VPDN Tunnel Management

This module contains information about managing virtual private dialup network (VPDN) tunnels and monitoring VPDN events. The tasks documented in this module should be performed only after configuring and deploying a VPDN.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for VPDN Tunnel Management

Before you can perform the tasks in this module, you must configure a VPDN deployment. For an overview of VPDN deployments, see the VPDN Technology Overview module.

Restrictions for VPDN Tunnel Management

VPDN tunnels using the Layer 2 Forwarding (L2F) protocol or Point-to-Point Tunnel Protocol (PPTP) are not supported.

Information About VPDN Tunnel Management

Termination of VPDN Tunnels

VPDN tunnels can be terminated manually or through a soft shutdown. Manual termination of a VPDN tunnel results in the immediate shutdown of the specified VPDN tunnel and all sessions within that tunnel, resulting in a sudden disruption of VPDN services. Enabling soft shutdown on a router prevents the establishment of new VPDN sessions in all VPDN tunnels that terminate on that router, but does not affect existing sessions. Opting to terminate a VPDN tunnel by enabling soft shutdown prevents the disruption of established sessions that occurs when a VPDN tunnel is manually terminated.

VPDN Session Limits

The number of simultaneous VPDN sessions that can be established on a router can be manually configured, providing network administrators more control over the network. VPDN session limits can increase performance and reduce latency for routers that are otherwise forced to operate at high capacity.

The maximum number of VPDN sessions can be configured globally, at the level of a VPDN group, or for all VPDN groups associated with a particular VPDN template.

The hierarchy for the application of VPDN session limits is as follows:

- Globally configured session limits take precedence over session limits configured for a VPDN group or in a VPDN template. The total number of sessions on a router cannot exceed a configured global session limit.
- Session limits configured for a VPDN template are enforced for all VPDN groups associated with that VPDN template. The total number of sessions for all of the associated VPDN groups cannot exceed the configured VPDN template session limit.
- Session limits configured for a VPDN group are enforced for that VPDN group.

Control Packet Parameters for VPDN Tunnels

Certain control packet timers, retry counters, and the advertised control packet receive window size can be configured for Layer 2 Transport Protocol (L2TP) or Layer 2 Forwarding (L2F) VPDN tunnels. Adjustments to these parameters allow fine-tuning of router performance to suit the particular needs of the VPDN deployment.

L2TP Congestion Avoidance

L2TP congestion avoidance provides packet flow control and congestion avoidance by throttling L2TP control messages as described in RFC 2661. Throttling L2TP control message packets prevents input buffer overflows on the peer tunnel endpoint, which can result in dropped sessions.

Before the introduction of L2TP congestion avoidance, the window size used to send packets between the network access server (NAS) and the tunnel server was set to the value advertised by the peer endpoint and was never changed. Configuring L2TP congestion avoidance allows the L2TP packet window to be dynamically

resized using a sliding window mechanism. The window size grows larger when packets are delivered successfully, and is reduced when dropped packets must be retransmitted.

L2TP congestion avoidance is useful in networks with a relatively high rate of calls being placed by either tunnel endpoint. L2TP congestion avoidance is also useful on highly scalable platforms that support many simultaneous sessions.

How L2TP Congestion Avoidance Works

TCP/IP and RFC 2661 define two algorithms--slow start and congestion avoidance--used to throttle control message traffic between a NAS and a tunnel server. Slow start and congestion avoidance are two independent algorithms that work together to control congestion. Slow start and congestion avoidance require that two variables, a slow start threshold (SSTHRESH) size and a congestion window (CWND) size, be maintained by the sending device for each connection.

The congestion window defines the number of packets that can be transmitted before the sender must wait for an acknowledgment from its peer. The size of the congestion window expands and contracts, but can never exceed the size of the peer device's advertised receive window.

The slow start threshold defines the point at which the sending device switches operation from slow start mode to congestion avoidance mode. When the congestion window size is smaller than the slow start threshold, the device operates in slow start mode. When the congestion window size equals the slow start threshold, the device switches to congestion avoidance mode.

When a new connection is established, the sending device initially operates in slow start mode. The congestion window size is initialized to one packet, and the slow start threshold is set to the receive window size advertised by the peer tunnel endpoint (the receiving side).

The sending device begins by transmitting one packet and waiting for it to be acknowledged. When the acknowledgment is received, the congestion window size is incremented from one to two, and two packets can be sent. When those two packets are each acknowledged, the congestion window is increased to four. The congestion window doubles for each complete round trip, resulting in an exponential increase in size.

When the congestion window size reaches the slow start threshold value, the sending device switches over to operate in congestion avoidance mode. Congestion avoidance mode slows down the rate at which the congestion window size grows. In congestion avoidance mode, for every acknowledgment received the congestion window increases at the rate of 1 divided by the congestion window size. This results in linear, rather than exponential, growth of the congestion window size.

At some point, the capacity of the peer device will be exceeded and packets will be dropped. This indicates to the sending device that the congestion window has grown too large. When a retransmission event is detected, the slow start threshold value is reset to half of the current congestion window size, the congestion window size is reset to one, and the device switches operation to slow start mode (if it was not already operating in that mode).

VPDN Event Logging

There are two types of VPDN event logging available, VPDN failure event logging and generic VPDN event logging. The logging of VPDN failure events is enabled by default. Generic VPDN event logging is disabled by default, and must be explicitly enabled before generic event messages can be viewed.

How to Manage VPDN Tunnels

Manually Terminating VPDN Tunnels

Manual termination of a VPDN tunnel results in the immediate shutdown of the specified VPDN tunnel and all sessions within that tunnel, resulting in a sudden disruption of VPDN services. Before manually terminating a VPDN tunnel, consider performing the task in the Enabling Soft Shutdown of VPDN Tunnels, on page 5 instead.

A manually terminated VPDN tunnel can be restarted immediately when a user logs in. Manually terminating and restarting a VPDN tunnel while VPDN event logging is enabled can provide useful troubleshooting information about VPDN session establishment.

Perform this task to manually shut down a specific VPDN tunnel, resulting in the termination of the tunnel and all sessions in that tunnel. You can perform this task on these devices:

- The tunnel server
- The NAS when it is functioning as a tunnel endpoint

Note

- For client-initiated L2TP tunnels, you can perform this task only on the tunnel server.
- Tunnels using the L2F protocol and PPTP are not supported.

SUMMARY STEPS

1. **enable**
2. **clear vpdn tunnel l2tp** {**all** | **hostname** *remote-name* [*local-name*] | **id** *local-id* | **ip** *local-ip-address* | **ip** *remote-ip-address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**clear vpdn tunnel l2tp** {**all**	**hostname** *remote-name* [*local-name*]	**id** *local-id*	**ip** *local-ip-address*	**ip** *remote-ip-address*} **Example:** `Router# clear vpdn tunnel l2tp all`	Shuts down a specified tunnel and all sessions within the tunnel.

Enabling Soft Shutdown of VPDN Tunnels

Enabling soft shutdown of VPDN tunnels on a router prevents the establishment of new VPDN sessions in all VPDN tunnels that terminate on that router, but does not affect existing sessions. Opting to terminate a VPDN tunnel by enabling soft shutdown prevents the disruption of established sessions that occurs when a VPDN tunnel is manually terminated. Enabling soft shutdown on a router or access server will affect all of the tunnels terminating on that device. There is no way to enable soft shutdown for a specific tunnel. If you want to shut down a specific tunnel on a device without affecting any other tunnels, see the Manually Terminating VPDN Tunnels, on page 4 instead.

When soft shutdown is performed on a NAS, the potential session will be authorized before it is refused. This authorization ensures that accurate accounting records can be kept.

When soft shutdown is performed on a tunnel server, the reason for the session refusal will be returned to the NAS. This information is recorded in the VPDN history failure table.

Note Enabling soft shutdown of VPDN tunnels does not affect the establishment of Multichassis Multilink PPP (MMP) tunnels.

Perform this task to prevent new sessions from being established in any VPDN tunnel terminating on the router without disturbing service for existing sessions. You can perform this task on these devices:

- The tunnel server
- The NAS when it is functioning as a tunnel endpoint

Note
- For client-initiated L2TP tunnels, you can perform this task only on the tunnel server.
- Enabling soft shutdown of VPDN tunnels will not prevent new MMP sessions from being established.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vpdn softshut**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**vpdn softshut** **Example:** `Router(config)# vpdn softshut`	Prevents new sessions from being established on a VPDN tunnel without disturbing existing sessions.

Verifying the Soft Shutdown of VPDN Tunnels

Perform this task to ensure that soft shutdown is working properly.

SUMMARY STEPS

1. Establish a VPDN session by dialing in to the NAS using an allowed username and password.
2. **enable**
3. **configure terminal**
4. **vpdn softshut**
5. **exit**
6. **show vpdn**
7. Attempt to establish a new VPDN session by dialing in to the NAS using a second allowed username and password.
8. **show vpdn history failure**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Establish a VPDN session by dialing in to the NAS using an allowed username and password.	
Step 2	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 3	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 4	**vpdn softshut** **Example:** `Router(config)# vpdn softshut`	Prevents new sessions from being established on a VPDN tunnel without disturbing existing sessions. You can issue this command on either the NAS or the tunnel server.
Step 5	**exit** **Example:** `Router(config)# exit`	Exits to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show vpdn** **Example:** `Router# show vpdn`	Displays information about active L2TP or L2F tunnels and message identifiers in a VPDN. Issue this command to verify that the original session is active:
Step 7	Attempt to establish a new VPDN session by dialing in to the NAS using a second allowed username and password.	If soft shutdown has been enabled, a system logging (syslog) message appears on the console of the soft shutdown router.
Step 8	**show vpdn history failure** **Example:** `Router# show vpdn history failure`	Displays the content of the history failure table.

Limiting the Number of Allowed Simultaneous VPDN Sessions

The number of simultaneous VPDN sessions that can be established on a router can be manually configured, providing network administrators more control over the network. VPDN session limits can increase performance and reduce latency for routers that are otherwise forced to operate at high capacity.

The maximum number of VPDN sessions can be configured globally, at the level of a VPDN group, or for all VPDN groups associated with a particular VPDN template.

The hierarchy for the application of VPDN session limits is as follows:

- Globally configured session limits take precedence over session limits configured for a VPDN group or in a VPDN template. The total number of sessions on a router cannot exceed a configured global session limit.
- Session limits configured for a VPDN template are enforced for all VPDN groups associated with that VPDN template. The total number of sessions for all of the associated VPDN groups cannot exceed the configured VPDN template session limit.
- Session limits configured for a VPDN group are enforced for that VPDN group.

For an example of the interactions of global, template-level, and group-level VPDN session limits, see the "Examples Configuring VPDN Session Limits" section.

Perform any or all of the following optional tasks to configure VPDN session limits:

You can perform these tasks on the NAS or the tunnel server.

Restrictions

For client-initiated L2TP tunnels, you can perform these tasks only on the tunnel server.

Configuring Global VPDN Session Limits

Perform this task to limit the total number of VPDN sessions allowed on the router.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **vpdn session-limit** *sessions*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vpdn session-limit** *sessions* **Example:** `Router(config)# vpdn session-limit 6`	Limits the number of simultaneous VPDN sessions globally on the router.

Configuring VPDN Session Limits in a VPDN Template

Perform this task to configure a session limit in a VPDN template. The session limit is applied across all VPDN groups associated with the VPDN template.

Before you begin

- A VPDN template must be configured. See the "Creating a VPDN Template" section in the "Configuring Additional VPDN Features" module.
- If you configure a named VPDN template, you must associate the desired VPDN groups with the VPDN template. See the "Associating a VPDN Group with a VPDN Template" section in the "Configuring Additional VPDN Features" module.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vpdn-template** [*name*]
4. **group session-limit** *sessions*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vpdn-template** [*name*] **Example:** `Router(config)# vpdn-template l2tp`	Creates a VPDN template and enters VPDN template configuration mode.
Step 4	**group session-limit** *sessions* **Example:** `Router(config-vpdn-templ)# group session-limit 6`	Specifies the maximum number of concurrent sessions allowed across all VPDN groups associated with a particular VPDN template.

Configuring Session Limits for a VPDN Group

Perform this task to limit the number of VPDN sessions at the VPDN group level.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vpdn-group** *name*
4. **session-limit** *number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vpdn-group** *name* **Example:** `Router(config)# vpdn-group 1`	Creates a VPDN group and enters VPDN group configuration mode.

	Command or Action	Purpose
Step 4	**session-limit** *number* **Example:** `Router(config-vpdn)# session-limit 2`	Limits the number of sessions that are allowed through a specified VPDN group.

Verifying VPDN Session Limits

Perform this task to ensure that VPDN sessions are being limited properly.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vpdn session-limit** *sessions*
4. Establish a VPDN session by dialing in to the NAS using an allowed username and password.
5. Attempt to establish a new VPDN session by dialing in to the NAS using a second allowed username and password.
6. **exit**
7. **show vpdn history failure**

DETAILED STEPS

Step 1 **enable**

Enter this command to enable privileged EXEC mode. Enter your password if prompted:

Example:

```
Router> enable
```

Step 2 **configure terminal**

Enters global configuration mode.

Example:

```
Router# configure terminal
```

Step 3 **vpdn session-limit** *sessions*

Limits the number of simultaneous VPDN sessions on the router to the number specified with the *sessions* argument.

Issue this command on either the NAS or the tunnel server.

Example:

```
Router(config)# vpdn session-limit 1
```

Step 4 Establish a VPDN session by dialing in to the NAS using an allowed username and password.

Step 5 Attempt to establish a new VPDN session by dialing in to the NAS using a second allowed username and password.

If VPDN session limits have been configured properly, this session will be refused and a syslog message similar to the following should appear on the console of the router:

Example:

```
00:11:17:%VPDN-6-MAX_SESS_EXCD:L2F HGW tunnelserver1 has exceeded configured local session-limit and
 rejected user user2@cisco.com
```

Step 6 **exit**

Exits to privileged EXEC mode.

Step 7 **show vpdn history failure**

Shows the content of the history failure table.

Example:

```
Router# show vpdn history failure
User:user2@scisco.com
 NAS:NAS1, IP address = 172.25.52.8, CLID = 2
 Gateway:tunnelserver1, 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.
!This output shows that the configured session limit is being properly applied.
 Failure reason:
```

Configuring L2TP Control Packet Parameters for VPDN Tunnels

Control packet timers, retry counters, and the advertised control packet receive window size can be configured for L2TP VPDN tunnels. Adjustments to these parameters allow fine-tuning of router performance to suit the particular needs of the VPDN deployment.

Perform this task to configure control packet parameters if your VPDN configuration uses L2TP tunnels. The configuration of each parameter is optional. If a parameter is not manually configured, the default value will be used.

You can perform this task on these devices:

- The tunnel server
- The NAS when it is functioning as a tunnel endpoint

Before you begin

Load balancing must be enabled for the configuration of the **l2tp tunnel retransmit initial timeout** command or the **l2tp tunnel retransmit initial retries** command to have any effect.

Note For client-initiated L2TP tunnels, you can perform this task only on the tunnel server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vpdn-group** *name*
4. **l2tp tunnel hello** *seconds*
5. **l2tp tunnel receive window** *packets*
6. **l2tp tunnel retransmit retries** *number*
7. **l2tp tunnel retransmit timeout** {**min** | **max**} *seconds*
8. **l2tp tunnel timeout no-session** {*seconds* | **never**}
9. **l2tp tunnel timeout setup** *seconds*
10. **l2tp tunnel zlb delay** *seconds*
11. **l2tp tunnel retransmit initial timeout** {**min** | **max**} *seconds*
12. **l2tp tunnel retransmit initial retries** *number*
13. **l2tp tunnel busy timeout** *seconds*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vpdn-group** *name* **Example:** `Router(config)# vpdn-group group1`	Creates a VPDN group and enters VPDN group configuration mode.
Step 4	**l2tp tunnel hello** *seconds* **Example:** `Router(config-vpdn)# l2tp tunnel hello 90`	(Optional) Set the number of seconds between sending hello keepalive packets for an L2TP tunnel. • *seconds* --Time, in seconds, that the NAS and tunnel server will wait before sending the next L2TP tunnel keepalive packet. Valid values range from 0 to 1000. The default value is 60.

	Command or Action	Purpose
Step 5	**l2tp tunnel receive window** *packets* **Example:** `Router(config-vpdn)# l2tp tunnel receive window 500`	(Optional) Configures the number of packets allowed in the local receive window for an L2TP control channel. • *packets* --Number of packets allowed in the receive window. Valid values range from 1 to 5000. The default value varies by platform.
Step 6	**l2tp tunnel retransmit retries** *number* **Example:** `Router(config-vpdn)# l2tp tunnel retransmit retries 8`	(Optional) Configures the number of retransmission attempts made for an L2TP control packet. • *number* --Number of retransmission attempts. Valid values range from 5 to 1000. The default value is 10.
Step 7	**l2tp tunnel retransmit timeout** {**min** | **max**} *seconds* **Example:** `Router(config-vpdn)# l2tp tunnel retransmit timeout max 4`	(Optional) Configures the amount of time that the router will wait before resending an L2TP control packet. • **min** --Specifies the minimum time that the router will wait before resending a control packet. • **max** --Specifies the maximum time that the router will wait before resending a control packet. • *seconds* --Timeout length, in seconds, the router will wait before resending a control packet. Valid values range from 1 to 8. The default minimum value is 1. The default maximum value is 8.
Step 8	**l2tp tunnel timeout no-session** {*seconds* | **never**} **Example:** `Router(config-vpdn)# l2tp tunnel timeout no-session never`	(Optional) Configures the time a router waits after an L2TP tunnel becomes empty before tearing down the tunnel. • *seconds* --Time, in seconds, the router will wait before tearing down an empty L2TP tunnel. Valid values range from 0 to 86400. If the router is configured as a NAS, the default is 15 seconds. If the router is configured as a tunnel server, the default is 10. • **never** --Specifies that the router will never tear down an empty L2TP tunnel.
Step 9	**l2tp tunnel timeout setup** *seconds* **Example:** `Router(config-vpdn)# l2tp tunnel timeout setup 25`	(Optional) Configures the amount of time that the router will wait for a confirmation message after sending out the initial L2TP control packet before considering a peer busy. • *seconds* --Time, in seconds, the router will wait for a confirmation message. Valid values range from 60 to 6000. The default value is 10.
Step 10	**l2tp tunnel zlb delay** *seconds* **Example:** `Router(config-vpdn)# l2tp tunnel zlb delay 2`	(Optional) Configures the delay time before a zero length bit (ZLB) control message must be acknowledged. • *seconds* --Maximum number of seconds the router will delay before acknowledging ZLB control

	Command or Action	Purpose
		messages. Valid values range from 1 to 5. The default value is 3.
Step 11	**l2tp tunnel retransmit initial timeout** {**min** | **max**} *seconds* **Example:** `Router(config-vpdn)# l2tp tunnel retransmit initial timeout min 2`	(Optional) Sets the amount of time, in seconds, that the router will wait before resending an initial packet out to establish a tunnel. • **min** --Specifies the minimum time that the router will wait before resending an initial packet. • **max** --Specifies the maximum time that the router will wait before resending an initial packet. • *seconds* --Timeout length, in seconds, the router will wait before resending an initial packet. Valid values range from 1 to 8. The default minimum value is 1. The default maximum value is 8. **Note** Load balancing must be configured for the retry counter configured with the **l2tp tunnel retransmit initial timeout** command to take effect.
Step 12	**l2tp tunnel retransmit initial retries** *number* **Example:** `Router(config-vpdn)# l2tp tunnel retransmit initial retries 5`	(Optional--Cisco IOS Release 12.2(4)T, Cisco IOS Release 12.2(28)SB, or a later release) Sets the number of times that the router will attempt to send out the initial control packet for tunnel establishment before considering a router busy. • *number* --Number of retransmission attempts. Valid values range from 1 to 1000. The default value is 2. **Note** Load balancing must be configured for the retry counter configured with the **l2tp tunnel retransmit initial retries** command to take effect.
Step 13	**l2tp tunnel busy timeout** *seconds* **Example:** `Router(config-vpdn)# l2tp tunnel busy timeout 90`	(Optional) Configures the amount of time, in seconds, that the router will wait before attempting to recontact a router that was previously busy. • *seconds* --Time, in seconds, the router will wait before checking for router availability. Valid values range from 60 to 6000. The default value is 300.

Configuring L2TP Congestion Avoidance

Perform this task to configure L2TP congestion avoidance on a tunnel endpoint, allowing dynamic throttling of the L2TP control packet window size.

You can perform this task on these devices:

- The tunnel server
- The NAS when it is functioning as a tunnel endpoint

This task need be performed only on the sending device.

Note

- This task is compatible only with VPDN deployments that use the L2TP tunneling protocol.
- For client-initiated L2TP tunnels, you can perform this task only on the tunnel server.
- The congestion window size cannot exceed the size of the advertised receive window set by the **l2tp tunnel receive-window** command on the peer device. To configure the advertised receive window on the remote peer device, see the Configuring L2TP Control Packet Parameters for VPDN Tunnels, on page 11.
- L2TP congestion avoidance is enabled (or disabled) only for those tunnels that are established after the configuration has been applied. Tunnels that already exist when the **l2tp congestion-control** command is issued are not affected by the command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **l2tp congestion-control**
4. **exit**
5. **show vpdn tunnel l2tp all**
6. **debug vpdn l2x-events**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**l2tp congestion-control** **Example:** `Router(config)# l2tp congestion-control`	Enables L2TP congestion avoidance.
Step 4	**exit** **Example:**	Exits to privileged EXEC mode.

	Command or Action	Purpose
	`Router(config)# exit`	
Step 5	**show vpdn tunnel l2tp all** **Example:** `Router# show vpdn tunnel l2tp all`	Displays information about all active L2TP VPDN tunnels.
Step 6	**debug vpdn l2x-events** **Example:** `Router(config)# debug vpdn l2x-events`	Displays troubleshooting information for protocol-specific VPDN tunneling events.

Configuring VPDN Failure Event Logging

Logging of a failure event to the history table is triggered by event logging by the syslog facility. The syslog facility creates a history failure table, which keeps records of failure events. The table defaults to a maximum of 20 entries, but the size of the table can be configured to retain up to 50 entries.

Failure entries are kept chronologically in the history table. Each entry records the relevant information of a failure event. Only the most recent failure event per user, unique to its name and tunnel client ID (CLID), is kept. When the total number of entries in the table reaches the configured maximum table size, the oldest record is deleted and a new entry is added.

The logging of VPDN failure events to the VPDN history failure table is enabled by default. You need enable VPDN failure event logging only if it has been previously disabled. Perform this task to enable VPDN failure event logging, to configure the maximum number of entries the history failure table can hold, and to display and clear the contents of the VPDN history failure table.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vpdn history failure**
4. **vpdn history failure table-size** *entries*
5. **exit**
6. **show vpdn history failure**
7. **clear vpdn history failure**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vpdn history failure** **Example:** `Router(config)# vpdn history failure`	(Optional) Enables logging of VPDN failure events to the history failure table. **Note** VPDN history failure logging is enabled by default. You need issue the **vpdn history failure** command only if you have previously disabled VPDN history failure logging using the **no vpdn history failure** command.
Step 4	**vpdn history failure table-size** *entries* **Example:** `Router(config)# vpdn history failure table-size 50`	(Optional) Sets the history failure table size. **Note** The VPDN history failure table size can be configured only when VPDN failure event logging is enabled using the **vpdn history failure** command.
Step 5	**exit** **Example:** `Router# exit`	Exits to privileged EXEC mode.
Step 6	**show vpdn history failure** **Example:** `Router# show vpdn history failure`	(Optional) Displays the contents of the history failure table.
Step 7	**clear vpdn history failure** **Example:** `Router# clear vpdn history failure`	(Optional) Clears the contents of the history failure table.

Enabling Generic VPDN Event Logging

Generic VPDN events are a mixture of error, warning, notification, and information reports logged by the syslog facility. When VPDN event logging is enabled locally or at a remote tunnel endpoint, VPDN event messages are printed to the console as the events occur. VPDN event messages can also be reported to a remote authentication, authorization, and accounting (AAA) server in a AAA vendor-specific attribute (VSA), allowing the correlation of VPDN call success rates with accounting records.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **vpdn logging** [**accounting** | **local** | **remote** | **tunnel-drop** | **user**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vpdn logging** [**accounting**	**local**	**remote**	**tunnel-drop**	**user**] **Example:** `Router(config)# vpdn logging remote`	(Optional) Enables the logging of generic VPDN events. • You can configure as many types of generic VPDN event logging as you want by issuing multiple instances of the **vpdn logging** command. **Note** The reporting of VPDN event log messages to a AAA server can be enabled independently of all other generic VPDN event logging configurations.

Configuration Examples for VPDN Tunnel Management

Examples Manually Terminating VPDN Tunnels

The following example manually terminates all L2TP tunnels that terminate on the router:

```
Router# clear vpdn tunnel l2tp all
```

Example Enabling Soft Shutdown of VPDN Tunnels

The following example enables soft shutdown of all VPDN tunnels that terminate on the device that the command is issue on:

```
Router# configure terminal
Router(config)# vpdn softshut
!The following syslog message will appear on the device whenever an attempt is made to
!establish a new VPDN session after soft shutdown is enabled.
!
00:11:17:%VPDN-6-SOFTSHUT:L2TP HGW tunnelserver1 has turned on softshut and rejected user
user2@cisco.com
```

Examples Configuring VPDN Session Limits

The following example configures a VPDN group named customer7 with a group-level session limit of 25. No more than 25 sessions can be associated with this VPDN group.

```
Router(config)# vpdn-group customer7
Router(config-vpdn)# session-limit 25
```

A VPDN template named customer4 is then created, and a session limit of 8 is configured at the VPDN template level. Two VPDN groups are associated with the VPDN template, each with a VPDN group-level session limit of 5.

```
Router(config)# vpdn-template customer4
Router(config-vpdn-templ)# group session-limit 8
!
Router(config)# vpdn-group customer4_l2tp
Router(config-vpdn)# source vpdn-template customer4
Router(config-vpdn)# session-limit 5
!
Router(config)# vpdn-group customer4_l2f
Router(config-vpdn)# source vpdn-template customer4
Router(config-vpdn)# session-limit 5
```

With this configuration, if the VPDN group named customer4_l2tp has 5 active sessions, the VPDN group named customer4_l2f can establish only 3 sessions. The VPDN group named customer7 can still have up to 25 active sessions.

If a global limit of 16 VPDN sessions is also configured, the global limit takes precedence over the configured VPDN group and VPDN template session limits:

```
Router# configure terminal
Router(config)# vpdn session-limit 16
```

The three VPDN groups will be able to establish a total of 16 sessions between them. For example, if the VPDN group named customer4_l2tp has the maximum allowable number of active sessions (5 sessions), and the VPDN group named customer4_l2f has 2 active sessions, the VPDN group named customer7 can establish only up to 9 sessions.

Example Verifying Session Limits for a VPDN Group

The following example creates the VPDN group named l2tp and restricts it to three sessions. The configured session limit is displayed when the **show vpdn group** command is issued.

```
Router# configure terminal
Router(config)# vpdn-group l2tp
Router(config-vpdn)# accept dialin
Router(config-vpdn-acc-in)# protocol l2tp
Router(config-vpdn-acc-in)# virtual-template 5
Router(config-vpdn-acc-in)# exit
Router(config-vpdn)# terminate-from hostname host1
Router(config-vpdn)# session-limit 3
Router(config-vpdn)# end
Router# show vpdn group l2tp
Tunnel (L2TP)
------
dnis:cg1
```

```
dnis:cg2
dnis:jan
cisco.com
Endpoint        Session Limit Priority Active Sessions Status Reserved Sessions
--------        ------------- -------- --------------- ------ -----------------
172.21.9.67     3             1        0               OK     -
--------------- ------------              ---------------        -----------------
Total           *                         0                      0
```

Example Configuring L2TP Control Packet Timers and Retry Counters for VPDN Tunnels

The following example configures custom values for all of the available L2TP control packet parameters for the VPDN group named l2tp:

```
Router# configure terminal

Router(config)# vpdn-group l2tp

Router(config-vpdn)# l2tp tunnel hello 90
Router(config-vpdn)# l2tp tunnel receive window 500
Router(config-vpdn)# l2tp tunnel retransmit retries 8
Router(config-vpdn)# l2tp tunnel retransmit timeout min 2
Router(config-vpdn)# l2tp tunnel timeout no-session 500
Router(config-vpdn)# l2tp tunnel timeout setup 25
Router(config-vpdn)# l2tp tunnel zlb delay 4
Router(config-vpdn)# l2tp tunnel retransmit initial timeout min 2
Router(config-vpdn)# l2tp tunnel retransmit initial retries 5
Router(config-vpdn)# l2tp tunnel busy timeout 90
```

Example Configuring Verifying and Debugging L2TP Congestion Avoidance

The following example configures a basic dial-in L2TP VPDN tunnel, sets the receive window size to 500 on the tunnel server (the receiving device), and enables L2TP congestion avoidance on the NAS (the sending device):

Tunnel Server Configuration

```
Router(config)# vpdn enable
!
Router(config)# vpdn-group 1
Router(config-vpdn)# accept-dialin
Router(config-vpdn-acc-in)# protocol l2tp
Router(config-vpdn-acc-in)# virtual-template 1
!
Router(config-vpdn)# terminate from hostname NAS1
Router(config-vpdn)# l2tp tunnel receive-window 500
```

NAS Configuration

```
Router(config)# vpdn enable
!
Router(config)# vpdn-group 1
```

```
Router(config-vpdn)# request-dialin
Router(config-vpdn-req-in)# protocol l2tp
Router(config-vpdn-req-in)# domain cisco.com
!
Router(config-vpdn)# initiate-to ip 172.22.66.25
Router(config-vpdn)# local name NAS1
!
Router(config)# l2tp congestion-control
```

The following example shows L2TP tunnel activity, including the information that L2TP congestion control is enabled. Note that the slow start threshold is set to the same size as the remote receive window size. The Remote RWS value advertised by the remote peer is shown in the Remote RWS field. When the actual RWS value differs from the advertised value, the actual RWS value will be displayed as *In Use Remote RWS <value>*.

```
Router# show vpdn tunnel l2tp all
L2TP Tunnel Information Total tunnels 1 sessions 1
Tunnel id 30597 is up, remote id is 45078, 1 active sessions
  Tunnel state is established, time since change 00:08:27
  Tunnel transport is UDP (17)
  Remote tunnel name is LAC1
    Internet Address 172.18.184.230, port 1701
  Local tunnel name is LNS1
    Internet Address 172.18.184.231, port 1701
  Tunnel domain unknown
  VPDN group for tunnel is 1
  L2TP class for tunnel is
  4 packets sent, 3 received
  194 bytes sent, 42 received
  Last clearing of "show vpdn" counters never
  Control Ns 2, Nr 4
  Local RWS 1024 (default), Remote RWS 256
  In Use Remote RWS 15
  Control channel Congestion Control is enabled
    Congestion Window size, Cwnd 3
    Slow Start threshold, Ssthresh 256
    Mode of operation is Slow Start
  Tunnel PMTU checking disabled
  Retransmission time 1, max 2 seconds
  Unsent queuesize 0, max 0
  Resend queuesize 0, max 1
  Total resends 0, ZLB ACKs sent 2
  Current nosession queue check 0 of 5
  Retransmit time distribution: 0 0 0 0 0 0 0 0 0
  Sessions disconnected due to lack of resources 0
  Control message authentication is disabled
```

The following partial output from the **debug vpdn l2x-events** command shows that congestion occurred. The congestion window size and the slow start threshold have been reset due to a packet retransmission event.

```
Router# debug vpdn l2x-events
!
*Jul 15 19:02:57.963:  Tnl 47100 L2TP: Congestion Control event received is retransmission
*Jul 15 19:02:57.963:  Tnl 47100 L2TP: Congestion Window size, Cwnd 1
*Jul 15 19:02:57.963:  Tnl 47100 L2TP: Slow Start threshold, Ssthresh 2
*Jul 15 19:02:57.963:  Tnl 47100 L2TP: Remote Window size, 500
*Jul 15 19:02:57.963:  Tnl 47100 L2TP: Control channel retransmit delay set to 4 seconds
*Jul 15 19:03:01.607:  Tnl 47100 L2TP: Update ns/nr, peer ns/nr 2/5, our ns/nr 5/2
!
```

The following partial output from the **debug vpdn l2x-events** command shows that traffic has been restarted with L2TP congestion avoidance operating in slow start mode.

```
Router# debug vpdn l2x-events
!
*Jul 15 14:45:16.123:  Tnl 30597 L2TP: Control channel retransmit delay set to 2 seconds
*Jul 15 14:45:16.123:  Tnl 30597 L2TP: Tunnel state change from idle to wait-ctl-reply
*Jul 15 14:45:16.131:  Tnl 30597 L2TP: Congestion Control event received is positive
acknowledgement
*Jul 15 14:45:16.131:  Tnl 30597 L2TP: Congestion Window size, Cwnd 2
*Jul 15 14:45:16.131:  Tnl 30597 L2TP: Slow Start threshold, Ssthresh 500
*Jul 15 14:45:16.131:  Tnl 30597 L2TP: Remote Window size, 500
*Jul 15 14:45:16.131:  Tnl 30597 L2TP: Congestion Ctrl Mode is Slow Start
!
```

Example Configuring VPDN Failure Event Logging

The following example first disables and then reenables VPDN failure event logging, and sets the maximum number of entries in the VPDN history failure table to 50. The contents of the history failure table are displayed and then cleared.

```
Router# configure terminal
Router(config)# no vpdn history failure
Router(config)# vpdn history failure
Router(config)# vpdn history failure table-size 50
Router(config)# end
Router# show vpdn history failure
!
Table size: 50
Number of entries in table: 1
User: user@cisco.com, MID = 1
NAS: isp, IP address = 172.21.9.25, CLID = 1
Gateway: hp-gw, IP address = 172.21.9.15, CLID = 1
Log time: 13:08:02, Error repeat count: 1
Failure type: The remote server closed this session
Failure reason: Administrative intervention
!
Router# clear vpdn history failure
```

Examples Configuring Generic VPDN Event Logging

The following example enables VPDN logging locally:

```
Router# configure terminal
Router(config)# vpdn logging local
```

The following example disables VPDN event logging locally, enables VPDN event logging at the remote tunnel endpoint, and enables the logging of both VPDN user and VPDN tunnel-drop events to the remote router:

```
Router# configure terminal
Router(config)# no vpdn logging local
Router(config)# vpdn logging remote
Router(config)# vpdn logging user
Router(config)# vpdn logging tunnel-drop
```

The following example disables the logging of VPDN events at the remote tunnel endpoint, and enables the logging of VPDN event log messages to the AAA server:

```
Router# configure terminal
Router(config)# no vpdn logging local
Router(config)# no vpdn logging remote
Router(config)# vpdn logging accounting
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
VPDN technology overview	VPDN Technology Overview module
VPDN commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*Cisco IOS VPDN Command Reference*
Technical support documentation for VPDNs	Virtual Private Dial-up Network (VPDN)
Dial Technologies commands: complete command syntax, command mode, defaults, usage guidelines, and examples	*Cisco IOS Dial Technologies Command Reference*
Concepts and tasks associated with configuring additional VPDN features	Configuring Additional VPDN Features module

Standards

Standard	Title
TCP/IP; slow start and congestion avoidance algorithms	*TCP/IP Illustrated, Volume 1*

MIBs

MIB	MIBs Link
• CISCO-VPDN-MGMT-MIB • CISCO-VPDN-MGMT-EXT-MIB	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 2661	*Layer Two Tunneling Protocol (L2TP)*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for VPDN Tunnel Management

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for VPDN Tunnel Management

Feature Name	Releases	Feature Information
L2TP Congestion Avoidance	Cisco IOS XE Release 2.3	This feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. It provides packet flow control and congestion avoidance by throttling Layer 2 Transport Protocol (L2TP) control messages as described in RFC 2661. The following commands were introduced or modified by this feature: **debug vpdn**, **l2tp congestion-control**.
Session Limit per VRF	Cisco IOS XE Release 2.1	This feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. It allows you to apply session limits on all VPDN groups associated with a common VPDN template. You can limit the number of VPDN sessions that terminate in a single VPN routing and forwarding (VRF) instance. The following commands were introduced or modified by this feature: **group session-limit**, **source vpdn-template,** and **vpdn-template**.
Timer and Retry Enhancements for L2TP	Cisco IOS XE Release 2.1	This feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. It allows the user to configure certain adjustable timers and counters for L2TP. The following commands were introduced by this feature: **l2tp tunnel busy timeout**, **l2tp tunnel retransmit initial retries**, and **l2tp tunnel retransmit initial timeout**.

Feature Name	Releases	Feature Information
VPDN Group Session Limiting	Cisco IOS XE Release 2.1	This feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. It allows the user to configure a limit on the number L2TP VPDN sessions allowed for each VPDN group. The following command was introduced by this feature: **session-limit** (VPDN).