:33.703:Vi1 IPCP:   Address 10.20.1.150 (0x030614140196)
*Apr 22 19:48:33.703:Vi1 CCP:O CONFREQ [Closed] id 1 len 10
*Apr 22 19:48:33.703:Vi1 CCP:   LZSDCP history 1 check mode SEQ process UNCOMPRESSSED
(0x170600010201)
*Apr 22 19:48:33.711:Vi1 IPCP:I CONFREQ [REQsent] id 1 len 10
*Apr 22 19:48:33.715:Vi1 IPCP:   Address 10.20.1.120 (0x030614140178)
*Apr 22 19:48:33.715:Vi1 IPCP:O CONFACK [REQsent] id 1 len 10
*Apr 22 19:48:33.715:Vi1 IPCP:   Address 10.20.1.120 (0x030614140178)
*Apr 22 19:48:33.715:Vi1 CCP:I CONFREQ [REQsent] id 1 len 10
*Apr 22 19:48:33.715:Vi1 CCP:   LZSDCP history 1 check mode SEQ process UNCOMPRESSSED
(0x170600010201)
*Apr 22 19:48:33.715:Vi1 CCP:O CONFACK [REQsent] id 1 len 10
*Apr 22 19:48:33.715:Vi1 CCP:   LZSDCP history 1 check mode SEQ process UNCOMPRESSSED
(0x170600010201)
*Apr 22 19:48:33.719:Vi1 IPCP:I CONFACK [ACKsent] id 1 len 10
*Apr 22 19:48:33.719:Vi1 IPCP:   Address 10.20.1.150 (0x030614140196)
*Apr 22 19:48:33.719:Vi1 IPCP:State is Open
*Apr 22 19:48:33.719:Vi1 DDR:Dialer protocol up
*Apr 22 19:48:33.719:Dialer0:dialer_ckt_swt_client_connect:incoming circuit switched call
*Apr 22 19:48:33.719:Di0 IPCP:Install route to 10.20.1.120
*Apr 22 19:48:33.719:Vi1 CCP:I CONFACK [ACKsent] id 1 len 10
*Apr 22 19:48:33.719:Vi1 CCP:   LZSDCP history 1 check mode SEQ process UNCOMPRESSSED
(0x170600010201)
*Apr 22 19:48:33.719:Vi1 CCP:State is Open
*Apr 22 19:48:34.699:%LINEPROTO-5-UPDOWN:Line protocol on Interface Virtual-Access1,
changed state to up
```

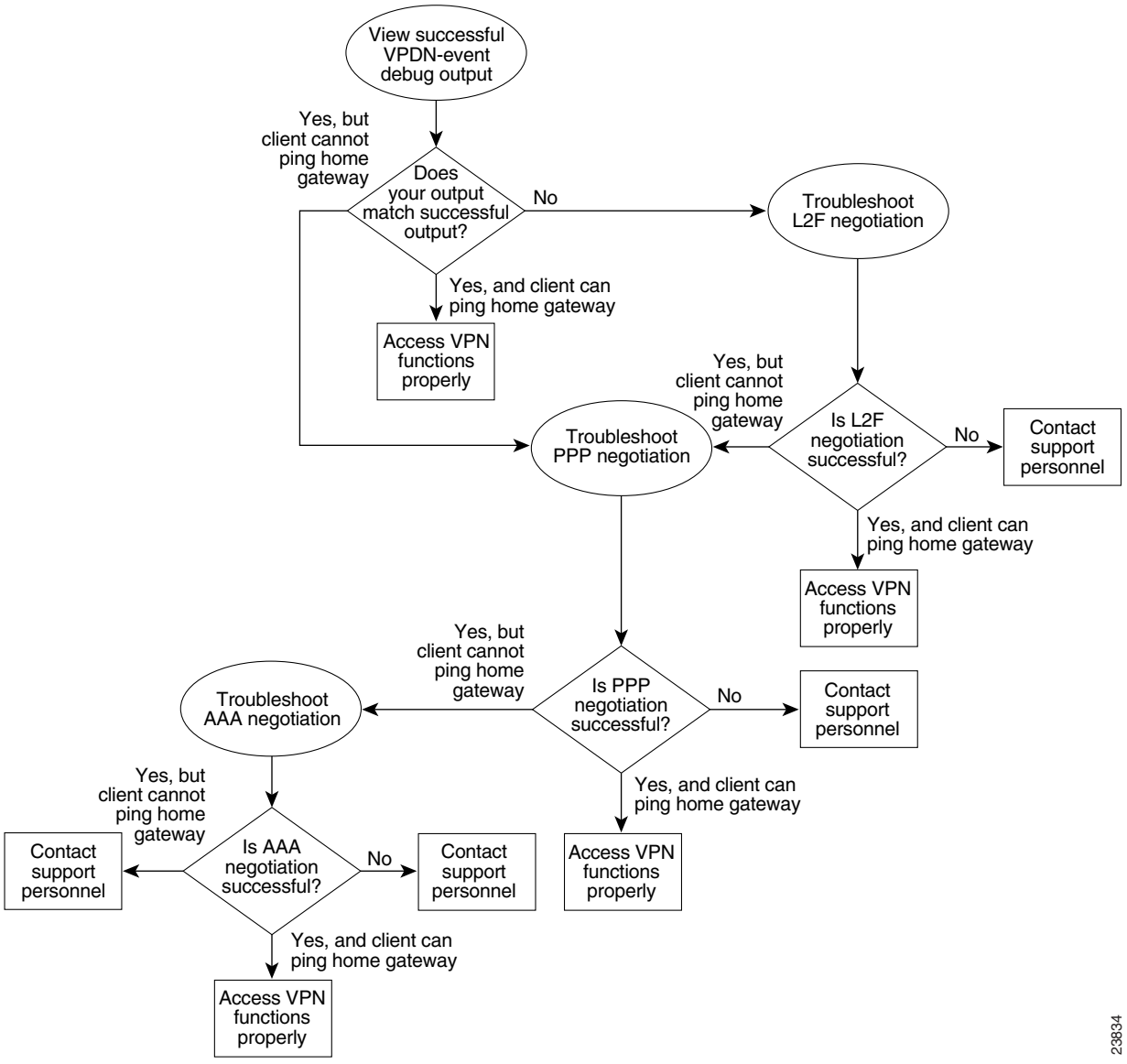
VPN Troubleshooting Methodology

This section describes a methodology for troubleshooting the VPN shown in Figure 80. First, view the debug output from a successful call. If your debug output does not match the successful output, follow the remaining steps to begin troubleshooting the network. The bolded lines of debug output indicate important information.

The following sections detail the steps involved in VPN troubleshooting:

- Comparing Your Debug Output to the Successful Debug Output
- Troubleshooting VPN Negotiation
- Troubleshooting PPP Negotiation
- Troubleshooting AAA Negotiation

Figure 80 Troubleshooting Flow Diagram for Access VPN with Remote AAA



If you are accessing the NAS and tunnel server through a Telnet connection, you need to enable the **terminal monitor** command. This command ensures that your EXEC session is receiving the logging and debug output from the devices.

When you finish troubleshooting, use the **undebug all** command to turn off all debug commands. Isolating debug output helps you efficiently build a network.

Comparing Your Debug Output to the Successful Debug Output

Enable the **debug vpdn-event** command on both the NAS and the tunnel server and dial in to the NAS. The following debug output shows successful VPN negotiation on the NAS and tunnel server:

```
NAS#
Jan  7 00:19:35.900: %LINK-3-UPDOWN: Interface Async9, changed state to up
Jan  7 00:19:39.532: sVPDN: Got DNIS string As9
Jan  7 00:19:39.532: As9 VPDN: Looking for tunnel -- hgw.com --
Jan  7 00:19:39.540: As9 VPDN: Get tunnel info for hgw.com with NAS ISP_NAS,
IP172.22.66.25
Jan  7 00:19:39.540: As9 VPDN: Forward to address 172.22.66.25
Jan  7 00:19:39.540: As9 VPDN: Forwarding...
Jan  7 00:19:39.540: As9 VPDN: Bind interface direction=1
Jan  7 00:19:39.540: As9 VPDN: jeremy@hgw.com is forwarded
Jan  7 00:19:40.540: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async9, changed state
to up

ENT_HGW#
Jan  7 00:19:39.967: VPDN: Chap authentication succeeded for ISP_NAS
Jan  7 00:19:39.967: Vi1 VPDN: Virtual interface created for jeremy@hgw.com
Jan  7 00:19:39.967: Vi1 VPDN: Set to Async interface
Jan  7 00:19:39.971: Vi1 VPDN: Clone from Vtemplate 1 filterPPP=0 blocking
6w5d: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
Jan  7 00:19:40.051: Vi1 VPDN: Bind interface direction=2
Jan  7 00:19:40.051: Vi1 VPDN: PPP LCP accepted rcv CONFACK
Jan  7 00:19:40.051: Vi1 VPDN: PPP LCP accepted sent CONFACK
6w5d: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
```

If you see the debug output shown but cannot ping the tunnel server, go to the next section, "Troubleshooting PPP Negotiation."

If you do not see the above debug output, go to the section "Troubleshooting VPN Negotiation" later in this chapter.

Troubleshooting VPN Negotiation

The following sections describe several common misconfigurations that prevent successful VPN (either L2F or L2TP) negotiation:

- Misconfigured NAS Tunnel Secret
- Misconfigured Tunnel Server Tunnel Secret
- Misconfigured Tunnel Name
- Control Packet Problem on the NAS

Misconfigured NAS Tunnel Secret

The NAS and the tunnel server must both have the same usernames with the same password to authenticate the L2F tunnel. These usernames are called the tunnel secret. In this scenario, these usernames are ISP_NAS and ENT_HGW. The password is cisco for both usernames on both systems.

If one of the tunnel secrets on the NAS is incorrect, you will see the following debug output when you dial in to the NAS and the **debug vpdn l2x-errors** command is enabled on the NAS and tunnel server:

```
NAS#
Jan  1 00:26:49.899: %LINK-3-UPDOWN: Interface Async3, changed state to up
Jan  1 00:26:54.643: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async3, cha
nged state to up
Jan  1 00:27:00.559: L2F: Resending L2F_OPEN, time #1
Jan  1 00:27:05.559: L2F: Resending L2F_ECHO, time #1
Jan  1 00:27:05.559: L2F: Resending L2F_OPEN, time #2
Jan  1 00:27:10.559: L2F: Resending L2F_ECHO, time #2
Jan  1 00:27:10.559: L2F: Resending L2F_OPEN, time #3
Jan  1 00:27:15.559: L2F: Resending L2F_ECHO, time #3
Jan  1 00:27:15.559: L2F: Resending L2F_OPEN, time #4
Jan  1 00:27:20.559: L2F: Resending L2F_ECHO, time #4
Jan  1 00:27:20.559: L2F: Resending L2F_OPEN, time #5
Jan  1 00:27:25.559: L2F: Resending L2F_ECHO, time #5
Jan  1 00:27:25.559: L2F: Resend packet (type 2) around too long, time to kill off the
tunnel
NAS#

ENT_HGW#
Jan  1 00:26:53.645: L2F: Packet has bogus2 key C8353FAB B6369121
5w6d: %VPDN-6-AUTHENFAIL: L2F HGW , authentication failure for  tunnel ISP_NAS; Invalid
key
5w6d: %VPDN-5-UNREACH: L2F NAS 172.22.66.23 is unreachable
Jan  1 00:27:00.557: L2F: Gateway received tunnel OPEN while in state closed
ENT_HGW#
```

The phrase “time to kill off the tunnel” in the NAS debug output indicates that the tunnel was not opened. The phrase “Packet has bogus2 key” in the tunnel server debug output indicates that the NAS has an incorrect tunnel secret.

To avoid this problem, make sure that you configure both the NAS and tunnel server for the same two tunnel secret usernames with the same password.

Misconfigured Tunnel Server Tunnel Secret

If one of the tunnel secret usernames on the tunnel server is incorrect, the following debug output appears when you dial in to the NAS and the **debug vpdn l2x-errors** command is enabled on the NAS and tunnel server:

```
NAS#
Jan  1 00:45:27.123: %LINK-3-UPDOWN: Interface Async7, changed state to up
Jan  1 00:45:30.939: L2F: Packet has bogus1 key B6C656EE 5FAC6B3
Jan  1 00:45:30.939: %VPDN-6-AUTHENFAIL: L2F NAS ISP_NAS, authentication failure
 for  tunnel ENT_HGW; Invalid key
Jan  1 00:45:31.935: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async7, cha
nged state to up
Jan  1 00:45:35.559: L2F: Resending L2F_OPEN, time #1
Jan  1 00:45:35.559: L2F: Packet has bogus1 key B6C656EE 5FAC6B3

ENT_HGW#
Jan  1 00:45:30.939: L2F: Tunnel authentication succeeded for ISP_NAS
Jan  1 00:45:35.559: L2F: Gateway received tunnel OPEN while in state open
Jan  1 00:45:40.559: L2F: Gateway received tunnel OPEN while in state open
```

```
Jan  1 00:45:45.559: L2F: Gateway received tunnel OPEN while in state open
Jan  1 00:45:50.559: L2F: Gateway received tunnel OPEN while in state open
```

Notice how this output is similar to the debug output you see when the NAS has a misconfigured tunnel secret username. This time you see the phrase "Packet has bogus1 key" on the NAS instead of the tunnel server. This phrase tells you that the tunnel server has an incorrect tunnel secret username.

To avoid this problem, make sure that you configure both the NAS and tunnel server for the same two tunnel secret usernames with the same password.

Misconfigured Tunnel Name

If the NAS and tunnel server do not have matching tunnel names, they cannot establish an L2F tunnel. On the tunnel server, these tunnel names are configured under the **vpdn-group 1** command by using the **local name** command. On the NAS, these names are configured on the RADIUS server.

The tunnel server must be configured to accept tunnels from the name that the NAS sends it. This is done using the **accept-dialin l2f virtual-template 1 remote ISP_NAS** command, where **ISP_NAS** is the name. The name it returns to the NAS is configured using the **local name ENT_HGW** command, where **ENT_HGW** is the name. These commands appear in the following running configuration:

```
vpdn-group 1
 accept-dialin l2f virtual-template 1 remote ISP_NAS
 local name ENT_HGW
```

On the RADIUS server, the tunnel names are configured by adding profiles to the **NAS_Group** group with the names **ISP_NAS** and **ENT_HGW**.

In the following debug output, the NAS attempted to open a tunnel using the name isp. Because the tunnel server did not know this name, it did not open the tunnel. To see the following debug output, enable the **debug vpdn l2x-events** and **debug vpdn l2x-errors** commands on the tunnel server:

```
ENT_HGW#
Jan  1 01:28:54.207: L2F: L2F_CONF received
Jan  1 01:28:54.207: L2X: Never heard of isp
Jan  1 01:28:54.207: L2F: Couldn't find tunnel named isp
```

To avoid the problem described, make sure that the tunnel names match on the tunnel server and on the RADIUS server.

Control Packet Problem on the NAS

The following example assumes that you suspect an error in parsing control packets. You can use the **debug vpdn packet** command with the **control** keyword to verify control packet information.

```
ISP_NAS# debug vpdn packet control

20:50:27: %LINK-3-UPDOWN: Interface Async7, changed state to up
20:50:29: Tnl 9 L2TP: O SCCRQ
20:50:29: Tnl 9 L2TP: O SCCRQ, flg TLF, ver 2, len 131, tnl 0, cl 0, ns 0, nr 0
20:50:29: contiguous buffer, size 131
         C8 02 00 83 00 00 00 00 00 00 00 00 80 08 00 00
         00 00 00 01 80 08 00 00 00 02 01 00 80 0A 00 00
         00 03 00 00 00 03 80 0A 00 00 00 04 00 00 00 ...
20:50:29: Tnl 9 L2TP: Parse AVP 0, len 8, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Parse SCCRP
20:50:29: Tnl 9 L2TP: Parse AVP 2, len 8, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Protocol Ver 256
20:50:29: Tnl 9 L2TP: Parse AVP 3, len 10, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Framing Cap 0x0x3
20:50:29: Tnl 9 L2TP: Parse AVP 4, len 10, flag 0x0x8000 (M)
```

```
20:50:29: Tnl 9 L2TP: Bearer Cap 0x0x3
20:50:29: Tnl 9 L2TP: Parse AVP 6, len 8, flag 0x0x0
20:50:29: Tnl 9 L2TP: Firmware Ver 0x0x1120
20:50:29: Tnl 9 L2TP: Parse AVP 7, len 12, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Hostname DJ
20:50:29: Tnl 9 L2TP: Parse AVP 8, len 25, flag 0x0x0
20:50:29: Tnl 9 L2TP: Vendor Name Cisco Systems, Inc.
20:50:29: Tnl 9 L2TP: Parse AVP 9, len 8, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Assigned Tunnel ID 8
20:50:29: Tnl 9 L2TP: Parse AVP 10, len 8, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Rx Window Size 4
20:50:29: Tnl 9 L2TP: Parse AVP 11, len 22, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Chlng D807308D106259C5933C6162ED3A1689
20:50:29: Tnl 9 L2TP: Parse AVP 13, len 22, flag 0x0x8000 (M)
20:50:29: Tnl 9 L2TP: Chlng Resp 9F6A3C70512BD3E2D44DF183C3FFF2D1
20:50:29: Tnl 9 L2TP: No missing AVPs in SCCRP
20:50:29: Tnl 9 L2TP: Clean Queue packet 0
20:50:29: Tnl 9 L2TP: I SCCRP, flg TLF, ver 2, len 153, tnl 9, cl 0, ns 0, nr 1
 contiguous pak, size 153
         C8 02 00 99 00 09 00 00 00 00 00 01 80 08 00 00
         00 00 00 02 80 08 00 00 00 02 01 00 80 0A 00 00
         00 03 00 00 00 03 80 0A 00 00 00 04 00 00 00 ...
20:50:29: Tnl 9 L2TP: I SCCRP from DJ
20:50:29: Tnl 9 L2TP: O SCCCN  to DJ tnlid 8
20:50:29: Tnl 9 L2TP: O SCCCN, flg TLF, ver 2, len 42, tnl 8, cl 0, ns 1, nr 1
20:50:29: contiguous buffer, size 42
         C8 02 00 2A 00 08 00 00 00 01 00 01 80 08 00 00
         00 00 00 03 80 16 00 00 00 0D 4B 2F A2 50 30 13
         E3 46 58 D5 35 8B 56 7A E9 85
20:50:29: As7 9/1 L2TP: O ICRQ to DJ 8/0
20:50:29: As7 9/1 L2TP: O ICRQ, flg TLF, ver 2, len 48, tnl 8, cl 0, ns 2, nr 1
20:50:29: contiguous buffer, size 48
         C8 02 00 30 00 08 00 00 00 02 00 01 80 08 00 00
         00 00 00 0A 80 08 00 00 00 0E 00 01 80 0A 00 00
         00 0F 00 00 00 04 80 0A 00 00 00 12 00 00 00 ...
20:50:29: Tnl 9 L2TP: Clean Queue packet 1
20:50:29: Tnl 9 L2TP: Clean Queue packet 2
20:50:29: Tnl 9 L2TP: I ZLB ctrl ack, flg TLF, ver 2, len 12, tnl 9, cl 0, ns 1, nr 2
 contiguous pak, size 12
         C8 02 00 0C 00 09 00 00 00 01 00 02
20:50:30: As7 9/1 L2TP: Parse AVP 0, len 8, flag 0x0x8000 (M)
20:50:30: As7 9/1 L2TP: Parse ICRP
20:50:30: As7 9/1 L2TP: Parse AVP 14, len 8, flag 0x0x8000 (M)
20:50:30: As7 9/1 L2TP: Assigned Call ID 1
20:50:30: As7 9/1 L2TP: No missing AVPs in ICRP
20:50:30: Tnl 9 L2TP: Clean Queue packet 2
20:50:30: As7 9/1 L2TP: I ICRP, flg TLF, ver 2, len 28, tnl 9, cl 1, ns 1, nr 3
 contiguous pak, size 28
         C8 02 00 1C 00 09 00 01 00 01 00 03 80 08 00 00
         00 00 00 0B 80 08 00 00 00 0E 00 01
20:50:30: As7 9/1 L2TP: O ICCN to DJ 8/1
20:50:30: As7 9/1 L2TP: O ICCN, flg TLF, ver 2, len 203, tnl 8, cl 1, ns 3, nr 2
20:50:30: contiguous buffer, size 203
         C8 02 00 CB 00 08 00 01 00 03 00 02 80 08 00 00
         00 00 00 0C 80 0A 00 00 00 18 00 00 DA C0 80 0A
         00 00 00 13 00 00 00 02 00 28 00 00 00 1B 02 ...
20:50:30: Tnl 9 L2TP: Clean Queue packet 3
20:50:30: As7 9/1 L2TP: I ZLB ctrl ack, flg TLF, ver 2, len 12, tnl 9, cl 1, ns 2, nr 4
 contiguous pak, size 12
         C8 02 00 0C 00 09 00 01 00 02 00 04
20:50:30: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async7, changed state to up
```

If you fixed the problem in your configuration, return to the section "Verifying VPN Sessions" earlier in this chapter.

If your call still cannot successfully complete L2F negotiation, contact your support personnel.

Troubleshooting PPP Negotiation

This section first shows debug output of successful PPP negotiation. The subsequent sections explain several common problems that prevent successful PPP negotiation:

- Successful PPP Negotiation Debug Output
- Non-Cisco Device Connectivity Problem
- Mismatched Username Example

Enable the **debug ppp negotiation** command on the tunnel server and dial in to the NAS.

Successful PPP Negotiation Debug Output

The following debug output shows successful PPP negotiation on the tunnel server:

```
1d02h: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
*Feb  4 14:14:40.505: Vi1 PPP: Treating connection as a dedicated line
*Feb  4 14:14:40.505: Vi1 PPP: Phase is ESTABLISHING, Active Open
*Feb  4 14:14:40.505: Vi1 PPP: Treating connection as a dedicated line
*Feb  4 14:14:40.505: Vi1 PPP: Phase is AUTHENTICATING, by this end
*Feb  4 14:14:40.509: Vi1 PPP: Phase is UP
```

If your call successfully completed PPP negotiation, but you still cannot ping the tunnel server, go to the section "Troubleshooting AAA Negotiation" later in this chapter.

Non-Cisco Device Connectivity Problem

The **debug ppp authentication** and **debug ppp negotiation** commands are enabled to decipher a CHAP negotiation problem. This is due to a connectivity problem between a Cisco and non-Cisco device. Also note that the **service-timestamps** command is enabled on the router. The **service-timestamps** command is helpful to decipher timing and keepalive issues, and we recommend that you always enable this command.

```
Router# debug ppp authentication

PPP authentication debugging is on

Router# debug ppp negotiation

PPP protocol negotiation debugging is on
3:22:53: ppp: sending CONFREQ, type = 3 (CI_AUTHTYPE), value = C223/5
3:22:53: ppp: sending CONFREQ, type = 5 (CI_MAGICNUMBER), value = C6091F.
3:22:55: ppp: sending CONFREQ, type = 3 (CI_AUTHTYPE), value = C223/5
3:22:55: ppp: sending CONFREQ, type = 5 (CI_MAGICNUMBER), value = C6091F
3:22:55: PPP BRI0: B-Channel 1: received config for type = 0x0 (??)
3:22:55: PPP BRI0: B-Channel 1: rcvd unknown option 0x0 rejected
3:22:55: PPP BRI0: B-Channel 1: received config for type = 0x1 (MRU) value = 0x5
F4 rejected
3:22:55: PPP BRI0: B-Channel 1: received config for type = 0x3 (AUTHTYPE) value
= 0xC223 value = 0x5 acked
3:22:55: PPP BRI0: B-Channel 1: received config for type = 0x11 (MULTILINK_MRRU)
rejected
3:22:55: PPP BRI0: B-Channel 1: received config for type = 0x13 (UNKNOWN)
3:22:55: PPP BRI0: B-Channel 1: rcvd unknown option 0x13 rejected
```

```
3:22:55: ppp: config REJ received, type = 3 (CI_AUTHTYPE), value = C223/5
3:22:55: ppp: sending CONFREQ, type = 3 (CI_AUTHTYPE), value = C223/5
3:22:55: ppp: sending CONFREQ, type = 5 (CI_MAGICNUMBER), value = C6091F
3:22:55: PPP BRI0: B-Channel 1: received config for type = 0x3 (AUTHTYPE) value= 0xC2.
Success rate is 0 percent (0/5)
moog#23 value = 0x5 acked
3:22:55: ppp: config REJ received, type = 3 (CI_AUTHTYPE), value = C223/5

3:22:55: ppp: BRI0: B-Channel 1 closing connection because remote won't authenticate

3:22:55: ppp: sending CONFREQ, type = 3 (CI_AUTHTYPE), value = C223/5
3:22:55: ppp: sending CONFREQ, type = 5 (CI_MAGICNUMBER), value = C6091F
3:22:55: %ISDN-6-DISCONNECT: Interface BRI0: B-Channel 1  disconnected from 0123
5820040 , call lasted 2 seconds
3:22:56: %LINK-3-UPDOWN: Interface BRI0: B-Channel 1, changed state to down
Indication:
```

Mismatched Username Example

The following **debug ppp chap** sample output excerpt shows a CHAP authentication failure caused by a configuration mismatch between devices. Verifying and correcting any username and password mismatch should remedy this problem.

```
Router# debug ppp chap

ppp: received conf.ig for type = 5 (MAGICNUMBER) value = 1E24718 acked
PPP BRI0: B-Channel 1: state = ACKSENT fsm_rconfack(C021): rcvd id E6
ppp: config ACK received, type = 3 (CI_AUTHTYPE), value = C223
ppp: config ACK received, type = 5 (CI_MAGICNUMBER), value = 28CEF76C
BRI0: B-Channel 1: PPP AUTH CHAP input code = 1 id = 83 len = 16
BRI0: B-Channel 1: PPP AUTH CHAP input code = 2 id = 96 len = 28
BRI0: B-Channel 1: PPP AUTH CHAP input code = 4 id = 83 len = 21
BRI0: B-Channel 1: Failed CHAP authentication with remote.
Remote message is: MD compare failed
```

If your call cannot successfully complete PPP negotiation, contact your support personnel.

Troubleshooting AAA Negotiation

This section first shows debug output of successful AAA negotiation. The subsequent sections explain several common misconfigurations that prevent successful AAA negotiation:

- Successful AAA Negotiation
- Incorrect User Password
- Error Contacting RADIUS Server
- Misconfigured AAA Authentication

Successful AAA Negotiation

Enable the **debug aaa authentication** and **debug aaa authorization** commands on the tunnel server and dial in to the NAS.

The following debug output shows successful AAA negotiation on the tunnel server. This output has been edited to exclude repetitive lines.

```
ENT_HGW#
Jan  7 19:29:44.132: AAA/AUTHEN: create_user (0x612D550C) user='ENT_HGW' ruser='
' port='' rem_addr='' authen_type=CHAP service=PPP priv=1
Jan  7 19:29:44.132: AAA/AUTHEN/START (384300079): port='' list='default' action
```

```
=SENDAUTH service=PPP
Jan  7 19:29:44.132: AAA/AUTHEN/START (384300079): found list default
Jan  7 19:29:44.132: AAA/AUTHEN/START (384300079): Method=LOCAL
Jan  7 19:29:44.132: AAA/AUTHEN (384300079): status = PASS
Jan  7 19:29:44.132: AAA/AUTHEN: create_user (0x612D550C) user='ISP_NAS' ruser='
' port='' rem_addr='' authen_type=CHAP service=PPP priv=1
Jan  7 19:29:44.132: AAA/AUTHEN/START (2545876944): port='' list='default' actio
n=SENDAUTH service=PPP
Jan  7 19:29:44.132: AAA/AUTHEN/START (2545876944): found list default
Jan  7 19:29:44.132: AAA/AUTHEN/START (2545876944): Method=LOCAL
Jan  7 19:29:44.132: AAA/AUTHEN (2545876944): status = PASS
Jan  7 19:29:44.228: AAA/AUTHEN: create_user (0x612F1F78) user='jeremy@hgw.com'
ruser='' port='Virtual-Access1' rem_addr='408/5550945' authen_type=CHAP service=
PPP priv=1
Jan  7 19:29:44.228: AAA/AUTHEN/START (101773535): port='Virtual-Access1' list=''
action=LOGIN service=PPP
Jan  7 19:29:44.228: AAA/AUTHEN/START (101773535): using "default" list
Jan  7 19:29:44.228: AAA/AUTHEN/START (101773535): Method=LOCAL
Jan  7 19:29:44.228: AAA/AUTHEN (101773535): status = ERROR
Jan  7 19:29:44.228: AAA/AUTHEN/START (101773535): Method=RADIUS
Jan  7 19:29:44.692: AAA/AUTHEN (101773535): status = PASS
Jan  7 19:29:44.692: Vi1 AAA/AUTHOR/LCP: Authorize LCP
Jan  7 19:29:44.692: AAA/AUTHOR/LCP Vi1 (3630870259): Port='Virtual-Access1' list=''
service=NET
Jan  7 19:29:44.692: AAA/AUTHOR/LCP: Vi1 (3630870259) user='jeremy@hgw.com'
Jan  7 19:29:44.692: AAA/AUTHOR/LCP: Vi1 (3630870259) send AV service=ppp
Jan  7 19:29:44.692: AAA/AUTHOR/LCP: Vi1 (3630870259) send AV protocol=lcp
Jan  7 19:29:44.692: AAA/AUTHOR/LCP (3630870259) found list "default"
Jan  7 19:29:44.692: AAA/AUTHOR/LCP: Vi1 (3630870259) Method=RADIUS
Jan  7 19:29:44.692: AAA/AUTHOR (3630870259): Post authorization status = PASS_REPL
Jan  7 19:29:44.696: Vi1 AAA/AUTHOR/FSM: We can start IPCP
6w5d: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
Jan  7 19:29:47.792: Vi1 AAA/AUTHOR/IPCP: Start.  Her address 0.0.0.0, we want 172.30.2.1
```

If the above debug output appears, but you still cannot ping the tunnel server, contact your support personnel and troubleshoot your network backbone.

If you did not see the debug output above, you need to troubleshoot AAA negotiation.

Incorrect User Password

If the user password is incorrect (or it is incorrectly configured), the tunnel will be established, but the tunnel server will not authenticate the user. If the user password is incorrect, the following debug output appears on the NAS and tunnel server when you dial in to the NAS and the **debug vpdn l2x-errors** and **debug vpdn l2x-events** commands are enabled:

```
ISP_NAS#
Jan  1 01:00:01.555: %LINK-3-UPDOWN: Interface Async12, changed state to up
Jan  1 01:00:05.299:  L2F: Tunnel state closed
Jan  1 01:00:05.299:  L2F: MID  state closed
Jan  1 01:00:05.299: L2F: Open UDP socket to 172.22.66.25
Jan  1 01:00:05.299:  L2F: Tunnel state opening
Jan  1 01:00:05.299: As12 L2F: MID jeremy@hgw.com state waiting_for_tunnel
Jan  1 01:00:05.303: L2F: L2F_CONF received
Jan  1 01:00:05.303: L2F: Removing resend packet (L2F_CONF)
Jan  1 01:00:05.303: ENT_HGW L2F: Tunnel state open
Jan  1 01:00:05.307: L2F: L2F_OPEN received
Jan  1 01:00:05.307: L2F: Removing resend packet (L2F_OPEN)
Jan  1 01:00:05.307: L2F: Building nas2gw_mid0
Jan  1 01:00:05.307: L2F: L2F_CLIENT_INFO: CLID/DNIS 4089548021/5550945
Jan  1 01:00:05.307: L2F: L2F_CLIENT_INFO: NAS-Port Async12
Jan  1 01:00:05.307: L2F: L2F_CLIENT_INFO: Client-Bandwidth-Kbps 115
Jan  1 01:00:05.307: L2F: L2F_CLIENT_INFO: NAS-Rate L2F/26400/28800
```

```
Jan  1 01:00:05.307: As12 L2F: MID jeremy@hgw.com state opening
Jan  1 01:00:05.307: L2F: Tunnel authentication succeeded for ENT_HGW
Jan  1 01:00:05.391: L2F: L2F_OPEN received
Jan  1 01:00:05.391: L2F: Got a MID management packet
Jan  1 01:00:05.391: L2F: Removing resend packet (L2F_OPEN)
Jan  1 01:00:05.391: As12 L2F: MID jeremy@hgw.com state open
Jan  1 01:00:05.391: As12 L2F: MID synced NAS/HG Clid=47/12 Mid=1
Jan  1 01:00:05.523: L2F: L2F_CLOSE received
Jan  1 01:00:05.523: %VPDN-6-AUTHENERR: L2F HGW ENT_HGW cannot locate a AAA server for
As12 user jeremy@hgw.com; Authentication failure

ENT_HGW#
Jan  1 01:00:05.302: L2F: L2F_CONF received
Jan  1 01:00:05.302: L2F: Creating new tunnel for ISP_NAS
Jan  1 01:00:05.302:  L2F: Tunnel state closed
Jan  1 01:00:05.302: L2F: Got a tunnel named ISP_NAS, responding
Jan  1 01:00:05.302: L2F: Open UDP socket to 172.22.66.23
Jan  1 01:00:05.302: ISP_NAS L2F: Tunnel state opening
Jan  1 01:00:05.306: L2F: L2F_OPEN received
Jan  1 01:00:05.306: L2F: Removing resend packet (L2F_CONF)
Jan  1 01:00:05.306: ISP_NAS L2F: Tunnel state open
Jan  1 01:00:05.306: L2F: Tunnel authentication succeeded for ISP_NAS
Jan  1 01:00:05.310: L2F: L2F_OPEN received
Jan  1 01:00:05.310: L2F: L2F_CLIENT_INFO: CLID/DNIS 4089548021/5550945
Jan  1 01:00:05.310: L2F: L2F_CLIENT_INFO: NAS-Port Async12
Jan  1 01:00:05.310: L2F: L2F_CLIENT_INFO: Client-Bandwidth-Kbps 115
Jan  1 01:00:05.310: L2F: L2F_CLIENT_INFO: NAS-Rate L2F/26400/28800
Jan  1 01:00:05.310: L2F: Got a MID management packet
Jan  1 01:00:05.310:  L2F: MID  state closed
Jan  1 01:00:05.310: L2F: Start create mid intf process for jeremy@hgw.com
5w6d: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
Jan  1 01:00:05.390: Vi1 L2X: Discarding packet because of no mid/session
Jan  1 01:00:05.390: Vi1 L2F: Transfer NAS-Rate L2F/26400/28800 to LCP
Jan  1 01:00:05.390: Vi1 L2F: Finish create mid intf for jeremy@hgw.com
Jan  1 01:00:05.390: Vi1 L2F: MID jeremy@hgw.com state open
5w6d: %VPDN-6-AUTHENERR: L2F HGW ENT_HGW cannot locate a AAA server for Vi1 user
jeremy@hgw.com; Authentication failure
```

Error Contacting RADIUS Server

If the **aaa authorization** command on the tunnel server is configured with the **default radius none** keywords, the tunnel server may allow unauthorized access to your network.

This command is an instruction to first use RADIUS for authorization. The tunnel server first contacts the RADIUS server (because of the **radius** keyword). If an error occurs when the tunnel server contacts the RADIUS server, the tunnel server does not authorize the user (because of the **none** keyword).

To see the following debug output, enable the **debug aaa authorization** command on the tunnel server and dial in to the NAS:

```
ENT_HGW#
*Feb  5 17:27:36.166: Vi1 AAA/AUTHOR/LCP: Authorize LCP
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP Vi1 (3192359105): Port='Virtual-Access1' list=''
service=NET
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP: Vi1 (3192359105) user='jeremy@hgw.com'
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP: Vi1 (3192359105) send AV service=ppp
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP: Vi1 (3192359105) send AV protocol=lcp
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP (3192359105) found list "default"
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP: Vi1 (3192359105) Method=RADIUS
*Feb  5 17:27:36.166: AAA/AUTHOR (3192359105): Post authorization status = ERROR
*Feb  5 17:27:36.166: AAA/AUTHOR/LCP: Vi1 (3192359105) Method=NONE
*Feb  5 17:27:36.166: AAA/AUTHOR (3192359105): Post authorization status = PASS_ADD
*Feb  5 17:27:36.166: Vi1 CHAP: O SUCCESS id 1 len 4
```

Caution Using the **none** keyword can allow unauthorized access to your network. Because of the risk of such errors occurring, we strongly recommend that you do not use the **none** keyword in your **aaa** commands.

Misconfigured AAA Authentication

If you reverse the order of the **local** and **radius** keywords in the **aaa authentication ppp** command on the tunnel server, the L2F tunnel cannot be established. The command should be configured as **aaa authentication ppp default local radius**.

If you configure the command as **aaa authentication ppp default radius local**, the tunnel server first tries to authenticate the L2F tunnel using RADIUS. The RADIUS server sends the following message to the tunnel server. To see this message, enable the **debug radius** command.

```
ENT_HGW#
Jan  1 01:34:47.827: RADIUS: SENDPASS not supported (action=4)
```

The RADIUS protocol does not support inbound challenges. This means that RADIUS is designed to authenticate user information, but it is not designed to be authenticated by others. When the tunnel server requests the tunnel secret from the RADIUS server, it responds with the "SENDPASS not supported" message.

To avoid this problem, use the **aaa authentication ppp default local radius** command on the tunnel server.

If your call still cannot successfully complete AAA negotiation, contact your support personnel.

Configuration Examples for VPN

This section provides the following configuration examples:

- Client-Initiated Dial-In Configuration Example
- VPN Tunnel Authentication Examples
- NAS Comprehensive Dial-In Configuration Example
- Tunnel Server Comprehensive Dial-in Configuration Example
- NAS Configured for Both Dial-In and Dial-Out Example
- Tunnel Server Configured for Both Dial-In and Dial-Out Example
- RADIUS Profile Examples
- TACACS+ Profile Examples

Client-Initiated Dial-In Configuration Example

The following example shows the running configuration of a tunnel server configured for PPTP using an ISA card to perform 40-bit MPPE encryption. It does not have an AAA configuration.

```
Current configuration
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
```

```
!
hostname PNS
!
no logging console guaranteed
enable password lab
!
username tester41 password 0 lab41
!
ip subnet-zero
no ip domain-lookup
!
vpdn enable
!
vpdn-group 1
! Default PPTP VPDN group
 accept-dialin
  protocol pptp
  virtual-template 1
 local name cisco_pns
!
memory check-interval 1
!
controller ISA 5/0
 encryption mppe
!
process-max-time 200
!
interface FastEthernet0/0
 ip address 10.1.1.12 255.255.255.0
 no ip directed-broadcast
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 10.1.2.12 255.255.255.0
 no ip directed-broadcast
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 no ip directed-broadcast
 shutdown
 framing c-bit
 cablelength 10
 dsu bandwidth 44210
!
interface Serial1/1
 no ip address
 no ip directed-broadcast
 shutdown
 framing c-bit
 cablelength 10
 dsu bandwidth 44210
!
interface FastEthernet4/0
 no ip address
 no ip directed-broadcast
 shutdown
 duplex half
!
interface Virtual-Template1
 ip unnumbered FastEthernet0/0
 no ip directed-broadcast
```

```
 ip mroute-cache
 no keepalive
 ppp encrypt mppe 40
 ppp authentication ms-chap
!
ip classless
ip route 172.29.1.129 255.255.255.255 1.1.1.1
ip route 172.29.63.9 255.255.255.255 1.1.1.1
no ip http server
!
line con 0
 exec-timeout 0 0
 transport input none
line aux 0
line vty 0 4
 login
!
end
```

VPN Tunnel Authentication Examples

The following examples shows several possibilities for performing local tunnel authentication. These examples only show the information relevant to tunnel authentication.

Tunnel Secret Configured Using the Local Name Command

The following examples are for a NAS and tunnel server that configure the tunnel names by using **local name** VPN group commands. The NAS tunnel name is ISP_NAS, the tunnel server tunnel name is ENT_HGW, and the tunnel secret is tunnelme.

NAS Configuration

The NAS tunnel name is specified by the **local name** command. The tunnel server tunnel name and tunnel secret are specified by the **username** command.

```
username ENT_HGW password 7 tunnelme
.
.
.
vpdn-group 1
 local name ISP_NAS
```

Tunnel Server Configuration

The tunnel server tunnel name is specified by the **local name** command. The NAS tunnel name and tunnel secret are specified by the **username** command.

```
username ISP_NAS password 7 tunnelme
.
.
.
vpdn-group 1
 local name ENT_HGW
```

Tunnel Secret Configured Using the L2TP Tunnel Password Command

The following example is for a NAS and tunnel server that both configure the tunnel secret using the **l2tp tunnel password** command. Because both routers use this command, they do not need to use either **username** or **local name** commands for tunnel authentication. The tunnel secret is tunnelme.

NAS Configuration

```
vpdn-group 1
 request-dialin
  protocol l2tp
 l2tp tunnel password tunnelme
```

Tunnel Server Configuration

```
vpdn-group 1
 accept-dialin
  protocol l2tp
 l2tp tunnel password tunnelme
```

Tunnel Secret Configuration Using Different Tunnel Authentication Methods

The follow example is for a NAS that uses the **username** command to specify the tunnel secret and a tunnel server that uses the **l2tp tunnel password** command to specify the tunnel secret.

NAS Configuration

```
username adrian password garf1eld
.
.
.
vpdn-group 1
 local name stella
```

Tunnel Server Configuration

```
vpdn-group 1
 accept--dialin
  protocol l2tp
 local name adrian
 l2tp tunnel password garf1eld
```

NAS Comprehensive Dial-In Configuration Example

The following example shows a NAS configured to tunnel PPP calls to a tunnel server using L2TP and local authentication and authorization:

```
! Enable AAA authentication and authorization with RADIUS as the default method
aaa new-model
aaa authentication ppp default radius
aaa authorization network default radius
!
username ISP_NAS password 7 tunnelme
username ENT_HGW password 7 tunnelme
!
vpdn enable
!
! Configure VPN to first search on the client domain name and then on the DNIS
vpdn search-order domain dnis
! Allow a maximum of 10 simultaneous VPN sessions
```

```
vpdn session-limit 10
!
! Configure VPN to initiate VPN dial-in sessions
vpdn-group 1
 request-dialin
! Specify L2TP as the tunneling protocol
  protocol l2TP
! Tunnel clients with the domain name "hgw.com"
  domain hgw.com
! Establish a tunnel with IP address 172.22.66.25
 initiate-to ip 172.22.66.25
! Identify the tunnel using the name "ISP_NAS"
 local name ISP_NAS
!
! Defines the ISDN switch type as primary-5ess
isdn switch-type primary-5ess
!
! Commissions the T1 controller to allow modem calls in to the NAS
controller T1 0
 framing esf
 clock source line primary
 linecode b8zs
 pri-group timeslots 1-24
!
interface Ethernet0
 ip address 172.22.66.23 255.255.255.192
!
! Configure the Serial channel to allow modem calls in to the NAS
interface Serial0:23
 no ip address
 isdn switch-type primary-5ess
 isdn incoming-voice modem
 no cdp enable
!
!
interface Group-Async1
 ip unnumbered Ethernet0
 encapsulation ppp
 async mode interactive
 no peer default ip address
 ppp authentication chap pap
group-range 1 96
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.22.66.1
!
! Specifies the RADIUS server IP address, authorization port, and accounting port
radius-server host 172.22.66.16 auth-port 1645 acct-port 1646
! Specifies the authentication key to be used with the RADIUS server
radius-server key cisco
!
line con 0
 transport input none
! Configures the modems
line 1 96
 autoselect during-login
 autoselect ppp
 modem InOut
line aux 0
line vty 0 4
!
end
```

Tunnel Server Comprehensive Dial-in Configuration Example

The following example show a tunnel server configured to accept L2TP tunnels from a NAS using local authentication and authorization:

```
aaa new-model
! Configure AAA to first use the local database and then contact the RADIUS server for
! PPP authentication
aaa authentication ppp default local radius
! Configure AAA network authorization and accounting by using the RADIUS server
aaa authorization network default radius
aaa accounting network default start-stop radius
!
username ISP_NAS password 7 tunnelme
username ENT_HGW password 7 tunnelme
!
vpdn enable
! Prevent any new VPN sessions from being established without disturbing existing
! sessions
vpdn softshut

!
! Configure VPN to accept dial-in sessions
vpdn-group 1
 accept-dialin
! Specify L2TP as the tunneling protocol
  protocol l2tp
! Specify that virtual-access interfaces be cloned from virtual template 1
  virtual-template 1
! Accept dial-in requests from a router using the tunnel name “ISP_NAS”
 terminate-from hostname ISP_NAS
! Identify the tunnel using the tunnel name “ENT_HGW”
 local name ENT_HGW
!
interface Ethernet0/0
 ip address 172.22.66.25 255.255.255.192
 no ip directed-broadcast
!
interface Virtual-Template1
! Use the IP address of interface Ethernet 0
 ip unnumbered Ethernet0
! Returns an IP address from the default pool to the VPN client
peer default ip address pool default
! Use CHAP to authenticate PPP
 ppp authentication chap
!
ip local pool default 172.30.2.1 172.30.2.96
ip classless
ip route 0.0.0.0 0.0.0.0 172.22.66.1
!
! Specifies the RADIUS server IP address, authorization port, and accounting port
radius-server host 172.22.66.13 auth-port 1645 acct-port 1646
! Specifies the authentication key to be used with the RADIUS server
radius-server key cisco
```

NAS Configured for Both Dial-In and Dial-Out Example

You can configure a NAS to simultaneously initiate L2TP or L2F dial-in tunnels to a tunnel server and also accept L2TP dial-out tunnels from a tunnel server.

In the following example, the VPN group of a NAS is configured to dial in using L2F and to dial out using L2TP as the tunneling protocol and dialer interface 2. The example only shows the VPN group and dialer configuration:

```
vpdn-group 1
 request-dialin
  protocol l2f
  domain jgb.com
 accept-dialout
  protocol l2tp
  dialer 2
 local name cerise
 terminate-from hostname reuben
 initiate-to ip 172.1.2.3
!
interface Dialer2
 ip unnumbered Ethernet0
 encapsulation ppp
 dialer in-band
 dialer aaa
 dialer-group 1
 ppp authentication chap
```

Tunnel Server Configured for Both Dial-In and Dial-Out Example

You can configure a tunnel server to simultaneously receive L2TP or L2F dial-in tunnels from a NAS and also initiate L2TP dial-out tunnels to a NAS.

In the following example, a tunnel server VPN group is configured to dial in using virtual template 1 to clone the virtual access interface and to dial out using dialer pool 1. The example only shows the VPN group and dialer configuration:

```
vpdn-group 1
 accept-dialin
  protocol l2tp
  virtual-template 1
 request-dialout
  protocol l2tp
  pool-member 1
 local name reuben
 terminate-from hostname cerise
 initiate-to ip 10.3.2.1
!
interface Dialer2
 ip address 172.19.2.3 255.255.128
 encapsulation ppp
 dialer remote-name reuben
 dialer string 5551234
 dialer vpdn
 dialer pool 1
 dialer-group 1
 ppp authentication chap
```

RADIUS Profile Examples

The following sections show VPN RADIUS profiles configured using CiscoSecure version 2.3.1:

- RADIUS Domain Profile
- RADIUS User Profile

RADIUS Domain Profile

The following example show a profile that is configured on the NAS RADIUS server to tunnel calls from users who dial-in with the domain name terrapin.com. The NAS will balance calls between the tunnel servers at 172.16.171.11 and 172.16.171.12. If both of those tunnel servers are unavailable, the NAS will tunnel calls to 172.16.171.13.

```
user = terrapin.com{
profile_id = 29
set server current-failed-logins = 0
profile_cycle = 7
radius=Cisco {
check_items= {
2=cisco
}
reply_attributes= {
9,1="vpdn:l2tp-tunnel-password=cisco123"
9,1="vpdn:tunnel-type=l2tp"
9,1="vpdn:ip-addresses=172.16.171.11 172.16.171.12/172.16.171.13"
9,1="vpdn:tunnel-id=tunnel"
}
}
}
```

Note check_items={2=cisco} is a hard-coded password. This password must be "cisco."

RADIUS User Profile

The following example shows a profile that is configured on the tunnel server RADIUS server to authorize and authenticate user sailor@terrapin.com:

```
user = sailor@terrapin.com{
profile_id = 28
profile_cycle = 2
radius=Cisco {
check_items= {
2=cisco
}
reply_attributes= {
6=2
7=1
}
}
}
```

Note check_items={2=cisco} is a hard-coded password. This password must be "cisco."

TACACS+ Profile Examples

The following sections show VPN TACACS+ profiles configured using CiscoSecure version 2.2.2:

- TACACS+ Domain Profile
- TACACS+ User Profile
- TACACS+ Tunnel Profiles

TACACS+ Domain Profile

The following example shows a profile that is configured on the NAS TACACS+ server to tunnel users who dial in with the domain name guava.com:

```
user = guava.com{
profile_id = 83
profile_cycle = 1
service=ppp {
protocol=vpdn {
set tunnel-id=isp
set ip-addresses="10.31.1.50"
set nas-password="little"
set gw-password="birdies"
}
protocol=lcp {
}
}
}
```

TACACS+ User Profile

The following example shows a profile that is configured on the tunnel server TACACS+ to authorize and authenticate user geaner@guava.com:

```
user = geaner@guava.com{
profile_id = 85
profile_cycle = 1
password = chap "daisies"
service=ppp {
protocol=ip {
default attribute=permit
}
protocol=lcp {
}
}
}
```

TACACS+ Tunnel Profiles

The following examples show a profile that is configured on the tunnel server TACACS+ server to authenticate the tunnel. See the "Configuring VPN Tunnel Authentication Using the Host Name or Local Name" and "Configuring VPN Tunnel Authentication Using the L2TP Tunnel Password" sections earlier in this chapter for more information on tunnel authentication.

Note Only the tunnel server AAA server can perform tunnel authentication. Tunnel authentication must be performed locally by the NAS.

```
user = tunnel-server {
profile_id = 82
profile_cycle = 1
password = chap "3stone"
service=ppp {
protocol=ip {
default attribute=permit
}
protocol=lcp {
}
```

```
}
}
```