PPPoE (PPP over Ethernet)—Encapsulates PPP frames within Ethernet frames, allowing PPP to be used with technologies like DSL and cable where ISP-supplied modems have an Ethernet jack facing the customer. ISPs like PPP for customer connections because it provides

- Authentication of the customer
- The ability to set the customer IP address—via IPCP, the PPP NCP (Network Control Protocol) for IP

Encapsulation and Tunneling

Since PPP is designed to work over serial lines, the L2 PPP frames must be encapsulated in L2 Ethernet frames for their trip to the ISP. Since both PPP and Ethernet frames have a "type" field, this works. That tunneling is what PPPoE does. Your router client and the ISP server are talking PPP to each other and PPPoE copes with delivery across the Ethernet in between.

Your router's Ethernet interface won't have an IP address because it won't directly handle IP packets.

CLIENT CONFIGURATION

For the CCNA, you only need to configure a client. So that you can practice that, Lab 15d shows a simplistic server configuration for the ISP side that you don't need to learn.

The Dialer Interface

CLIENT DIALER INTERFACE

```
1 interface Dialer 1
2  mtu 1492
3  ip address negotiated
4  encapsulation ppp
5  ppp chap hostname myISPaccount
6  ppp chap password myPassword
7  dialer pool 42
```

MTU (Maximum Transmission Unit) [Line 2]—Since PPPoE encapsulation adds 8 bytes of bloat to our frames (2 for PPP, 6 for PPPoE), we'll reduce how much data we put in each one from the Ethernet standard of 1500 bytes to 1492.

The IP address [Line 3] will be negotiated by PPP. Notice I didn't say that the address comes from DHCP, a common misconception that would use a different command (ip address dhcp).

L2 Encapsulation [Line 4]—We'll tell the dialer interface to use PPP encapsulation on our L3 IP packets. This is the same command we used for 802.1Q VLAN encapsulation, just using PPP instead of 802.1Q.

Authentication [Lines 5 and 6]—If your ISP requires authentication, you'll configure that on the dialer interface. You'll recall from the PPP chapter that if you don't configure a specific username for authentication [Line 6], IOS will default to the hostname of your router, explaining the peculiar keyword "hostname" in that command. Your client won't normally include the command "ppp authentication chap" unless you really do want the ISP to authenticate itself to you. Here, we're just doing one-way authentication, where the ISP challenges and identifies us—perhaps for billing purposes.

Dialer Pool [Line 7]—This will indirectly show which Ethernet interface we're using (more later). The pool number (42) has nothing to do with the dialer interface number (1).

The Physical Ethernet Interface

CLIENT ETHERNET INTERFACE

```
 8 interface GigabitEthernet0/0
 9  no ip address
10  pppoe-client dial-pool-number 42
11  pppoe enable
12  no shutdown
```

No IP Address [Line 9]—As mentioned, IP packets will be safely encapsulated inside PPP frames before they get to the Ethernet and this interface is being used for PPPoE only, so you won't configure an IP address on it. Since Ethernet interfaces lack an IP address by default, this won't show in a "show run" and you won't type it unless you're removing an earlier address.

Dialer Pool [Lines 10 & 7 (dialer config)]—Now, we can tie together our dialer interface and our Ethernet physical interface by giving them the same dial pool number. Notice that the dialer pool doesn't actually get configured, just mentioned. For our limited purposes, it's just a number (1-255). And again, there is no relationship between the dialer and dialer pool numbers.

PPPoE Encapsulation [Line 10]—A side effect of declaring the interface to be a PPPoE client is that IOS will automatically enable a PPPoE process (session) [Line 11]. Since it's automatic, you don't actually have to type line 11, but it's harmless if you do.

MAC Address—If your ISP requires a known MAC address, perhaps the one on the equipment they issued to you, you can spoof it in the physical interface. This can also make some "show" commands more readable.

```
CLIENT(config)# interface GigabitEthernet 0/0
CLIENT(config-if)# mac-address 0200.000.0001
```

Setting the 7th bit from the left to "1" (resulting in the 2 on the second hex digit) signifies a MAC address that wasn't officially issued by a central authority (we made it up).

THE DIALER INTERFACE

We'll use the dialer interface for all things layer 3, just like we've used Ethernet and serial interfaces in the past

- NAT/PAT would be configured on the dialer interface

  ```
  CLIENT(config)# interface Dialer 1
  CLIENT(config-if)# ip nat outside
  ```

- Routes would refer to the dialer as their exit interface and it would appear in the routing table.

  ```
  CLIENT(config)# ip route 0.0.0.0 0.0.0.0 Dialer 1
  ```

Show IP Interface Brief [Dialer 1]

```
1 CLIENT# show ip interface brief Dialer 1
2 Interface                  IP-Address      OK? Method Status                Protocol
3 Dialer1                    172.16.1.4      YES IPCP   up                    up
```

This shows us the IP address of the dialer interface. A "method" of "IPCP" [Line 3, column 4] refers to how the address got assigned (negotiated with the ISP via PPP's IP Control Protocol).

VIRTUAL-ACCESS INTERFACE

Once the PPPoE session is established with the ISP, IOS will automatically create a virtual access interface for it to run on. Ours is called "Virtual-Access1," or "Vi1" for short. "Di1" is short for "Dialer 1."

```
Jan  2 12:07:40.659: %DIALER-6-BIND: Interface Vi1 bound to profile Di1
Jan  2 12:07:40.663: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
Jan  2 12:07:40.739: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-
                Access1, changed state to up
```

The configuration for the virtual access interface will be derived, using the dialer interface you typed as a template. You can even see its configuration in the running config.

DIALER INTERFACE (WE TYPED)	VIRTUAL-ACCESS INTERFACE (IOS CREATED)
`CLIENT#` **`show run interface Dialer 1`**	`CLIENT#` **`show run interface Virtual-Access 1`**
`interface Dialer1`	`interface Virtual-Access1`
` mtu 1492`	` mtu 1492`
` ip address negotiated`	` no ip address`
` encapsulation ppp`	
` dialer pool 42`	
` ppp chap hostname myISPaccount`	
` ppp chap password 0 myPassword`	

Technically, the virtual access interface, running the PPPoE session, negotiates the actual PPP connection, but you don't need to know that.

Show Interfaces Virtual-Access 1 Configuration

This way of looking at the virtual-access interface's configuration adds the chap authentication lines to what is shown in the running config. Line 4 reminds us that the router used the dialer config as a template to derive the virtual-access interface configuration.

```
1  CLIENT# show interfaces Virtual-Access 1 configuration
2  Virtual-Access1 is a PPP over Ethernet link (sub)interface
3
4  Derived configuration : 107 bytes
5  !
6  interface Virtual-Access1
7   mtu 1492
8   ppp chap hostname myISPaccount
9   ppp chap password 0 myPassword
10 end
```

Binding Between Dialer and Virtual-Access Interfaces

The number "1," applied to Dialer1 and Virtual-Access1 does not imply any connection between the two; it's a coincidence. Virtual access interface numbers are applied sequentially, beginning with "1" and the virtual interfaces don't go away just because you eliminate the dialer interface that caused them. When experimenting in a lab, it's easy for the numbers not to match. Even a reboot can cause the router to create a new virtual access interface instead of reusing the old one. You can see the actual binding in the "show interfaces" and "show pppoe session" commands (details under "Verification").

```
CLIENT# show interfaces Dialer 1
CLIENT# show pppoe session
```

Recap of the Interfaces

In the table below, the "OSI Layer" column refers to what gets encapsulated/de-encapsulated at the interface (on the dialer, it's L3 IP packets), while the "Encapsulation" column refers to the encapsulation applied or removed, depending on the data's direction of travel.

INTERFACE	OSI LAYER	ENCAPSULATION	CONFIGURATION & PURPOSE
Dialer	3	PPP	MTU adjusted to accommodate PPP+PPPoE headers. Linked to a physical interface via a dial-pool. PPP Authentication (optional)
Virtual Access	2	PPPoE	Auto-derived from dialer. Created once the PPPoE session is established. The PPPoE session runs on this interface and negotiates to establish the PPP session
Ethernet	2	Ethernet	Declared to be a PPPoE client Linked to the associated dialer via a dial-pool. MAC Address (optional)

This hints at the source of PPPoE's opacity. Since virtual access interfaces are dynamically created only when they're needed, you can't type their configuration directly into them. It has to be crammed into a "virtual template" interface, the dialer, and copied out when the virtual access interface is created. In other words, there's a real disconnect between what you type and what you see running (or failing) on your router.

VERIFICATION

Show PPPoE Session [Interface Gi 0/0]

This command refers to the PPPoE session running on the virtual-access interface and quickly shows you which dialer, virtual-access, and Ethernet interfaces are involved. In this case, Di1 (Dialer 1), Vi1 (Virtual-Access 1), and GigabitEthernet 0/0 [Line 6].

```
1 CLIENT# show pppoe session interface GigabitEthernet 0/0
2      1 client session
3
4 Uniq ID  PPPoE  RemMAC          Port                    VT  VA         State
5            SID  LocMAC                                      VA-st      Type
6     N/A      3  001b.d476.6058  Gi0/0                  Di1  Vi1        UP
7                 0200.0000.0001                              UP
```

The formatting of this command's output is a bit of a mess. Realize that there are two lines of headings [Lines 4 and 5], followed by two lines of output per interface. For example the "LocMAC" heading (Local MAC) corresponds to 0200.0000.0001.

Show Interfaces Dialer 1

This command will show you information about both the dialer interface and its associated virtual access interface.

```
 1 CLIENT# show interfaces Dialer 1
 2 Dialer1 is up, line protocol is up (spoofing)
 3   Hardware is Unknown
 4   Internet address is 172.16.1.2/32
 5   MTU 1492 bytes, BW 56 Kbit/sec, DLY 20000 usec,
 6      reliability 255/255, txload 1/255, rxload 1/255
 7   Encapsulation PPP, LCP Closed, loopback not set
 8   Keepalive set (10 sec)
 9   DTR is pulsed for 1 seconds on reset
10   Interface is bound to Vi1
11   Last input never, output never, output hang never
12   Last clearing of "show interface" counters 00:07:51
13   Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
14   Queueing strategy: weighted fair
15   Output queue: 0/1000/64/0 (size/max total/threshold/drops)
16      Conversations  0/0/16 (active/max active/max total)
17      Reserved Conversations 0/0 (allocated/max allocated)
18      Available Bandwidth 42 kilobits/sec
19   5 minute input rate 0 bits/sec, 0 packets/sec
20   5 minute output rate 0 bits/sec, 0 packets/sec
21      4 packets input, 56 bytes
22      80 packets output, 1120 bytes
23 Bound to:
24 Virtual-Access1 is up, line protocol is up
25   Hardware is Virtual Access interface
26   MTU 1492 bytes, BW 56 Kbit/sec, DLY 20000 usec,
27      reliability 255/255, txload 1/255, rxload 1/255
28   Encapsulation PPP, LCP Open
29   Stopped: CDPCP
30   Open: IPCP
31   PPPoE vaccess, cloned from Dialer1
32   Vaccess status 0x44, loopback not set
33   Keepalive set (10 sec)
34   DTR is pulsed for 5 seconds on reset
35   Interface is bound to Di1 (Encapsulation PPP)
…
```

Spoofing [Line 2]—This just tells us that the L2 processing is actually on our related virtual access interface, not on the dialer interface we specified. Normally, the "show interfaces" command tells us L2 data about whatever interface we specify. Note: if the interface state *also* shows "up (spoofing)" instead of simply "up," you have a problem at L2 [see section 15d (troubleshooting)].

Learned IP Address [Line 4]—This is the address that was negotiated through PPP

Virtual Interface [Line 10]—This tells us which virtual access interface is bound to the dialer and detailed further down the screen / page [Lines 24-on].

PPP Encapsulation & Control Protocols [Lines 7 and 28]—The LCP is closed on the dialer interface but open, along with IPCP, on the virtual-access interface. This happens for the same reason that the L2 line protocol was listed as spoofing on the dialer [Line 2]; to wit, L3 IP happens on the dialer and L2 PPP happens on the derived virtual-access interface.

LCP (Link Control Protocol) [Line 28]—Handles core PPP functions across the link. "Open" is good—think "open for business".

Show Interfaces Virtual-Access 1

This command gives you exactly the same output as you saw in the second half of the "show interfaces dialer" command.

```
CLIENT# show interfaces Virtual-Access 1
Virtual-Access1 is up, line protocol is up
  Hardware is Virtual Access interface
  MTU 1492 bytes, BW 1000000 Kbit/sec, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, LCP Open
  Stopped: CDPCP
  Open: IPCP
  PPPoE vaccess, cloned from Dialer1
  Vaccess status 0x44, loopback not set
  Keepalive set (10 sec)
  Interface is bound to Di1 (Encapsulation PPP)
```

EXTRA RESOURCES

"Dialer Profiles." Cisco, n.d. Web. 19 Mar. 2017. <http://www.cisco.com/en/US/tech/tk801/tk133/tk159/tsd_technology_support_sub-protocol_home.html>. Concepts and Terminology.

"Configuring and Troubleshooting Dialer Profiles." *Document ID 10219*. Cisco, 15 Sept. 2005. Web. 19 Mar. 2017. <http://www.cisco.com/c/en/us/support/docs/dial-access/dial-on-demand-routing-ddr/10219-23.html>. Configuration.

"PPP over Ethernet Client." IOS 12.2t Feature Guide. Cisco, n.d. Web. 17 Apr. 2017. <http://www.cisco.com/en/US/docs/ios/12_2t/12_2t2/feature/guide/ftpppoec_support_TSD_Island_of_Content_Chapter.html>. Configuring, verifying, and troubleshooting PPPoE client session, including ATM (DSL WIC cards).

"PPP over Ethernet Client." (2012): n. pag. Cisco, 23 Feb. 2012. Web. 1 Apr. 2017. <http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/bbdsl/configuration/xe-3s/bba-pppoe-client.pdf>. PPPoE client configuration guide.

"Virtual Access PPP Features in Cisco IOS." Cisco. Cisco, 9 Sept. 2005. Web. 15 Apr. 2017. <http://www.cisco.com/c/en/us/support/docs/wan/point-to-point-protocol-ppp/14943-4.html>. Explains dialer (rotary group) and virtual-access interfaces.

REMAINING QUESTIONS

Mr. Odom ignores MSS Adjust—Does he trust the L2 MTU change to cascade upward to the TCP handshake? Is he counting on reassembly happening before TCP notices the fragmentation? Would fragmentation and reassembly simply be inefficient rather than problematic? Is he just illustrating the concept with the L2 MTU change and shortchanging reality to keep the books under 2000 pages, combined?

Bandwidth—both the dialer interface and the virtual-access interface default to dial-up 56K. Changing the bandwidth setting on the dialer after they're up does cascade, but it takes a while to show up—I noticed it the next day. How long does it really take and what caused the update?