

PPP over Ethernet Client

The PPP over Ethernet Client feature provides PPP over Ethernet (PPPoE) client support on routers.

PPPoE is a commonly used application in the deployment of digital subscriber lines (DSLs). The PPP over Ethernet Client feature expands PPPoE functionality by providing support for PPPoE on the client and the server.

- Prerequisites for PPP over Ethernet Client, on page 1
- Restrictions for PPP over Ethernet Client, on page 1
- Information About PPP over Ethernet Client, on page 2
- How to Configure PPP over Ethernet Client, on page 4
- Configuration Examples for the PPP over Ethernet Client, on page 14
- Additional References, on page 20
- Feature Information for PPP over Ethernet Client, on page 21

Prerequisites for PPP over Ethernet Client

PPP connections must be established between two endpoints over a serial link.

Restrictions for PPP over Ethernet Client

The PPPoE client does not support the following:

- More than ten clients per customer premises equipment (CPE)
- Quality of service (QoS) transmission with queueing on the dialer interface
- Dial-on-demand
- Easy VPN
- Native IPv6
- PPPoE client over ATM permanent virtual circuit (PVC)
- You can configure a dial-pool-number on a physical interface or sub-interface using the **pppoe-client dial-pool-number** *pool-number* command.

Note The pool number being unique cannot be used to configure with the same number on any other interfaces.

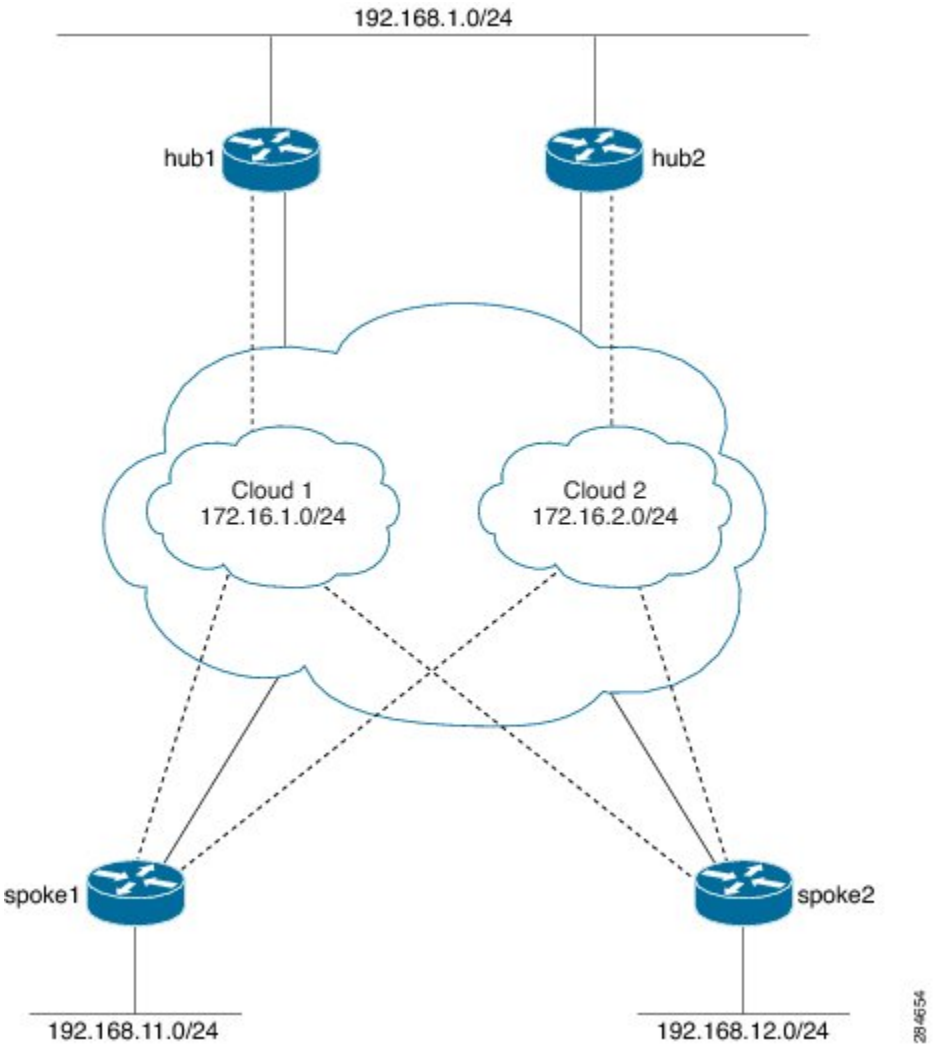
- Co-existence of the PPPoE client and server on the same device.
- Multilink PPP (MLP) on dialer interfaces
- Nonstop forwarding (NSF) with stateful switchover (SSO)
- When an IPv6 address is assigned to a subinterface from a server and if you remove the subinterface on client device, the IPv6 address might not be removed from the interface resulting in a ping failure after reconfiguring the subinterface. This is because you must shut the interface, first and then remove the subinterface.

Information About PPP over Ethernet Client

PPP over Ethernet Client Network Topology

The PPPoE Client feature provides PPPoE client support on routers at customer premises. Before the introduction of this feature, Cisco IOS XE software supported PPPoE only on the access server side. The figure below shows Dynamic Multipoint VPN (DMVPN) access to multiple hosts from the same PPPoE client using a common dialer interface and shared IPsec.

Figure 1: DMVPN Access to Multiple Hosts from the Same PPPoE Client



PPP over Ethernet Client Session Initiation

A PPPoE session is initiated by the PPPoE client. If the session has a timeout or is disconnected, the PPPoE client will immediately attempt to reestablish the session.

The following steps describe the exchange of packets that occurs when a PPPoE client initiates a PPPoE session:

1. The client broadcasts a PPPoE active discovery initiation (PADI) packet.
2. When the access concentrator receives a PADI packet that it can serve, it replies by sending a PPPoE active discovery offer (PADO) packet to the client.
3. Because the PADI packet was broadcast, the host may receive more than one PADO packet. The host looks through the PADO packets it receives and chooses one. The choice can be based on the access concentrator name or on the services offered. The host then sends a single PPPoE active discovery request (PADR) packet to the access concentrator that it has chosen.

4. The access concentrator responds to the PADR packet by sending a PPPoE active discovery session-confirmation (PADS) packet. At this point, a virtual access interface is created that will then negotiate PPP and the PPPoE session will run on this virtual access.

If a client does not receive a PADO packet for a PADI packet already received, the client sends out a PADI packet at predetermined intervals. That interval length is doubled for every successive PADI packet that does not evoke a response, until the interval reaches the configured maximum.

If PPP negotiation fails or the PPP line protocol is brought down for any reason, the PPPoE session and the virtual access will be brought down and the client will wait for a predetermined number of seconds before trying to establish another PPPoE session.

PPPoE Client over VLAN Interface

When configuring a PPPoE client on a VLAN interface, the device sends and receives PADO and PADI packets, but the PPPoE engine does not process PADO packets, and the PPPoE session does not come up. This enhancement provides the option to configure the PPPoE client over a VLAN, such as a Switched Virtual Interface (SVI), and establish the PPPoE session.

Figure 2: Sample Topology of a PPPoE Client and Server over a VLAN Interface



How to Configure PPP over Ethernet Client

Configuring a PPPoE Client

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no ip address**
5. **pppoe enable group global**
6. **pppoe-client dial-pool-number** *number*
7. **no shutdown**
8. **exit**
9. **interface dialer** *number*
10. **dialer** *pool number*
11. **encapsulation** *type*
12. **ipv6 enable**
13. Do one of the following:
    - **ip address negotiated**

- **ipv6 address autoconfig**
- **ipv6 dhcp client pd** *prefix-name*

14. **mtu** *size*
15. **ppp authentication pap callin**
16. **ppp pap sent-username** *username* **password** *password*
17. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Configures an interface and enters interface configuration mode.
Step 4	**no ip address** **Example:** `Device(config-if)# no ip address`	Removes the IP address.
Step 5	**pppoe enable group global** **Example:** `Device(config-if)# pppoe enable group global`	Enables a PPPoE session on the Gigabit Ethernet interface.
Step 6	**pppoe-client dial-pool-number** *number* **Example:** `Device(config-if)# pppoe-client dial-pool-number 1`	Configures a PPPoE client and specifies dial-on-demand routing (DDR) functionality.
Step 7	**no shutdown** **Example:** `Device(config-if)# no shutdown`	Removes the IP address.
Step 8	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 9	**interface dialer** *number* **Example:** `Device(config)# interface dialer 1`	Defines a dialer rotary group and enters interface configuration mode.
Step 10	**dialer** *pool number* **Example:** `Device(config-if)# dialer pool 1`	Specifies the dialing pool that the dialer interface uses to connect to a specific destination subnetwork.
Step 11	**encapsulation** *type* **Example:** `Device(config-if)# encapsulation ppp`	Specifies the encapsulation type. • Sets PPP as the encapsulation type.
Step 12	**ipv6 enable** **Example:** `Device(config-if)# ipv6 enable`	Enables IPv6 on the dialer interface.

Step 13	Do one of the following:
	• **ip address negotiated**
	• **ipv6 address autoconfig**
	• **ipv6 dhcp client pd** *prefix-name*
	Example:
	For IPv4
	`Device(config-if)# ip address negotiated`
	Example:
	For IPv6
	`Device(config-if)# ipv6 address autoconfig`
	Example:
	For DHCP IPv6
	`Device(config-if)# ipv6 dhcp client pd pd1`
		Specifies how the IP address is obtained for the dialer interface. This can be through one of the following as specified:
		• PPP/IP Control Protocol (IPCP) address negotiation
		• Dynamic Host Configuration Protocol (DHCP)

Step 14	**mtu** *size* **Example:** `Device(config-if)# mtu 1492`	Sets the maximum transmission unit (MTU) size.
Step 15	**ppp authentication pap callin** **Example:** `Device(config-if)# ppp authentication pap callin`	Enables at least one PPP authentication protocol and specifies the order in which protocols are selected on the interface.
Step 16	**ppp pap sent-username** *username* **password** *password* **Example:** `Device(config-if)# ppp pap sent-username username1 password password1`	Reenables remote Password Authentication Protocol (PAP) support for an interface and reuses the username and password parameters in the PAP authentication packet to the peer.

	Command or Action	Purpose
Step 17	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring PPPoE on the Server

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username** *username* **password** *password*
4. **bba-group pppoe** *bba-group-name*
5. **virtual-template** *template-number*
6. **exit**
7. **interface loopback** *interface-number*
8. Do one of the following:
   - **ip address** *ip-address mask*
   - **ipv6 address** *ipv6-address /prefix*
9. **exit**
10. **interface** *type number*
11. Do one of the following:
    - **no ip address**
    - **no ipv6 address**
12. **pppoe enable group** *bba-group-name*
13. **exit**
14. **interface virtual-template** *number*
15. Do one of the following:
    - **ip unnumbered loopback** *number*
    - **ipv6 unnumbered loopback** *number*
16. **description** *description*
17. **mtu** *size*
18. Do one of the following:
    - **peer default ip address pool** *local-pool-name*
    - **peer default ipv6 address pool** *local-pool-name*
    - **ipv6 dhcp server** *dhcp-pool-name*
19. **ppp authentication** *protocol*
20. **exit**
21. **ipv6 dhcp pool** *dhcp-pool-name*
22. **prefix-delegation pool** *local-pool-name*

23. Do one of the following:
    - **ip local pool** *pool-name* [*low-ip-address* [*high-ip-address*]]
    - **ipv6 local pool** *pool-name ipv6-subnet-id /prefix prefix-length*
24. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**username** *username* **password** *password* **Example:** `Device(config)# username username1 password password1`	Creates a PPPoE profile and enters BBA group configuration mode.
Step 4	**bba-group pppoe** *bba-group-name* **Example:** `Device(config)# bba-group pppoe bba1`	Creates a PPPoE profile and enters BBA group configuration mode.
Step 5	**virtual-template** *template-number* **Example:** `Device(config-bba-group)# virtual-template 1`	Creates a virtual template for a PPPoE profile with an identifying number to be used for cloning virtual access interfaces. • The range is 1 to 4095.
Step 6	**exit** **Example:** `Device(config-bba-group)# exit`	Exits BBA group configuration mode and returns to global configuration mode.
Step 7	**interface loopback** *interface-number* **Example:** `Device(config)# interface loopback 1`	Creates a loopback interface that emulates an interface that is always up and enters interface configuration mode. • The range is from 0 to 2147483647.
Step 8	Do one of the following: • **ip address** *ip-address mask* • **ipv6 address** *ipv6-address /prefix* **Example:**	Assigns an IP address to the loopback interface.

	Command or Action	Purpose

	Using an IPv4 address:
	`Device(config-if)# ip address 192.2.0.2 255.255.255.0`
	Example:
	Using an IPv6 address:
	`Device(config-if)# ipv6 address 2001:DB8:2::1/40`	

Step 9	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 10	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Configures an interface and enters interface configuration mode.

Step 11	Do one of the following:
	• **no ip address**
	• **no ipv6 address**
	Example:
	For an IPv4 address:
	`Device(config-if)# no ip address`
	Example:
	For an IPv6 address:
	`Device(config-if)# no ipv6 address`
		Removes the IP address.

Step 12	**pppoe enable group** *bba-group-name* **Example:** `Device(config-if)# pppoe enable group bba1`	Enables PPPoE sessions on the Gigabit Ethernet interface.
Step 13	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 14	**interface virtual-template** *number* **Example:** `Device(config)# interface virtual-template 1`	Creates a virtual template interface that can be configured and applied dynamically to create virtual access interfaces and enters interface configuration mode.
Step 15	Do one of the following: • **ip unnumbered loopback** *number* • **ipv6 unnumbered loopback** *number* **Example:** For IPv4:	Enables IP processing on an interface without explicitly assigning an IP address to the interface. • The *type* argument is the interface on which the router has assigned an IP address. • The *number* argument is the number of the interface on which you want to enable IP processing.

	Command or Action	Purpose

	`Device(config-if)# ip unnumbered loopback 1`
	Example:
	For IPv6:
	`Device(config-if)# ipv6 unnumbered loopback 1`	

Step 16	**description** *description* **Example:** `Device(config-if)# description pppoe bba1`	Adds a description to an interface configuration
Step 17	**mtu** *size* **Example:** `Device(config-if)# mtu 1492`	Sets the MTU size. • The range is from 64 to 9216.

Step 18	Do one of the following:
	• **peer default ip address pool** *local-pool-name*
	• **peer default ipv6 address pool** *local-pool-name*
	• **ipv6 dhcp server** *dhcp-pool-name*
	Example:
	For IPv4 addresses
	`Device(config-if)# peer default ip address pool pool1`
	Example:
	For IPv6 addresses
	`Device(config-if)# peer default ipv6 address pool pool1`
	Example:
	For DHCP assigned addresses:
	`Device(config-if)# ipv6 dhcp server dhcpv6pool`
		Specifies an address pool to provide IP addresses for remote peers connecting to this interface.

Step 19	**ppp authentication** *protocol* **Example:** `Device(config-if)# ppp authentication pap`	Enables at least one PPP authentication protocol and specifies the order in which the protocols are selected on the interface.
Step 20	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and returns to global configuration mode.
Step 21	**ipv6 dhcp pool** *dhcp-pool-name* **Example:** `Device(config)# ipv6 dhcp pool dhcpv6pool`	Creates a DHCP information pool and configures a local prefix pool from which prefixes can be delegated to clients.
Step 22	**prefix-delegation pool** *local-pool-name* **Example:**	Specifies a local prefix pool.

	Command or Action	Purpose
	`Device(config-dhcpv6)# prefix-delegation pool pool1`	

Step 23	Do one of the following:
	• **ip local pool** *pool-name* [*low-ip-address* [*high-ip-address*]]
	• **ipv6 local pool** *pool-name ipv6-subnet-id /prefix prefix-length*
	Example:
	For IPv4 addresses
	`Device(config)# ip local pool pool1 192.2.0.1 192.2.0.10`
	Example:
	For IPv6 addresses
	`Device(config)# ipv6 local pool pool1 2001:DB8:1200::/40 48`
		Configures a local pool of IP addresses to be used when a remote peer connects to a point-to-point interface.

Step 24	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Configuring a PPPoE Client over a VLAN Interface

Perform this task to configure a PPPoE client over a VLAN interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **switchport trunk allowed vlan** *vlan-list*
5. **switchport mode** {*access* | *trunk*}
6. **end**
7. **interface** *type number*
8. **no ip address**
9. **pppoe enable group global**
10. **pppoe-client dial-pool-number** *number*
11. **end**
12. **interface** *type number*
13. **ip address negotiated**
14. **encapsulation** *encapsulation-type*
15. **dialer pool** *number*
16. **pppoe pap sent-username** *username* **password** *password*
17. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface GigabitEthernet 0/0/0`	Configures an interface and enters interface configuration mode.
Step 4	**switchport trunk allowed vlan** *vlan-list* **Example:** `Device(config-if)# switchport trunk allowed vlan 10`	Sets the list of allowed VLANs that transmit traffic from this interface in tagged format when in trunking mode.
Step 5	**switchport mode** {*access* | *trunk*} **Example:** `Device(config-if)# switchport mode trunk`	Enables switchport mode. To specify a trunking VLAN interface, use the **switchport mode trunk** command.
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface vlan 10`	Configures an interface and enters interface configuration mode.
Step 8	**no ip address** **Example:** `Device(config-if)# no ip address`	Disables IP processing for an interface.
Step 9	**pppoe enable group global** **Example:** `Device(config-if)# pppoe enable group global`	Enables a PPPoE session on the Gigabit Ethernet interface.
Step 10	**pppoe-client dial-pool-number** *number* **Example:** `Device(config-if)# pppoe-client dial-pool-number 2`	Configures a PPPoE client and specifies dial-on-demand routing (DDR) functionality.

	Command or Action	Purpose
Step 11	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.
Step 12	**interface** *type number* **Example:** `Device(config)# interface Dialer2`	Configures an interface and enters interface configuration mode.
Step 13	**ip address negotiated** **Example:** `Device(config-if)# ip address negotiated`	Specifies that the IP address for the interface is obtained through PPP or IPCP (IP Control Protocol) address negotiation.
Step 14	**encapsulation** *encapsulation-type* **Example:** `Device(config-if)# encapsulation ppp`	Sets the encapsulation method used by the interface. To set PPP as the encapsulation type, use the **encapsulation ppp** command.
Step 15	**dialer pool** *number* **Example:** `Device(config-if)# dialer pool 2`	Specifies the dialing pool for a dialer interface to connect to a specific destination subnetwork.
Step 16	**pppoe pap sent-username** *username* **password** *password* **Example:** `Device(config-if)# ppp pap sent-username cisco@cisco.com password cisco`	Enables remote PAP support for an interface and authenticates the request packet to the peer.
Step 17	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and returns to privileged EXEC mode.

Verifying a PPPoE Client over a VLAN Interface

To verify that the PPPoE session is established, run the **show pppoe session** command. You should see the state type as UP.

```
ovld1# show pppoe session

     1 client session

Uniq ID  PPPoE  RemMAC          Port               VT   VA         State
           SID  LocMAC                                 VA-st      Type
    N/A    1    0000.aaaa.bbb0  V110               Di2  Vi2        UP
                0101.xxx1.dccc                          UP
```

Configuration Examples for the PPP over Ethernet Client

Example: Configuring a PPPoE Client

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# no ip address
Device(config-if)# pppoe enable
Device(config-if)# pppoe-client dial-pool-number 1
Device(config-if)# no shutdown
Device(config-if)# exit
Device(config)# interface dialer 1
Device(config-if)# mtu 1492
Device(config-if)# ip address negotiated
Device(config-if)# encapsulation ppp
Device(config-if)# dialer pool 1
Device(config-if)# ppp pap sent-username username1 password password1
Device(config-if)# end
```

Example: Configuring PPPoE on IPv4

Example: Server Configuration

```
Device> enable
Device# configure terminal
Device# username username1 password password1
Device(config)# bba-group pppoe bba1
Device(config-bba-group)# virtual-template 1
Device(config-bba-group)# exit
Device(config)# interface loopback 1
Device(config-if)# ip address 192.2.0.2 255.255.255.0
Device(config-if)# exit
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# no ip address
Device(config-if)# negotiation auto
Device(config-if)# pppoe enable group bba1
Device(config-if)# no shutdown
Device(config-if)# exit
Device(config)# interface virtual-template 1
Device(config-if)# description pppoe bba1
Device(config-if)# mtu 1492
Device(config-if)# ip unnumbered loopback 1
Device(config-if)# peer default ip address pool pool1
Device(config-if)# ppp authentication pap
Device(config-if)# exit
Device(config)# ip local pool pool1 192.2.0.1 192.2.0.10
Device(config)# end
```

Example: Client Configuration

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 0/0/0
Device(config-if)# no ip address
Device(config-if)# pppoe enable
Device(config-if)# pppoe-client dial-pool-number 1
Device(config-if)# no shutdown
Device(config-if)# exit
Device(config)# interface dialer 1
Device(config-if)# mtu 1492
Device(config-if)# ip address negotiated
Device(config-if)# encapsulation ppp
Device(config-if)# dialer pool 1
Device(config-if)# ppp pap sent-username username1 password password1
Device(config-if)# end
```

Example: Configuring PPPoE on IPv6 using DHCP

Example: Server Configuration using DHCP

Configure a username and a password for PPP client:

```
Host(config)# username username1 password password1
```

Create a PPP group GROUPA and associate it with a Virtual Template 1:

```
Host(config)# bba-group pppoe GROUPA
Host(config-bba-group)# virtual-template 1
Host(config-bba-group)# exit
```

Configure a loopback interface to be used on the Virtual Template 1:

```
Host(config)# interface loopback 1
Host(config-if)# ipv6 address 2001:DB8:2::1/40

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Host(config-if)# exit
```

Create a Virtual Template 1 and use the loopback interface as the IP address:

```
Host(config)# interface virtual-template 1
Host(config-if)# ipv6 unnumbered loopback 1
Host(config-if)# description pppoe GROUPA
Host(config-if)# mtu 1492

!Specify that PPP PAP authentication is used for authenticating connecting PPP
!clients
Host(config-if)# ppp authentication pap

!Enables DHCP for IPv6 service for the interface and specifies a pool for prefix
!delegation.
Host(config-if)# ipv6 dhcp server dhcpv6pool

Host(config-dhcp)# exit
```

Associate a physical interface with the PPP group GROUPA:

```
Host(config)# interface FastEthernet 0/0
Host(config-if)# no ip address
Host(config-if)# pppoe enable group GROUPA
Host(config-if)# no shutdown
Host(config-if)# exit
```

Create the local IPV6 address pool pool1 referred to in the Virtual Template 1

```
Host(config)# ipv6 local pool pool1 2001:DB8:1200::/40 48
```

Create a DHCP information pool and configure a local prefix pool from which prefixes can be delegated to clients.

```
Host(config)# ipv6 dhcp pool dhcpv6pool

!Specify local prefix pool
Host(config-dhcpv6)# prefix-delegation pool pool1
Host(config-dhcpv6)# end
```

Example: Client Configuration using DHCP

```
Device> enable
Device# configure terminal
Device(config)# hostname Client
```

Configure a physical interface and allocate it to a dialer pool. A logical dialer interface associated with the dialer pool can select a physical interface from this dialer pool when needed.

```
Client(config)# interface FastEthernet 0/0
Client(config-if)# no ip address
Client(config-if)# pppoe enable group global

!Allocate the physical interface to the dialer pool
Client(config-if)# pppoe-client dial-pool-number 1
Client(config-if)# no shutdown

%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
%LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
Client(config-if)# exit
```

Create the logical dialer interface and configure the pool used to pick physical interfaces

```
Client(config)# interface dialer 1

!Configure the pool used to pick physical interfaces.
Client(config-if)# dialer pool 1

!Sets the encapsulation method used by the interface to PPP.
Client(config-if)# encapsulation ppp
Client(config-if)# ipv6 enable

*Jun  2 23:51:36.455: %DIALER-6-BIND: Interface Vi2 bound to profile Di1
*Jun  2 23:51:36.459: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to up
*Jun  2 23:51:36.507: %DIALER-6-UNBIND: Interface Vi2 unbound from profile Di1
*Jun  2 23:51:36.519: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to down

!Enable Prefix delegation on the interface
Client(config-if)# ipv6 dhcp client pd dhcp_prefix_label

!Reduce MTU of the dialer interface to avoid unnecessary fragmentation caused by added
```

```
PPP headers.
Client(config-if)# mtu 1492
Client(config-if)# ppp authentication pap callin

!Configures the username and password that the client can use to authenticate with the
 server.
Client(config-if)# ppp pap sent-username username1 password password1

*Jun  2 23:52:20.999: %DIALER-6-BIND: Interface Vi2 bound to profile Di1
*Jun  2 23:52:21.003: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to up
*Jun  2 23:52:21.103: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2, changed state to up
Client(config-if)# end
```

Example: Verifying the PPPoE connection

Observe the interfaces of the client:

```
Client#show ipv6 interface brief

FastEthernet0/0            [up/up]
    unassigned
Dialer1                    [up/up]
    FE80::205:FF:FE50:6C08
Virtual-Access1            [up/up]
    unassigned
```

Observe the PPPoE session on the client:

```
Client# show pppoe session

     1 client session

Uniq ID  PPPoE  RemMAC          Port                   VT  VA         State
           SID  LocMAC                                     VA-st      Type
    N/A    324  0005.0050.9c08  Fa0/0                  Di1  Vi2        UP
                0005.0050.6c08                              UP
```

Observe the packets exchanged during the PPPoE session:

```
Client# show pppoe session packets

Total PPPoE sessions 1

SID     Pkts-In         Pkts-Out        Bytes-In        Bytes-Out
2846    0               6216            0               128136
```

Observe the DHCP session:

```
Server# show ipv6 dhcp binding

Client: FE80::205:FF:FE50:6C08
  DUID: 00030001000500506C08
  Username : unassigned
  Interface : Virtual-Access1.1
  IA PD: IA ID 0x000D0001, T1 302400, T2 483840
    Prefix: 2001:DB8::/48
            preferred lifetime 604800, valid lifetime 2592000
            expires at Jul 01 2013 09:17 PM (2591979 seconds)

Server# show ipv6 dhcp pool
```

```
DHCPv6 pool: dhcpv6pool
  Prefix pool: pool1
               preferred lifetime 604800, valid lifetime 2592000
  Active clients: 1
```

Example: Configuring PPPoE on IPv6

Configuring PPPoE on the Server

```
Device> enable
Device# configure terminal
Device(config)# hostname Host
```

Configure a username and a password for PPP client:

```
Host# username username1 password password1
```

Create a PPP group GROUPA and associate it with a Virtual Template 1:

```
Host(config)# bba-group pppoe GROUPA

*Jun  1 21:30:55.587: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
*Jun  1 21:30:55.591: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1,
changed state to up
Host(config-bba-group)# virtual-template 1
Host(config-bba-group)# exit
```

Configure a loopback interface to be used on the Virtual Template 1:

```
Host(config)# interface loopback 1
Host(config-if)# ipv6 address 2001:DB8:2::1/40

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up
Host(config-if)# exit
```

Create a Virtual Template 1 and use the loopback interface as the IP address:

```
Host(config)# interface virtual-template 1
Host(config-if)# ipv6 unnumbered loopback 1
Host(config-if)# description pppoe GROUPA
Host(config-if)# mtu 1492

!Configure the Virtual Template to hand out IP addresses from pool1
Host(config-if)# peer default ipv6 pool pool1

!Specify that PPP PAP authentication is used for authenticating connecting PPP clients
Host(config-if)# ppp authentication pap
Host(config-if)# exit
```

Associate a physical interface with the PPP group GROUPA:

```
Host(config)# interface FastEthernet 0/0
Host(config-if)# no ip address
Host(config-if)# pppoe enable group GROUPA
Host(config-if)# no shutdown
Host(config-if)# exit

*Jun  1 21:33:07.199: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
*Jun  1 21:33:08.199: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up
```

Create the IPV6 address pool pool1 referred to in the Virtual Template 1:

```
Host(config)# ipv6 local pool pool1 2001:DB8:1200::/40 48
Host(config)# end
```

Configuring PPPoE on the Client

```
Device> enable
Device# configure terminal
Device(config)# hostname Host
```

Configure a physical interface and allocate it to a dialer pool. A logical dialer interface associated with the dialer pool can select a physical interface from this dialer pool when needed.

```
Client(config)# interface FastEthernet 0/0
Client(config-if)# no ip address
Client(config-if)# pppoe enable group global

!Allocate the physical interface to the dialer pool
Client(config-if)# pppoe-client dial-pool-number 1
Client(config-if)# no shutdown

%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
%LINK-3-UPDOWN: Interface FastEthernet0/0, changed state to up
Client(config-if)# exit
```

Create the logical dialer interface and configure the pool used to pick physical interfaces

```
Client(config)# interface dialer 1

!Configure the pool used to pick physical interfaces.
Client(config-if)# dialer pool 1

!Sets the encapsulation method used by the interface to PPP.
Client(config-if)# encapsulation ppp
Client(config-if)# ipv6 enable

*Jun  3 00:10:48.031: %DIALER-6-BIND: Interface Vi2 bound to profile Di1
*Jun  3 00:10:48.035: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to up
*Jun  3 00:10:48.083: %DIALER-6-UNBIND: Interface Vi2 unbound from profile Di1
*Jun  3 00:10:48.091: %LINK-3-UPDOWN: Interface Virtual-Access2, changed state to down

!Configure the PPP clients to get IP addresses for dialer interfaces by using
!PPP negotiations with the server.
Client(config-if)# ipv6 address autoconfig

!Reduce MTU of the dialer interface to avoid unnecessary fragmentation caused by added PPP

!headers
Client(config-if)# mtu 1492
Client(config-if)# ppp authentication pap callin

!Configures the username and password that the client can use to authenticate with the
!server.
Client(config-if)# ppp pap sent-username username1 password password1

*Jun  3 00:11:54.843: %DIALER-6-BIND: Interface Vi2 bound to profile Di1
*Jun  3 00:11:54.847: %LINK-3-UPDOWN: Interface Virtual-Access2, changed
 state to up
*Jun  3 00:11:54.939: %LINEPROTO-5-UPDOWN: Line protocol on Interface
```

```
Virtual-Access2, changed state to up

Client(config-if)# end
```

Verifying the PPPoE connection

Observe the interfaces of the client:

```
Client# show ipv6 interface brief

FastEthernet0/0            [up/up]
    unassigned
Dialer1                    [up/up]
    FE80::205:FF:FE50:6C08
Virtual-Access1            [up/up]
    unassigned
```

Observe the PPPoE session on the client:

```
Client# show pppoe session

     1 client session

Uniq ID  PPPoE  RemMAC          Port                    VT  VA         State
           SID  LocMAC                                      VA-st      Type
    N/A    324  0005.0050.9c08  Fa0/0                   Di1  Vi2        UP
                0005.0050.6c08                               UP
```

Observe the packets exchanged during the PPPoE session:

```
Client# show pppoe session packets

Total PPPoE sessions 1

SID     Pkts-In         Pkts-Out        Bytes-In        Bytes-Out
2846    0               6216            0               128136
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	
Broadband Access Aggregation and DSL commands	Cisco IOS Broadband Access Aggregation and DSL Command Reference

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 PPP over Ethernet Client

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 PPP over Ethernet Client

Feature Name	Releases	Feature Information
PPP over Ethernet Client for IPv6	Cisco IOS XE Release 3.9S	The PPP over Ethernet Client feature provides IPv6 support.
PPP over Ethernet Client	Cisco IOS XE Release 3.5S	This feature was introduced. The PPP over Ethernet Client feature provides PPPoE client support on routers.
PPPoE Client over VLAN Interface	Cisco IOS XE Release Bengaluru 17.6	This feature was introduced. The PPPoE Client over VLAN Interface feature allows users to establish a PPPoE session over a VLAN interface.