

# Configuring a Cisco SOHO77 Router as a PPPoE Client with NAT

Document ID: 12925

---

## Introduction

### Prerequisites

- Requirements
- Components Used
- Conventions

### Configure

- Network Diagram
- Configurations

### Verify

### Troubleshoot

- DSL Physical Layer
- ATM Layer
- Ethernet Layer
- PPP Layer

### Debug

- DSL Physical Layer
- ATM Layer
- Ethernet Layer
- PPP Layer

### Related Information

---

## Introduction

In Cisco IOS® Software Release 12.1(3)XG, a Point-to-Point Protocol over Ethernet (PPPoE) client feature was introduced for the Cisco Small Office/Home Office (SOHO) 77. This feature allows the PPPoE functionality to be moved to the router. Multiple PCs can be installed behind the Cisco SOHO77, and before their traffic is sent to the PPPoE session, the data can be encrypted and filtered, and Network Address Translation (NAT) can run.

This document shows a PPPoE client configured on the Asynchronous Transfer Mode (ATM) interface (the DSL interface) of the Cisco SOHO77. This configuration can also be used on a Cisco 1700 with an Asymmetric Digital Subscriber Line (ADSL) WAN Interface Card (WIC).

The configuration on the Cisco 6400 Node Route Processor (NRP) can also be used on another router used as an aggregator and with an ATM interface.

## Prerequisites

### Requirements

There are no specific requirements for this document.

## Components Used

The information in this document is based on these software and hardware versions:

- Cisco SOHO77 Customer Premises Equipment (CPE) IOS Software Release 12.1(3)XP2
- Cisco 6400 UAC–NRP IOS Software Release 12.1(3)DC1
- Cisco 6400 UAC–NSP IOS Software Release 12.1(3)DB
- Cisco 6130 DSLAM–NI2 IOS Software Release 12.1(5)DA

The information presented in this document was created from devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If you are working in a live network, ensure that you understand the potential impact of any command before using it.

## Conventions

For more information on document conventions, see the Cisco Technical Tips Conventions.

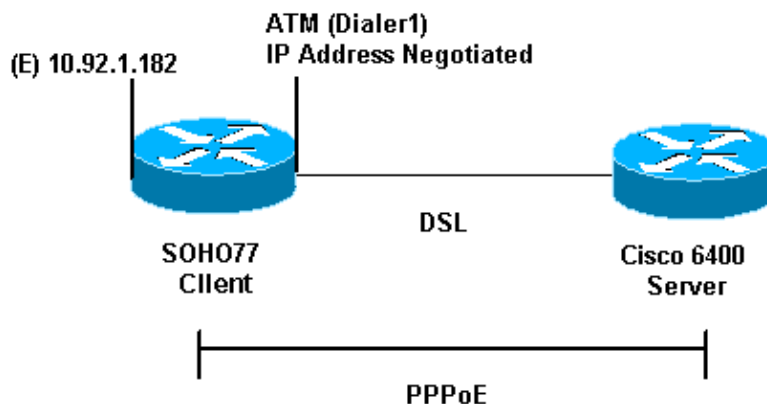
## Configure

In this section, you are presented with the information to configure the features described in this document.

**Note:** To find additional information on the commands used in this document, use the Command Lookup Tool (registered customers only).

## Network Diagram

This document uses this network setup:



## Configurations

This document uses these configurations:

- Cisco SOHO77
- Cisco 6400

PPPoE is configured on the Cisco SOHO77 with Virtual Private Dial-up Network (VPDN) commands. Make sure you configure these commands first.

```

!
vpdn enable
no vpdn logging

!--- Default.

!
vpdn-group pppoe
request-dialin

!--- The PPPoE client requests a session with the aggregation unit (6400 NRP).

    protocol pppoe
!

!--- Internal Ethernet network.

!
interface Ethernet0
ip address 10.92.1.182 255.255.255.0
ip nat inside

!--- DSL interface.

!
interface ATM0
no ip address
no atm ilmi-keepalive
bundle-enable
dsl operating-mode auto
hold-queue 224 in
!--- all defaults

!--- PPPoE runs on top of AAL5SNAP, but the encap aal5snap command is not used.

!
interface ATM0.1 point-to-point
pvc 1/1
    pppoe-client dial-pool-number 1

!--- pvc 1/1 is an example value that you must change to match the value
    !--- used by the Internet Service Provider (ISP).

!

!--- The PPPoE client code ties into a dialer interface, upon
!--- which a virtual-access interface is cloned.

!
interface Dialer1
ip address negotiated
ip mtu 1492

!--- Ethernet MTU is 1500 by default. In other words, 1492 + PPPoE headers = 1500.

ip nat outside
encapsulation ppp
dialer pool 1

!--- Ties to ATM interface.

```

```

ppp authentication chap callin
ppp chap hostname <hostname>
ppp chap password <password>
!

!--- The ISP instructs you about the type of authentication to use.
!--- To change from PPP CHAP to PPP PAP, replace the following three lines:
!--- ppp authentication chap callin
!--- ppp chap hostname <hostname>
!--- ppp chap password <password>
!--- with the following two lines:
!--- ppp authentication pap callin
!--- ppp pap sent-username <username> password <password>
!--- For NAT we are going to overload on the Dialer1 interface
!--- Then add a default route out since dialer IP address can change.

ip nat inside source list 1 interface Dialer1 overload
ip classless
ip route 0.0.0.0 0.0.0.0 dialer1
no ip http server
!
access-list 1 permit 10.92.1.0 0.0.0.255

!--- For NAT.

!

```

## Cisco 6400

```

*** local ppp user

!--- You can also use AAA.

username <username> password <password>

!--- Begin with the VPDN commands.
!--- Note the PPPoE binding to a virtual-template instead of on the ATM interface.
!--- You cannot (at this time) use more than one virtual-template (or VPDN group)
!--- for PPPoE beginning with the VPDN commands.

vpdn enable
no vpdn logging
!
vpdn-group pppoe
accept-dialin

!--- PPPoE server mode.

    protocol pppoe
    virtual-template 1
!
!
interface ATM0/0/0
no ip address
no atm ilmi-keepalive
hold-queue 500 in

!--- The binding to the virtual-template interface is configured in the VPDN group.

```

```

!
interface ATM0/0/0.182 point-to-point
 pvc 1/82
  encapsulation aal5snap

!--- The command is needed on the server side.

  protocol pppoe
  !
!

!--- Virtual-template is used instead of dialer interface.

!
interface Virtual-Template1
 ip unnumbered Loopback10
 ip mtu 1492
 peer default ip address pool ippool
 ppp authentication chap
!
!
interface Loopback10
 ip address 8.8.8.1 255.255.255.0
!
ip local pool ippool 9.9.9.1 9.9.9.5

```

## Verify

There is currently no verification procedure available for this configuration.

## Troubleshoot

This section provides information you can use to troubleshoot and debug your configuration.

To debug the PPPoE client on the Cisco SOHO77 or Cisco 6400, you must consider the protocol stack. Troubleshoot beginning at the bottom protocol layer.

1. DSL physical layer
2. ATM layer
3. Ethernet layer
4. PPP layer

## DSL Physical Layer

Ensure that the line is up and trained.

Enter the **show** commands as indicated in this example. The resultant output indicates the status of the line.

```

show int atm0

ATM0 is up, line protocol is up
  Hardware is PQUICC_SAR (with Alcatel ADSL Module)

show dsl interface atm0

!--- Look for "Showtime" in the first few lines.

```

Modem Status: ATU-R (DS) ATU-C (US)  
Showtime (DMTDSL\_SHOWTIME)

## ATM Layer

If the ATM interface is up, you can use the **debug atm packets** command to see if anything comes in from the ISP.

**Note:** You will not see outgoing packets with this command because of the way the packets are processed.

Enter the **debug atm packets** command as shown in this example.

```
debug atm packet
03:21:32: ATM0(I):
VCD:0x2 VPI:0x1 VCI:0x1 Type:0x0 SAP:AAAA CTL:03 OUI:0080C2 TYPE:0007
Length:0x30
03:21:32: 0000 0050 7359 35B7 0001 96A4 84AC 8864 1100 0001 000E C021 09AB
000C 0235
03:21:32: 279F 0000 0000
03:21:32:
```

The output as shown above, must contain the same Type, SAP, CTL, and OUI fields to indicate that the incoming ATM packet is AAL5SNAP.

## Ethernet Layer

Complete Ethernet frames are in the AAL5SNAP packets. There is no **debug ethernet packet** command, but you must perform some VPDN debug operations to view the PPPoE frames.

For reference, an Ethernet frame which is a PPPoE frame contains one of two ethertypes:

- 0x8863 Ethertype = PPPoE control packet (handles the PPPoE session).
- 0x8864 Ethertype = PPPoE data packet (contains PPP packets).

One important note is that there are two sessions in PPPoE. The PPPoE session which is a VPDN L2TP type session and the PPP session. So, to establish PPPoE we have a PPPoE session establishment phase and a PPP session establishment phase.

Termination usually involves a PPP termination phase and a PPPoE termination phase.

The PPPoE establishment phase consists of two steps:

- Step 1: Identify the PPPoE client and server (the MAC addresses).
- Step 2: Assign a session ID.

After this is complete, the normal PPP establishment occurs just like any other PPP connection.

To debug, use VPDN PPPoE debugs to determine if the PPPoE connect phase is successful.

1. Enter a debug command as shown in this example:

```
#debug vpdn pppoe-events
06:17:58: Sending PADI: vc=1/1
```

```

!--- A broadcast Ethernet frame (here, encapsulated in ATM) requests
!--- a PPPoE server with the message, "Is there a PPPoE server out there?"

06:18:00: PPPOE: we have got our pado, and the pado timer went off

!--- This is a unicast reply from a PPPoE server (similar to a DHCP offer).

06:18:00: OUT PADR from PPPoE tunnel

!--- This is a unicast reply to accept the offer.

06:18:00: IN PADS from PPPoE tunnel

!--- This is a confirmation that completes the establishment.

```

2. Initiate the PPP connection. The PPP establishment will now begin as in any other PPP initiation. After the PPPoE session is established, you can use **show vpdn** commands to get the status, as shown here:

```

#show vpdn
%No active L2TP tunnels
%No active L2F tunnels

PPPoE Tunnel and Session Information Total tunnels 1 sessions 1

PPPoE Tunnel Information

Session count: 1

PPPoE Session Information
SID      RemMAC      LocMAC      Intf      VASt      OIntf      VC
1        0050.7359.35b7  0001.96a4.84ac  Vi1      UP        AT0        1        1

```

3. Get packet count information with the **show vpdn session all** command, as shown here:

```

show vpdn session all
%No active L2TP tunnels
%No active L2F tunnels

PPPoE Session Information Total tunnels 1 sessions 1

session id: 1
local MAC address: 0001.96a4.84ac, remote MAC address: 0050.7359.35b7
virtual access interface: Vi1, outgoing interface: AT0, vc: 1/1
    1656 packets sent, 1655 received, 24516 bytes sent, 24486 received

```

Other potentially useful debug commands include **debug vpdn pppoe-data**, **debug pppoe-errors**, and **debug pppoe-packets**.

## PPP Layer

After the PPPoE session has been established, the PPP debugs are the same as for any other PPP establishment. The same **debug ppp negotiation** and **debug ppp authentication** commands are used.

**Note:** In the following sample, the hostname is "client1" and the name of the remote Cisco 6400 is "nrp-b".

Activate PPP negotiation or PPP authentication from the command line. The resultant output will resemble this:

```
06:36:03: Vi1 PPP: Treating connection as a callout
```

```

06:36:03: Vi1 PPP: Phase is ESTABLISHING, Active Open [0 sess, 1 load]
06:36:03: Vi1 PPP: No remote authentication for call-out
06:36:03: Vi1 LCP: O CONFREQ [Closed] id 1 len 10
06:36:03: Vi1 LCP:   MagicNumber 0x03013D43 (0x050603013D43)
06:36:03: Vi1 LCP: I CONFACK [REQsent] id 1 len 10
06:36:03: Vi1 LCP:   MagicNumber 0x03013D43 (0x050603013D43)
06:36:05: Vi1 LCP: I CONFREQ [ACKrcvd] id 2 len 15
06:36:05: Vi1 LCP:   AuthProto CHAP (0x0305C22305)
06:36:05: Vi1 LCP:   MagicNumber 0x65E315E5 (0x050665E315E5)
06:36:05: Vi1 LCP: O CONFACK [ACKrcvd] id 2 len 15
06:36:05: Vi1 LCP:   AuthProto CHAP (0x0305C22305)
06:36:05: Vi1 LCP:   MagicNumber 0x65E315E5 (0x050665E315E5)
06:36:05: Vi1 LCP: State is Open
06:36:05: Vi1 PPP: Phase is AUTHENTICATING, by the peer [0 sess, 1 load]
06:36:05: Vi1 CHAP: I CHALLENGE id 9 len 26 from "nrp-b"
06:36:05: Vi1 CHAP: Using alternate hostname client1
06:36:05: Vi1 CHAP: Username nrp-b not found
06:36:05: Vi1 CHAP: Using default password
06:36:05: Vi1 CHAP: O RESPONSE id 9 len 28 from "client1"
06:36:05: Vi1 CHAP: I SUCCESS id 9 len 4
06:36:05: Vi1 PPP: Phase is FORWARDING [0 sess, 1 load]
06:36:05: Vi1 PPP: Phase is AUTHENTICATING [0 sess, 1 load]
06:36:05: Vi1 PPP: Phase is UP [0 sess, 1 load]
06:36:05: Vi1 IPCP: O CONFREQ [Closed] id 1 len 10
06:36:05: Vi1 IPCP:   Address 0.0.0.0 (0x030600000000)
06:36:05: Vi1 CDPCP: O CONFREQ [Closed] id 1 len 4
06:36:05: Vi1 IPCP: I CONFREQ [REQsent] id 1 len 10
06:36:05: Vi1 IPCP:   Address 8.8.8.1 (0x030608080801)
06:36:05: Vi1 IPCP:   Address 8.8.8.1 (0x030608080801)
06:36:05: Vi1 IPCP:   Address 9.9.9.2 (0x030609090902)
06:36:05: Vi1 IPCP: O CONFREQ [ACKsent] id 2 len 10
06:36:05: Vi1 IPCP:   Address 9.9.9.2 (0x030609090902)
06:36:05: Vi1 LCP: I PROTREJ [Open] id 3 len 10 protocol
CDPCP (0x820701010004)
06:36:05: Vi1 CDPCP: State is Closed
06:36:05: Vi1 IPCP: I CONFACK [ACKsent] id 2 len 10
06:36:05: Vi1 IPCP:   Address 9.9.9.2 (0x030609090902)
06:36:05: Vi1 IPCP: State is Open
06:36:05: Di1 IPCP: Install negotiated IP interface address 9.9.9.2
06:36:05: Di1 IPCP: Install route to 8.8.8.1
06:36:06: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1,
changed state to up

```

## Debug

To debug the Cisco 6400 (the PPPoE server), you can use the same bottom-up procedure used for the Cisco SOHO77 (the client). The difference is in the DSL physical layer, where you need to check the DSLAM.

1. DSL physical layer
2. ATM layer
3. Ethernet layer
4. PPP layer

## DSL Physical Layer

To check the DSL physical layer, you need to see the DSL statistics on the DSLAM. For Cisco DSLAMs, use the **show dsl interface** command.

## ATM Layer

On the Cisco 6400 side, you can also use a **debug atm packet** command, and enable the Cisco 6400 for a specific PVC.

Enter **debug atm packet** with appropriate parameters from the command line, as follows:

```
debug atm packet interface atm 0/0/0.182 vc 1/82
```

The resultant output resembles this:

```
4d04h: ATM0/0/0.182(I):  
VCD:0x3 VPI:0x1 VCI:0x52 Type:0x900 SAP:AAAA CTL:03 OUI:0080C2 TYPE:0007 Length:0x30  
4d04h: 0000 0001 96A4 84AC 0050 7359 35B7 8864 1100 0001 000E C021 0A2E 000C 65E3  
4d04h: 15E5 0000 0000
```

**Note:** You will not see outgoing packets with this command because of the way the packets are processed.

## Ethernet Layer

The same VPDN **show** commands and debugs used on the Cisco SOHO77 can be used on the Cisco 6400 to look at the PPPoE establishment.

The following samples illustrate **show** and **debug** commands in context, along with their output. Use these commands as required.

```
#debug vpdn pppoe-events
```

```
4d04h: IN PADI from PPPoE tunnel  
  
4d04h: OUT PADO from PPPoE tunnel  
  
4d04h: IN PADR from PPPoE tunnel  
  
4d04h: PPPoE: Create session  
4d04h: PPPoE: VPN session created.  
  
4d04h: OUT PADS from PPPoE tunnel
```

```
#show vpdn
```

```
%No active L2TP tunnels  
%No active L2F tunnels
```

```
PPPoE Tunnel and Session Information Total tunnels 1 sessions 1
```

```
PPPoE Tunnel Information
```

```
Session count: 1
```

```
PPPoE Session Information
```

SID	RemMAC	LocMAC	Intf	VASt	OIntf	VC
1	0001.96a4.84ac	0050.7359.35b7	Vi4	UP	AT0/0/0	1 82

```
#show vpdn session all
```

```
nrp-b#show vpdn session all
```

```
%No active L2TP tunnels  
%No active L2F tunnels
```

PPPoE Session Information Total tunnels 1 sessions 1

session id: 1  
local MAC address: 0050.7359.35b7, remote MAC address: 0001.96a4.84ac  
virtual access interface: Vi4, outgoing interface: AT0/0/0, vc: 1/82  
30 packets sent, 28 received, 422 bytes sent, 395 received

Other debug commands include **debug vpdn pppoe-data**, **debug pppoe-errors**, and **debug pppoe-packets**.

## PPP Layer

Following is PPP debug output from the Cisco 6400 that corresponds to the earlier debug from the Cisco SOHO77.

Enter this command from the command line interface:

```
debug ppp negotiation and debug ppp authentication
```

The resultant output looks like this:

```
4d04h: Vi2 PPP: Treating connection as a dedicated line
4d04h: Vi2 PPP: Phase is ESTABLISHING, Active Open [0 sess, 1 load]
4d04h: Vi2 LCP: O CONFREQ [Closed] id 1 len 15
4d04h: Vi2 LCP:   AuthProto CHAP (0x0305C22305)
4d04h: Vi2 LCP:   MagicNumber 0x65F62814 (0x050665F62814)
4d04h: Vi2 LCP: I CONFREQ [REQsent] id 1 len 10
4d04h: Vi2 LCP:   MagicNumber 0x03144FF9 (0x050603144FF9)
4d04h: Vi2 LCP: O CONFACK [REQsent] id 1 len 10
4d04h: Vi2 LCP:   MagicNumber 0x03144FF9 (0x050603144FF9)
4d04h: Vi3 LCP: I ECHOREQ [Open] id 60 len 8 magic 0xA60C0000
4d04h: Vi3 LCP: O ECHOREP [Open] id 60 len 8 magic 0x51A0BEF6
4d04h: Vi2 LCP: TIMEOUT: State ACKsent
4d04h: Vi2 LCP: O CONFREQ [ACKsent] id 2 len 15
4d04h: Vi2 LCP:   AuthProto CHAP (0x0305C22305)
4d04h: Vi2 LCP:   MagicNumber 0x65F62814 (0x050665F62814)
4d04h: Vi2 LCP: I CONFACK [ACKsent] id 2 len 15
4d04h: Vi2 LCP:   AuthProto CHAP (0x0305C22305)
4d04h: Vi2 LCP:   MagicNumber 0x65F62814 (0x050665F62814)
4d04h: Vi2 LCP: State is Open
4d04h: Vi2 PPP: Phase is AUTHENTICATING, by this end [0 sess, 1 load]
4d04h: Vi2 CHAP: O CHALLENGE id 10 len 26 from "nrp-b"
4d04h: Vi2 CHAP: I RESPONSE id 10 len 28 from "client1"
4d04h: Vi2 PPP: Phase is FORWARDING [0 sess, 1 load]
4d04h: Vi2 PPP: Phase is AUTHENTICATING [0 sess, 1 load]
4d04h: Vi2 CHAP: O SUCCESS id 10 len 4
4d04h: Vi2 PPP: Phase is UP [0 sess, 1 load]
4d04h: Vi2 IPCP: O CONFREQ [Closed] id 1 len 10
4d04h: Vi2 IPCP:   Address 8.8.8.1 (0x030608080801)
4d04h: Vi2 IPCP: I CONFREQ [REQsent] id 1 len 10
4d04h: Vi2 IPCP:   Address 0.0.0.0 (0x030600000000)
4d04h: Vi2 IPCP: Pool returned 9.9.9.2
4d04h: Vi2 IPCP: O CONFNAK [REQsent] id 1 len 10
4d04h: Vi2 IPCP:   Address 9.9.9.2 (0x030609090902)
4d04h: Vi2 CDPCP: I CONFREQ [Not negotiated] id 1 len 4
4d04h: Vi2 LCP: O PROTREJ [Open] id 3 len 10 protocol CDPCP (0x820701010004)
4d04h: Vi2 IPCP: I CONFACK [REQsent] id 1 len 10
4d04h: Vi2 IPCP:   Address 8.8.8.1 (0x030608080801)
4d04h: Vi2 IPCP: I CONFREQ [ACKrcvd] id 2 len 10
4d04h: Vi2 IPCP:   Address 9.9.9.2 (0x030609090902)
4d04h: Vi2 IPCP: O CONFACK [ACKrcvd] id 2 len 10
4d04h: Vi2 IPCP:   Address 9.9.9.2 (0x030609090902)
4d04h: Vi2 IPCP: State is Open
4d04h: Vi2 IPCP: Install route to 9.9.9.2
```

4d04h: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access2,  
changed state to up

---

## Related Information

- [Cisco DSL Technical Support](#)
  - [Technical Support – Cisco Systems](#)
- 

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

---

Updated: Nov 23, 2007

Document ID: 12925

---