

# debug voip settlement error

To show all the settlement errors, enter the **debug voip settlement error** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement error**

**no debug voip settlement error**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(4)XH1	This command was introduced.

**Examples** The following is sample output from the **debug voip settlement error** command:

```
Router# debug voip settlement error

00:45:50:OSP:OSPPSockProcessRequest:http rcv init header failed
00:45:50:OSP:ospHttpSetupAndMonitor:attempt#0 on http=0x6141A514, limit=1 error=14310
```

**Usage Guidelines** **Error Code Definitions**

```
-1:OSP internal software error.
16:A bad service was chosen.
17:An invalid parameter was passed to OSP.
9010:Attempted to access an invalid pointer.
9020:A time related error occurred.

10010:OSP provider module failed initialization.
10020:OSP provider tried to access a NULL pointer.
10030:OSP provider could not fine transaction collection.
10040:OSP provider failed to obtain provider space.
10050:OSP provider tried to access an invalid handle.
10060:OSP provider has reached the maximum number of providers.

11010:OSP transaction tried to delete a transaction which was not allowed.
11020:OSP transaction tried a transaction which does not exist.
11030:OSP transaction tried to start a transaction, but data had already been delivered.
11040:OSP transaction could not identify the response given.
11050:OSP transaction failed to obtain transaction space.
11060:OSP transaction failed (possibly ran out) to allocate memory.
11070:OSP transaction tried to perform a transaction which is not allowed.
11080:OSP transaction found no more responses.
11090:OSP transaction could not find a specified value.
```

11100:OSP transaction did not have enough space to copy.  
11110:OSP transaction - call id did not match destination.  
11120:OSP transaction encountered an invalid entry.  
11130:OSP transaction tried to use a token too soon.  
11140:OSP transaction tried to use a token too late.  
11150:OSP transaction - source is invalid.  
11160:OSP transaction - destination is invalid.  
11170:OSP transaction - calling number is invalid.  
11180:OSP transaction - called number is invalid.  
11190:OSP transaction - call id is invalid.  
11200:OSP transaction - authentication id is invalid.  
11210:OSP transaction - call id was not found  
11220:OSP transaction - The IDS of the called number was invalid.  
11230:OSP transaction - function not implemented.  
11240:OSP transaction tried to access an invalid handle.  
11250:OSP transaction returned an invalid return code.  
11260:OSP transaction reported an invalid status code.  
11270:OSP transaction encountered an invalid token.  
11280:OSP transaction reported a status which could not be identified.  
11290:OSP transaction in now valid after it was not found.  
11300:OSP transaction could not find the specified destination.  
11310:OSP transaction is valid until not found.  
11320:OSP transaction - invalid signaling address.  
11330:OSP transaction could not find the ID of the transmitter.  
11340:OSP transaction could not find the source number.  
11350:OSP transaction could not find the destination number.  
11360:OSP transaction could not find the token.  
11370:OSP transaction could not find the list.  
11380:OSP transaction was not allowed to accumulate.  
11390:OSP transaction - transaction usage was already reported.  
11400:OSP transaction could not find statistics.  
11410:OSP transaction failed to create new statistics.  
11420:OSP transaction made an invalid calculation.  
11430:OSP transaction was not allowed to get the destination.  
11440:OSP transaction could not fine the authorization request.  
11450:OSP transaction - invalid transmitter ID.  
11460:OSP transaction could not find any data.  
11470:OSP transaction found no new authorization requests.

12010:OSP security did not have enough space to copy.  
12020:OSP security received and invalid argument.  
12030:OSP security could not find the private key.  
12040:OSP security encountered an un-implemented function.  
12050:OSP security ran out of memory.  
12060:OSP security received an invalid signal.  
12065:OSP security could not initialize the SSL database.  
12070:OSP security could not find space for the certificate.  
12080:OSP security has no local certificate info defined.  
12090:OSP security encountered a zero length certificate.

12100:OSP security encountered a certificate that is too big.  
12110:OSP security encountered an invalid certificate.  
12120:OSP security encountered a NULL certificate.  
12130:OSP security has too many certificates.  
12140:OSP security has no storage provided.  
12150:OSP security has no private key.  
12160:OSP security encountered an invalid context.  
12170:OSP security was unable to allocate space.  
12180:OSP security - CA certificates do not match.  
12190:OSP security found no authority certificates

12200:OSP security - CA certificate index overflow.

13010:OSP error message - failed to allocate memory.

13110:OSP MIME error - buffer is too small.  
13115:OSP MIME error - failed to allocate memory.  
13120:OSP MIME error - could not find variable.  
13125:OSP MIME error - no input was found.  
13130:OSP MIME error - invalid argument.  
13135:OSP MIME error - no more space.  
13140:OSP MIME error - received an invalid type.  
13145:OSP MIME error - received an invalid subtype.  
13150:OSP MIME error - could not find the specified protocol.  
13155:OSP MIME error - could not find MICALG.  
13160:OSP MIME error - boundary was not found.  
13165:OSP MIME error - content type was not found.  
13170:OSP MIME error - message parts were not found.

13301:OSP XML error - received incomplete XML data.  
13302:OSP XML error - bad encoding of XML data.  
13303:OSP XML error - bad entity in XML data.  
13304:OSP XML error - bad name in XML data.  
13305:OSP XML error - bad tag in XML data.  
13306:OSP XML error - bad attribute in XML data.  
13307:OSP XML error - bad CID encoding in XML data.  
13308:OSP XML error - bad element found in XML data.  
13309:OSP XML error - no element found in XML data.  
13310:OSP XML error - no attribute found in XML data.  
13311:OSP XML error - OSP received invalid arguments.  
13312:OSP XML error - failed to create a new buffer.  
13313:OSP XML error - failed to get the size of a buffer.  
13314:OSP XML error - failed to send the buffer.  
13315:OSP XML error - failed to read a block from the buffer.  
13316:OSP XML error - failed to allocate memory.  
13317:OSP XML error - could not find the parent.  
13318:OSP XML error - could not find the child.  
13319:OSP XML error - data type not found in XML data.  
13320:OSP XML error - failed to write a clock to the buffer.

13410:OSP data error - no call id preset.  
13415:OSP data error - no token present.  
13420:OSP data error - bad number presented.  
13425:OSP data error - no destination found.  
13430:OSP data error - no usage indicator present.  
13435:OSP data error - no status present.  
13440:OSP data error - no usage configured.  
13445:OSP data error - no authentication indicator.  
13450:OSP data error - no authentication request.  
13455:OSP data error - no authentication response.  
13460:OSP data error - no authentication configuration.  
13465:OSP data error - no re-authentication request.  
13470:OSP data error - no re-authentication response.  
13475:OSP data error - invalid data type present.  
13480:OSP data error - no usage information available.  
13485:OSP data error - no token info present.  
13490:OSP data error - invalid data present.

13500:OSP data error - no alternative info present.  
13510:OSP data error - no statistics available.  
13520:OSP data error - no delay present.  
13610:OSP certificate error - memory allocation failed.

14010:OSP communications error - invalid communication size.  
14020:OSP communications error - bad communication value.  
14030:OSP communications error - parser error.  
14040:OSP communications error - no more memory available.  
14050:OSP communications error - communication channel currently in use.

14060:OSP communications error - invalid argument passed.  
14070:OSP communications error - no service points present.  
14080:OSP communications error - no service points available.  
14085:OSP communications error - thread initialization failed.  
14086:OSP communications error - communications is shutdown.

14110:OSP message queue error - no more memory available.  
14120:OSP message queue error - failed to add a request.  
14130:OSP message queue error - no event queue present.  
14140:OSP message queue error - invalid arguments passed.

14210:OSP HTTP error - 100 - bad header.  
14220:OSP HTTP error - 200 - bad header.  
14221:OSP HTTP error - 400 - bad request.  
14222:OSP HTTP error - bad service port present.  
14223:OSP HTTP error - failed to add a request.  
14230:OSP HTTP error - invalid queue present.  
14240:OSP HTTP error - bad message received.  
14250:OSP HTTP error - invalid argument passed.  
14260:OSP HTTP error - memory allocation failed.  
14270:OSP HTTP error - failed to create a new connection.  
14280:OSP HTTP error - server error.  
14290:OSP HTTP error - HTTP server is shutdown.  
14292:OSP HTTP error - failed to create a new SSL connection.  
14295:OSP HTTP error - failed to create a new SSL context.  
14297:OSP HTTP error - service unavailable.

14300:OSP socket error - socket select failed.  
14310:OSP socket error - socket receive failed.  
14315:OSP socket error - socket send failed.  
14320:OSP socket error - failed to allocate memory for the receive buffer.  
14320:OSP socket error - socket reset.  
14330:OSP socket error - failed to create the socket.  
14340:OSP socket error - failed to close the socket.  
14350:OSP socket error - failed to connect the socket.  
14360:OSP socket error - failed to block I/O on the socket.  
14370:OSP socket error - failed to disable nagle on the socket.

14400:OSP SSL error - failed to allocate memory.  
14410:OSP SSL error - failed to initialize the context.  
14420:OSP SSL error - failed to retrieve the version.  
14430:OSP SSL error - failed to initialize the session.  
14440:OSP SSL error - failed to attach the socket.  
14450:OSP SSL error - handshake failed.  
14460:OSP SSL error - failed to close SSL.  
14470:OSP SSL error - failed to read from SSL.  
14480:OSP SSL error - failed to write to SSL.  
14490:OSP SSL error - could not get certificate.  
14495:OSP SSL error - no root certificate found.  
14496:OSP SSL error - failed to set the private key.  
14497:OSP SSL error - failed to parse the private key.  
14498:OSP SSL error - failed to add certificates.  
14499:OSP SSL error - failed to add DN.

15410:OSP utility error - not enough space for copy.  
15420:OSP utility error - no time stamp has been created.  
15430:OSP utility error - value not found.  
15440:OSP utility error - failed to allocate memory.  
15450:OSP utility error - invalid argument passed.

15500:OSP buffer error - buffer is empty.  
15510:OSP buffer error - buffer is incomplete.

15980:OSP POW error.  
15990:OSP Operating system conditional variable timeout.

16010:OSP X509 error - serial number undefined.  
16020:OSP X509 error - certificate undefined.  
16030:OSP X509 error - invalid context.  
16040:OSP X509 error - decoding error.  
16050:OSP X509 error - unable to allocate space.  
16060:OSP X509 error - invalid data present.  
16070:OSP X509 error - certificate has expired.  
16080:OSP X509 error - certificate not found.

17010:OSP PKCS1 error - tried to access invalid private key pointer  
17020:OSP PKCS1 error - unable to allocate space.  
17030:OSP PKCS1 error - invalid context found.  
17040:OSP PKCS1 error - tried to access NULL pointer.  
17050:OSP PKCS1 error - private key overflow.

18010:OSP PKCS7 error - signer missing.  
18020:OSP PKCS7 error - invalid signature found.  
18020:OSP PKCS7 error - unable to allocate space.  
18030:OSP PKCS7 error - encoding error.  
18040:OSP PKCS7 error - tried to access invalid pointer.  
18050:OSP PKCS7 error - buffer overflow.

19010:OSP ASN1 error - tried to access NULL pointer.  
19020:OSP ASN1 error - invalid element tag found.  
19030:OSP ASN1 error - unexpected high tag found.  
19040:OSP ASN1 error - invalid primitive tag found.  
19050:OSP ASN1 error - unable to allocate space.  
19060:OSP ASN1 error - invalid context found.  
19070:OSP ASN1 error - invalid time found.  
19080:OSP ASN1 error - parser error occurred.  
19090:OSP ASN1 error - parsing complete.  
19100:OSP ASN1 error - parsing defaulted.  
19110:OSP ASN1 error - length overflow.  
19120:OSP ASN1 error - unsupported tag found.  
19130:OSP ASN1 error - object ID not found.  
19140:OSP ASN1 error - object ID mismatch.  
19150:OSP ASN1 error - unexpected int base.  
19160:OSP ASN1 error - buffer overflow.  
19170:OSP ASN1 error - invalid data reference ID found.  
19180:OSP ASN1 error - no content value for element found.  
19190:OSP ASN1 error - integer overflow.

20010:OSP Crypto error - invalid parameters found.  
20020:OSP Crypto error - unable to allocate space.  
20030:OSP Crypto error - could not verify signature.  
20040:OSP Crypto error - implementation specific error.  
20050:OSP Crypto error - tried to access invalid pointer.  
20060:OSP Crypto error - not enough space to perform operation.

21010:OSP PKCS8 error - invalid private key pointer found.  
21020:OSP PKCS8 error - unable to allocate space for operation.  
21030:OSP PKCS8 error - invalid context found.  
21040:OSP PKCS8 error - tried to access NULL pointer.  
21050:OSP PKCS8 error - private key overflow.

22010:OSP Base 64 error - encode failed.  
22020:OSP Base 64 error - decode failed.

22510:OSP audit error - failed to allocate memory.

156010:OSP RSN failure error - no data present.

156020:OSP RSN failure error - data is invalid.

# debug voip settlement exit

To show all the settlement function exits, enter the **debug voip settlement exit** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement exit**

**no debug voip settlement exit**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(4)XH1	This command was introduced.

**Examples** The following is sample output from the **debug voip settlement exit** command:

```
Router# debug voip settlement exit

01:21:10:OSP:EXIT :OSPPMimeMessageInit()
01:21:10:OSP:EXIT :OSPPMimeMessageSetContentAndLength()
01:21:10:OSP:EXIT :OSPPMimeMessageBuild()
01:21:10:OSP:EXIT :OSPPMimePartFree()
01:21:10:OSP:EXIT :OSPPMimePartFree()
01:21:10:OSP:EXIT :OSPPMimeDataFree()
01:21:10:OSP:EXIT :OSPPMimeMessageCreate()
01:21:10:OSP:EXIT :OSPPMsgInfoAssignRequestMsg()
01:21:10:OSP:EXIT :osppHttpSelectConnection
01:21:10:OSP:EXIT :OSPPSockCheckServicePoint() isconnected(1)
01:21:10:OSP:EXIT :osppHttpBuildMsg()
01:21:10:OSP:EXIT :OSPPSockWrite() (0)
01:21:10:OSP:EXIT :OSPPSSLSessionWrite() (0)
01:21:10:OSP:EXIT :OSPPSSLSessionRead() (0)
01:21:10:OSP:EXIT :OSPPSSLSessionRead() (0)
01:21:10:OSP:EXIT :OSPPHttpParseHeader
01:21:10:OSP:EXIT :OSPPHttpParseHeader
01:21:10:OSP:EXIT :OSPPSSLSessionRead() (0)
01:21:10:OSP:EXIT :OSPPUtilMemCaseCmp()
```

# debug voip settlement misc

To show the details on the code flow of each settlement transaction, enter the **debug voip settlement misc** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement misc**

**no debug voip settlement misc**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(4)XH1	This command was introduced.

**Examples** The following is sample output from the **debug voip settlement misc** command:

```
Router# debug voip settlement misc

00:52:03:OSP:osp_authorize:callp=0x6142770C
00:52:03:OSP:OSPPTTransactionRequestNew:ospvTrans=0x614278A8
00:52:03:OSP:osppCommMonitor:major:minor=(0x2:0x1)
00:52:03:OSP:HTTP connection:reused
00:52:03:OSP:osppHttpSetupAndMonitor:HTTP=0x6141A514, QUEUE_EVENT from eventQ=0x6141A87C,
comm=0x613F16C4, msginfo=0x6142792C
00:52:03:OSP:osppHttpSetupAndMonitor:connected = <TRUE>
00:52:03:OSP:osppHttpSetupAndMonitor:HTTP=0x6141A514, build msginfo=0x6142792C, trans=0x2
00:52:04:OSP:osppHttpSetupAndMonitor:HTTP=0x6141A514, msg built and sent:error=0,
msginfo=0x6142792C
00:52:04:OSP:osppHttpSetupAndMonitor:monitor exit. errorcode=0
00:52:04:OSP:osppHttpSetupAndMonitor:msginfo=0x6142792C, error=0, shutdown=0
00:52:04:OSP:OSPMsgInfoProcessResponse:msginfo=0x6142792C, err=0, trans=0x614278A8,
handle=2
00:52:04:OSP:OSPMsgInfoChangeState:transp=0x614278A8, msgtype=12 current state=2
00:52:04:OSP:OSPMsgInfoChangeState:transp=0x614278A8, new state=4
00:52:04:OSP:OSPMsgInfoProcessResponse:msginfo=0x6142792C, context=0x6142770C, error=0
00:52:04:OSP:osp_get_destination:trans_handle=2, get_first=1, callinfop=0x614275E0
00:52:04:OSP:osp_get_destination:callinfop=0x614275E0 get dest=1.14.115.51,
validafter=1999-01-20T02:04:32Z, validuntil=1999-01-20T02:14:32Z
00:52:04:OSP:osp_parse_destination:dest=1.14.115.51
00:52:04:OSP:osp_get_destination:callinfop=0x614275E0, error=0, ip_addr=1.14.115.51,
credit=60
00:52:06:OSP:stop_settlement_ccapi_accounting:send report for callid=0x11, transhandle=2
00:52:06:OSP:osp_report_usage:transaction=2, duration=0, lostpkts=0, lostfrs=0,
lostpktr=0, lostfrr=0
```

# debug voip settlement network

To show all the messages exchanged between a router and a settlement provider, enter the **debug voip settlement network** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement network**

**no debug voip settlement network**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(4)XH1	This command was introduced.

**Usage Guidelines** Using the **debug voip settlement network** command shows messages, in detail, in HTTP and XML formats.

**Examples** The following is sample output from the **debug voip settlement network** command:

```
Router# debug voip settlement network

00:47:25:OSP:HTTP connection:reused
00:47:25:OSP:OSPPSockWaitTillReady:HTTPCONN=0x6141A514, fd=0
00:47:25:OSP:OSPPSockWaitTillReady:read=0, timeout=0, select=1
00:47:25:OSP:osppHttpBuildAndSend():http=0x6141A514 sending:
POST /scripts/simulator.dll?handler HTTP/1.1
Host:1.14.115.12
content-type:text/plain
Content-Length:439
Connection:Keep-Alive

Content-Type:text/plain
Content-Length:370

<?xml version="1.0"?><Message messageId="1" random="8896">
<AuthorisationRequest componentId="1">
<Timestamp>
1993-03-01T00:47:25Z</Timestamp>
<CallId>
<![CDATA[12]]></CallId>
<SourceInfo type="e164">
5551111</SourceInfo>
<DestinationInfo type="e164">
```

```

5552222</DestinationInfo>
<Service/>
<MaximumDestinations>
3</MaximumDestinations>
</AuthorisationRequest>
</Message>

00:47:25:OSP:OSPPSockWaitTillReady:HTTPCONN=0x6141A514, fd=0
00:47:25:OSP:OSPPSockWaitTillReady:read=0, timeout=1, select=1
00:47:25:OSP:OSPM_SEND:bytes_sent = 577
00:47:25:OSP:OSPPSockProcessRequest:SOCKFD=0, Expecting 100, got
00:47:25:OSP:OSPPSockWaitTillReady:HTTPCONN=0x6141A514, fd=0
00:47:25:OSP:OSPPSockWaitTillReady:read=1, timeout=1, select=1
00:47:25:OSP:OSPPSSLSessionRead() recving 1 bytes:
HTTP/1.1 100 Continue
Server:Microsoft-IIS/4.0
Date:Wed, 20 Jan 1999 02:01:54 GMT
00:47:25:OSP:OSPPSockProcessRequest:SOCKFD=0, Expecting 200, got
00:47:25:OSP:OSPPSockWaitTillReady:HTTPCONN=0x6141A514, fd=0
00:47:25:OSP:OSPPSockWaitTillReady:read=1, timeout=1, select=1
00:47:25:OSP:OSPPSSLSessionRead() recving 1 bytes:
HTTP/1.1 200 OK
Server:Microsoft-IIS/4.0
Date:Wed, 20 Jan 1999 02:01:54 GMT
Connection:Keep-Alive
Content-Type:multipart/signed; protocol="application/pkcs7-signature"; micalg=sha1;
boundary=bar
Content-Length:1689

00:47:25:OSP:OSPPSockProcessRequest:SOCKFD=0, error=0, HTTP response

00:47:25:OSP:OSPPSockWaitTillReady:HTTPCONN=0x6141A514, fd=0
00:47:25:OSP:OSPPSockWaitTillReady:read=1, timeout=1, select=1
00:47:25:OSP:OSPPSSLSessionRead() recving 1689 bytes:

--bar
Content-Type:text/plain
Content-Length:1510

<?xml version="1.0"?><Message messageId="1" random="27285">
<AuthorisationResponse componentId="1">
<Timestamp>
1999-01-20T02:01:54Z</Timestamp>
<Status>
<Description>
success</Description>
<Code>
200</Code>
</Status>
<TransactionId>
101</TransactionId>
<Destination>
<AuthorityURL>
http://www.myauthority.com</AuthorityURL>
<CallId>
<![CDATA[12]]></CallId>
<DestinationInfo type="e164">
5552222</DestinationInfo>
<DestinationSignalAddress>
1.14.115.51</DestinationSignalAddress>
<Token encoding="base64">
PD94bWwgdmVyc2lvcj0xLjA/PjxNZXNzYWdlIG1lc3NhZ2VJZD0iMSIgcGFuZG9tPSIxODM0OSI+PFRva2VuSW5mbz
48U291cmNlSW5mbyB0eXB1PSJlMTY0Ij41NTUxMTEwPC90b3R5Y2VJbmZvPjxZXXN0aW5hdGlvbkluZm8gdHlwZT0i
ZTE2NCI+NTU1MjIyMjwvRGVzdGluYXRpb25JbmZvPjxYXWwSWQ+PCFbQ0RBVEFbMV1dPjwvQ2FsbElkPjxWYXpZE

```

```

FmdGVyPjE5OTgtMTItMDhUMjA6MDQ6MFo8L1ZhbG1kQWZ0ZXI+PFZhbG1kVW50aWw+MTk5OS0xMi0zMVQyMzo1OT01
OV08L1ZhbG1kVW50aWw+PFYyYjY5ZjYwN0aW9uSWQ+MTAxPC9UcmFuc2FjdGlvbk1kPjxVc2FnZURldGFpbD48QW1vdW
50PjE0NDAwPC9BbW91bnQ+PEluY3JlbWVudD4xPC9JbmNyZW11bnQ+PFN1cnZpY2UvPjxVbml0PnM8L1VuaXQ+PC9V
c2FnZURldGFpbD48L1Rva2VuSW5mbz48L011c3NhZ2U+</Token>
<UsageDetail>
<Amount>
60</Amount>
<Increment>
1</Increment>
<Service/>
<Unit>
s</Unit>
</UsageDetail>
<ValidAfter>
1999-01-20T01:59:54Z</ValidAfter>
<ValidUntil>
1999-01-20T02:09:54Z</ValidUntil>
</Destination>
<transnexus.com:DelayLimit critical="False">
1000</transnexus.com:DelayLimit>
<transnexus.com:DelayPreference critical="False">
1</transnexus.com:DelayPreference>
</AuthorisationResponse>
</Message>

--bar
Content-Type:application/pkcs7-signature
Content-Length:31

This is your response signature

--bar--

```

# debug voip settlement security

To show all the tracing related to security, such as Secure Socket Layer (SSL) or Secure Multipurpose Internet Mail Extensions (S/MIME), enter the **debug voip settlement security** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement security**

**no debug voip settlement security**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.0(4)XH1	This command was introduced.

**Examples** Not available because of security issues.

# debug voip settlement ssl

To display information about the Secure Socket Layer (SSL) connection, use the **debug voip settlement ssl** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement ssl**

**no debug voip settlement ssl**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.2(11)T	This command was introduced.

**Usage Guidelines** For complete information about the SSL connection, use the **debug voip settlement ssl** command if you see one of the following errors generated from the **debug voip settlement error** command.

```
14400:OSP SSL error - failed to allocate memory.
14410:OSP SSL error - failed to initialize the context.
14420:OSP SSL error - failed to retrieve the version.
14430:OSP SSL error - failed to initialize the session.
14440:OSP SSL error - failed to attach the socket.
14450:OSP SSL error - handshake failed.
14460:OSP SSL error - failed to close SSL.
14470:OSP SSL error - failed to read from SSL.
14480:OSP SSL error - failed to write to SSL.
14490:OSP SSL error - could not get certificate.
14495:OSP SSL error - no root certificate found.
14496:OSP SSL error - failed to set the private key.
14497:OSP SSL error - failed to parse the private key.
14498:OSP SSL error - failed to add certificates.
14499:OSP SSL error - failed to add DN.
```

**Examples** The following example shows the debug output when the SSL is making a good connection to the Open Settlement Protocol server:

```
*May 15 11:53:42.871:OSP:
*May 15 11:53:42.871:OSPSSLConnect:***** SSL HANDSHAKE SUCCEED !!**** retry=2
```

When the SSL connection is closed, the following message appears:

```
*May 15 11:57:42.541:OSP:osp_ssl_close:OSPSSLClose succeed
```

The following are possible output trace messages:

```
osp_ssl_callback_add_session:session not found, add it.  
osp_ssl_callback_add_session:session found, but not equal, delete old one  
osp_ssl_callback_add_session:Copy new session data  
osp_ssl_callback_add_session:session found and equal. no add  
osp_ssl_callback_get_session:No Session exist  
osp_ssl_callback_get_session:Session found, copy to sslref length=756  
osp_ssl_callback_delete_session:session not found
```

These messages do not indicate an error but indicate the result of the operation.

To display actual error messages, enter the **debug voip settlement error** command.

# debug voip settlement transaction

To see all the attributes of the transactions on the settlement gateway, use the **debug voip settlement transaction** in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug voip settlement transaction**

**no debug voip settlement transaction**

---

**Syntax Description** This command has no arguments or keywords.

---

**Defaults** Disabled

---

**Command Modes** Privileged EXEC

---

Command History	Release	Modification
	12.0(4)XH1	This command was introduced.

---



---

**Examples** The following is sample output from the originating gateway:

```
00:44:54:OSP:OSPPTTransactionNew:trans=0, err=0
00:44:54:OSP:osp_authorize:authorizing trans=0, err=0
router>
00:45:05:OSP:stop_settlement_ccapi_accounting:send report for
callid=7, trans
=0, calling=5710868, called=15125551212, curr_Dest=1
00:45:05:OSP:OSPPTTransactionDelete:deleting trans=0
```

The following is sample output from the terminating gateway:

```
00:44:40:OSP:OSPPTTransactionNew:trans=0, err=0
00:44:40:OSP:osp_validate:validated trans=0, error=0, authorised=1
```

# debug vpdn

To troubleshoot Layer 2 Forwarding (L2F) or Layer 2 Tunnel Protocol (L2TP) virtual private dialup network (VPDN) tunneling events and infrastructure, use the **debug vpdn** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

```
debug vpdn { call { event | fsm } | error | event [disconnect] | l2tp-sequencing | l2x-data |
l2x-errors | l2x-events | l2x-packets | message | packet [detail | errors] | sss {error | event |
fsm}}
```

```
no debug vpdn { call { event | fsm } | error | event [disconnect] | l2tp-sequencing | l2x-data |
l2x-errors | l2x-events | l2x-packets | message | packet [detail | errors] | sss {error | event |
fsm}}
```

## Syntax Description

<b>call event</b>	Displays significant events in the VPDN call manager.
<b>call fsm</b>	Displays significant events in the VPDN call manager finite state machine (fsm).
<b>error</b>	Displays VPDN errors.
<b>event</b>	Displays VPDN events.
<b>disconnect</b>	(Optional) Displays VPDN disconnect events.
<b>l2tp-sequencing</b>	Displays significant events related to L2TP sequence numbers such as mismatches, resend queue flushes, and drops.
<b>l2x-data</b>	Displays errors that occur in data packets.
<b>l2x-errors</b>	Displays errors that occur in protocol-specific conditions.
<b>l2x-events</b>	Displays events resulting from protocol-specific conditions.
<b>l2x-packets</b>	Displays detailed information about control packets in protocol-specific conditions.
<b>message</b>	Displays VPDN interprocess messages.
<b>packet</b>	Displays information about VPDN packets.
<b>detail</b>	(Optional) Displays detailed packet information, including packet dumps.
<b>errors</b>	(Optional) Displays errors that occur in packet processing.
<b>sss error</b>	Displays debug information about VPDN Subscriber Service Switch (SSS) errors.
<b>sss event</b>	Displays debug information about VPDN SSS events.
<b>sss fsm</b>	Displays debug information about the VPDN SSS fsm.

## Command Modes

Privileged EXEC

## Command History

OS Release	Modification
12.0(23)S	This command was integrated into Cisco IOS Release 12.0(23)S.
S Release	Modification
12.2(22)S	This command was integrated into Cisco IOS Release 12.2(22)S.

T Release	Modification
11.2	This command was introduced.
12.0(5)T	Support was added for L2TP debugging messages. The <b>l2tp-sequencing</b> and <b>errors</b> keywords were added. The <b>l2f-errors</b> , <b>l2f-events</b> , and <b>l2f-packets</b> keywords were changed to <b>l2x-errors</b> , <b>l2x-events</b> , and <b>l2x-packets</b> .
12.2(4)T	Support was added for the <b>message</b> and <b>call {event   fsm}</b> keywords.
12.2(11)T	Support was added for the <b>detail</b> keyword.
12.2(13)T	Support was added for the <b>sss {error   event   fsm}</b> keywords.

### Usage Guidelines

Note that the **debug vpdn packet** and **debug vpdn packet detail** commands generate several debug operations per packet. Depending on the L2TP traffic pattern, these commands may cause the CPU load to increase to a high level that impacts performance.

### Examples

This section contains the following examples:

- [Debugging VPDN Events on a NAS—Normal L2F Operations](#)
- [Debugging VPDN Events on the Tunnel Server—Normal L2F Operations](#)
- [Debugging VPDN Events on the NAS—Normal L2TP Operations](#)
- [Debugging VPDN Events on the Tunnel Server—Normal L2TP Operations](#)
- [Debugging Protocol-Specific Events on the NAS—Normal L2F Operations](#)
- [Debugging Protocol-Specific Events on the Tunnel Server—Normal L2F Operations](#)
- [Debugging Errors on the NAS—L2F Error Conditions](#)
- [Debugging L2F Control Packets for Complete Information](#)
- [Debugging an L2TPv3 Xconnect Session—Normal Operations](#)

#### Debugging VPDN Events on a NAS—Normal L2F Operations

The network access server (NAS) has the following VPDN configuration:

```
vpdn-group 1
 request-dialin
  protocol l2f
  domain cisco.com
  initiate-to ip 172.17.33.125
 username nas1 password nas1
```

The following is sample output from the **debug vpdn event** command on a NAS when an L2F tunnel is brought up and Challenge Handshake Authentication Protocol (CHAP) authentication of the tunnel succeeds:

```
Router# debug vpdn event

%LINK-3-UPDOWN: Interface Async6, changed state to up
*Mar 2 00:26:05.537: looking for tunnel -- cisco.com --
*Mar 2 00:26:05.545: Async6 VPN Forwarding...
*Mar 2 00:26:05.545: Async6 VPN Bind interface direction=1
*Mar 2 00:26:05.553: Async6 VPN vpn_forward_user user6@cisco.com is forwarded
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to up
*Mar 2 00:26:06.289: L2F: Chap authentication succeeded for nas1.
```

The following is sample output from the **debug vpdn event** command on a NAS when the L2F tunnel is brought down normally:

```
Router# debug vpdn event

%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to down
%LINK-5-CHANGED: Interface Async6, changed state to reset
*Mar 2 00:27:18.865: Async6 VPN cleanup
*Mar 2 00:27:18.869: Async6 VPN reset
*Mar 2 00:27:18.873: Async6 VPN Unbind interface
%LINK-3-UPDOWN: Interface Async6, changed state to down
```

Table 262 describes the significant fields shown in the two previous displays. The output describes normal operations when an L2F tunnel is brought up or down on a NAS.

**Table 262** *debug vpdn event Field Descriptions for the NAS*

Field	Description
<b>Asynchronous interface coming up</b>	
%LINK-3-UPDOWN: Interface Async6, changed state to up	Asynchronous interface 6 came up.
looking for tunnel -- cisco.com -- Async6 VPN Forwarding...	Domain name is identified.
Async6 VPN Bind interface direction=1	Tunnel is bound to the interface. These are the direction values: <ul style="list-style-type: none"> <li>• 1—From the NAS to the tunnel server</li> <li>• 2—From the tunnel server to the NAS</li> </ul>
Async6 VPN vpn_forward_user user6@cisco.com is forwarded	Tunnel for the specified user and domain name is forwarded.
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to up	Line protocol is up.
L2F: Chap authentication succeeded for nas1.	Tunnel was authenticated with the tunnel password nas1.
<b>Virtual access interface coming down</b>	
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to down	Normal operation when the virtual access interface is taken down.
Async6 VPN cleanup Async6 VPN reset Async6 VPN Unbind interface	Normal cleanup operations performed when the line or virtual access interface goes down.

### Debugging VPDN Events on the Tunnel Server—Normal L2F Operations

The tunnel server has the following VPDN configuration, which uses nas1 as the tunnel name and the tunnel authentication name. The tunnel authentication name might be entered in a users file on an authentication, authorization, and accounting (AAA) server and used to define authentication requirements for the tunnel.

```
vpdn-group 1
 accept-dialin
 protocol l2f
```

```
virtual-template 1
terminate-from hostname nas1
```

The following is sample output from the **debug vpdn event** command on the tunnel server when an L2F tunnel is brought up successfully:

```
Router# debug vpdn event

L2F: Chap authentication succeeded for nas1.
Virtual-Access3 VPN Virtual interface created for user6@cisco.com
Virtual-Access3 VPN Set to Async interface
Virtual-Access3 VPN Clone from Vtemplate 1 block=1 filterPPP=0
%LINK-3-UPDOWN: Interface Virtual-Access3, changed state to up
Virtual-Access3 VPN Bind interface direction=2
Virtual-Access3 VPN PPP LCP accepted sent & rcv CONFACK
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access3, changed state to up
```

The following is sample output from the **debug vpdn event** command on a tunnel server when an L2F tunnel is brought down normally:

```
Router# debug vpdn event

%LINK-3-UPDOWN: Interface Virtual-Access3, changed state to down
Virtual-Access3 VPN cleanup
Virtual-Access3 VPN reset
Virtual-Access3 VPN Unbind interface
Virtual-Access3 VPN reset
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access3, changed state to down
```

Table 263 describes the fields shown in two previous outputs. The output describes normal operations when an L2F tunnel is brought up or down on a tunnel server.

**Table 263** debug vpdn event Field Descriptions for the Tunnel Server

Field	Description
<b>Tunnel coming up</b>	
L2F: Chap authentication succeeded for nas1.	PPP CHAP authentication status for the tunnel named nas1.
Virtual-Access3 VPN Virtual interface created for user6@cisco.com	Virtual access interface was set up on the tunnel server for the user user6@cisco.com.
Virtual-Access3 VPN Set to Async interface	Virtual access interface 3 was set to asynchronous for character-by-character transmission.
Virtual-Access3 VPN Clone from Vtemplate 1 block=1 filterPPP=0	Virtual template 1 was applied to virtual access interface 3.
%LINK-3-UPDOWN: Interface Virtual-Access3, changed state to up	Link status is set to up.
Virtual-Access3 VPN Bind interface direction=2	Tunnel is bound to the interface. These are the direction values: <ul style="list-style-type: none"> <li>1—From the NAS to the tunnel server</li> <li>2—From the tunnel server to the NAS</li> </ul>
Virtual-Access3 VPN PPP LCP accepted sent & rcv CONFACK	PPP link control protocol (LCP) configuration settings (negotiated between the remote client and the NAS) were copied to the tunnel server and acknowledged.

**Table 263** *debug vpdn event Field Descriptions for the Tunnel Server (continued)*

Field	Description
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access3, changed state to up	Line protocol is up; the line can be used.
<b>Tunnel coming down</b>	
%LINK-3-UPDOWN: Interface Virtual-Access3, changed state to down	Virtual access interface is coming down.
Virtual-Access3 VPN cleanup Virtual-Access3 VPN reset Virtual-Access3 VPN Unbind interface Virtual-Access3 VPN reset	Router is performing normal cleanup operations when a virtual access interface used for an L2F tunnel comes down.
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access3, changed state to down	Line protocol is down for virtual access interface 3; the line cannot be used.

**Debugging VPDN Events on the NAS—Normal L2TP Operations**

The following is sample output from the **debug vpdn event** command on the NAS when an L2TP tunnel is brought up successfully:

```
Router# debug vpdn event
```

```
20:19:17: L2TP: I SCCRQ from ts1 tnl 8
20:19:17: L2X: Never heard of ts1
20:19:17: Tnl 7 L2TP: New tunnel created for remote ts1, address 172.21.9.4
20:19:17: Tnl 7 L2TP: Got a challenge in SCCRQ, ts1
20:19:17: Tnl 7 L2TP: Tunnel state change from idle to wait-ctl-reply
20:19:17: Tnl 7 L2TP: Got a Challenge Response in SCCCN from ts1
20:19:17: Tnl 7 L2TP: Tunnel Authentication success
20:19:17: Tnl 7 L2TP: Tunnel state change from wait-ctl-reply to established
20:19:17: Tnl 7 L2TP: SM State established
20:19:17: Tnl/Cl 7/1 L2TP: Session FS enabled
20:19:17: Tnl/Cl 7/1 L2TP: Session state change from idle to wait-for-tunnel
20:19:17: Tnl/Cl 7/1 L2TP: New session created
20:19:17: Tnl/Cl 7/1 L2TP: O ICRP to ts1 8/1
20:19:17: Tnl/Cl 7/1 L2TP: Session state change from wait-for-tunnel to wait-connect
20:19:17: Tnl/Cl 7/1 L2TP: Session state change from wait-connect to established
20:19:17: Vi1 VPDN: Virtual interface created for bun1@cisco.com
20:19:17: Vi1 VPDN: Set to Async interface
20:19:17: Vi1 VPDN: Clone from Vtemplate 1 filterPPP=0 blocking
20:19:18: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
20:19:18: Vi1 VPDN: Bind interface direction=2
20:19:18: Vi1 VPDN: PPP LCP accepting rcv CONFACK
20:19:19: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
```

**Debugging VPDN Events on the Tunnel Server—Normal L2TP Operations**

The following is sample output from the **debug vpdn event** command on the tunnel server when an L2TP tunnel is brought up successfully:

```
Router# debug vpdn event
```

```
20:47:33: %LINK-3-UPDOWN: Interface Async7, changed state to up
20:47:35: As7 VPDN: Looking for tunnel -- cisco.com --
```

```

20:47:35: As7 VPDN: Get tunnel info for cisco.com with NAS nas1, IP 172.21.9.13
20:47:35: As7 VPDN: Forward to address 172.21.9.13
20:47:35: As7 VPDN: Forwarding...
20:47:35: As7 VPDN: Bind interface direction=1
20:47:35: Tnl/Cl 8/1 L2TP: Session FS enabled
20:47:35: Tnl/Cl 8/1 L2TP: Session state change from idle to wait-for-tunnel
20:47:35: As7 8/1 L2TP: Create session
20:47:35: Tnl 8 L2TP: SM State idle
20:47:35: Tnl 8 L2TP: Tunnel state change from idle to wait-ctl-reply
20:47:35: Tnl 8 L2TP: SM State wait-ctl-reply
20:47:35: As7 VPDN: bum1@cisco.com is forwarded
20:47:35: Tnl 8 L2TP: Got a challenge from remote peer, nas1
20:47:35: Tnl 8 L2TP: Got a response from remote peer, nas1
20:47:35: Tnl 8 L2TP: Tunnel Authentication success
20:47:35: Tnl 8 L2TP: Tunnel state change from wait-ctl-reply to established
20:47:35: Tnl 8 L2TP: SM State established
20:47:35: As7 8/1 L2TP: Session state change from wait-for-tunnel to wait-reply
20:47:35: As7 8/1 L2TP: Session state change from wait-reply to established
20:47:36: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async7, changed state to up

```

### Debugging Protocol-Specific Events on the NAS—Normal L2F Operations

The following is sample output from the **debug vpdn l2x-events** command on the NAS when an L2F tunnel is brought up successfully:

```
Router# debug vpdn l2x-events
```

```

%LINK-3-UPDOWN: Interface Async6, changed state to up
*Mar 2 00:41:17.365: L2F Open UDP socket to 172.21.9.26
*Mar 2 00:41:17.385: L2F_CONF received
*Mar 2 00:41:17.389: L2F Removing resend packet (type 1)
*Mar 2 00:41:17.477: L2F_OPEN received
*Mar 2 00:41:17.489: L2F Removing resend packet (type 2)
*Mar 2 00:41:17.493: L2F building nas2gw_mid0
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to up
*Mar 2 00:41:18.613: L2F_OPEN received
*Mar 2 00:41:18.625: L2F Got a MID management packet
*Mar 2 00:41:18.625: L2F Removing resend packet (type 2)
*Mar 2 00:41:18.629: L2F MID synced NAS/HG Clid=7/15 Mid=1 on Async6

```

The following is sample output from the **debug vpdn l2x-events** command on a NAS when an L2F tunnel is brought down normally:

```
Router# debug vpdn l2x-events
```

```

%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to down
%LINK-5-CHANGED: Interface Async6, changed state to reset
*Mar 2 00:42:29.213: L2F_CLOSE received
*Mar 2 00:42:29.217: L2F Destroying mid
*Mar 2 00:42:29.217: L2F Removing resend packet (type 3)
*Mar 2 00:42:29.221: L2F Tunnel is going down!
*Mar 2 00:42:29.221: L2F Initiating tunnel shutdown.
*Mar 2 00:42:29.225: L2F_CLOSE received
*Mar 2 00:42:29.229: L2F_CLOSE received
*Mar 2 00:42:29.229: L2F Got closing for tunnel
*Mar 2 00:42:29.233: L2F Removing resend packet
*Mar 2 00:42:29.233: L2F Closed tunnel structure
%LINK-3-UPDOWN: Interface Async6, changed state to down
*Mar 2 00:42:31.793: L2F Closed tunnel structure
*Mar 2 00:42:31.793: L2F Deleted inactive tunnel

```

[Table 264](#) describes the fields shown in the displays.

**Table 264** debug vpdn l2x-events Field Descriptions—NAS

Field	Descriptions
<b>Tunnel coming up</b>	
%LINK-3-UPDOWN: Interface Async6, changed state to up	Asynchronous interface came up normally.
L2F Open UDP socket to 172.21.9.26	L2F opened a User Datagram Protocol (UDP) socket to the tunnel server IP address.
L2F_CONF received	L2F_CONF signal was received. When sent from the tunnel server to the NAS, an L2F_CONF indicates the tunnel server's recognition of the tunnel creation request.
L2F Removing resend packet (type ...)	Removing the resend packet for the L2F management packet.  There are two resend packets that have different meanings in different states of the tunnel.
L2F_OPEN received	L2F_OPEN management message was received, indicating that the tunnel server accepted the NAS configuration of an L2F tunnel.
L2F building nas2gw_mid0	L2F is building a tunnel between the NAS and the tunnel server, using the Multiplex ID (MID) MID0.
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to up	Line protocol came up. Indicates whether the software processes that handle the line protocol regard the interface as usable.
L2F_OPEN received	L2F_OPEN management message was received, indicating that the tunnel server accepted the NAS configuration of an L2F tunnel.
L2F Got a MID management packet	MID management packets are used to communicate between the NAS and the tunnel server.
L2F MID synced NAS/HG Clid=7/15 Mid=1 on Async6	L2F synchronized the Client IDs on the NAS and the tunnel server, respectively. A multiplex ID is assigned to identify this connection in the tunnel.
<b>Tunnel coming down</b>	
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async6, changed state to down	Line protocol came down. Indicates whether the software processes that handle the line protocol regard the interface as usable.
%LINK-5-CHANGED: Interface Async6, changed state to reset	Interface was marked as reset.
L2F_CLOSE received	NAS received a request to close the tunnel.
L2F Destroying mid	Connection identified by the MID is being taken down.
L2F Tunnel is going down!	Advisory message about impending tunnel shutdown.
L2F Initiating tunnel shutdown.	Tunnel shutdown has started.
L2F_CLOSE received	NAS received a request to close the tunnel.
L2F Got closing for tunnel	NAS began tunnel closing operations.

**Table 264 debug vpdn l2x-events Field Descriptions—NAS (continued)**

Field	Descriptions
%LINK-3-UPDOWN: Interface Async6, changed state to down	Asynchronous interface was taken down.
L2F Closed tunnel structure	NAS closed the tunnel.
L2F Deleted inactive tunnel	Now-inactivated tunnel was deleted.

**Debugging Protocol-Specific Events on the Tunnel Server—Normal L2F Operations**

The following is sample output from the **debug vpdn l2x-events** command on a tunnel server when an L2F tunnel is created:

```
Router# debug vpdn l2x-events

L2F_CONF received
L2F Creating new tunnel for nas1
L2F Got a tunnel named nas1, responding
L2F Open UDP socket to 172.21.9.25
L2F_OPEN received
L2F Removing resend packet (type 1)
L2F_OPEN received
L2F Got a MID management packet
%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up
```

The following is sample output from the **debug vpdn l2x-events** command on a tunnel server when the L2F tunnel is brought down normally:

```
Router# debug vpdn l2x-events

L2F_CLOSE received
L2F Destroying mid
L2F Removing resend packet (type 3)
L2F Tunnel is going down!
L2F Initiating tunnel shutdown.
%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to down
L2F_CLOSE received
L2F Got closing for tunnel
L2F Removing resend packet
L2F Removing resend packet
L2F Closed tunnel structure
L2F Closed tunnel structure
L2F Deleted inactive tunnel
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to down
```

Table 265 describes the significant fields shown in the displays.

**Table 265 debug vpdn l2x-events Field Descriptions—Tunnel Server**

Field	Description
<b>Tunnel coming up</b>	
L2F_CONF received	L2F configuration is received from the NAS. When sent from a NAS to a tunnel server, the L2F_CONF is the initial packet in the conversation.
L2F Creating new tunnel for nas1	Tunnel named <i>nas1</i> is being created.
L2F Got a tunnel named nas1, responding	Tunnel server is responding.

**Table 265** debug vpdn l2x-events Field Descriptions— Tunnel Server (continued)

Field	Description
L2F Open UDP socket to 172.21.9.25	Opening a socket to the NAS IP address.
L2F_OPEN received	L2F_OPEN management message was received, indicating the NAS is opening an L2F tunnel.
L2F Removing resend packet (type ...)	Removing the resend packet for the L2F management packet.  The two resend packet types have different meanings in different states of the tunnel.
L2F Got a MID management packet	L2F MID management packets are used to communicate between the NAS and the tunnel server.
%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up	Tunnel server is bringing up virtual access interface 1 for the L2F tunnel.
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to up	Line protocol is up. The line can be used.
<b>Tunnel coming down</b>	
L2F_CLOSE received	NAS or tunnel server received a request to close the tunnel.
L2F Destroying mid	Connection identified by the MID is being taken down.
L2F Removing resend packet (type ...)	Removing the resend packet for the L2F management packet.  There are two resend packets that have different meanings in different states of the tunnel.
L2F Tunnel is going down! L2F Initiating tunnel shutdown.	Router is performing normal operations when a tunnel is coming down.
%LINK-3-UPDOWN: Interface Virtual-Access1, changed state to down	The virtual access interface is coming down.
L2F_CLOSE received L2F Got closing for tunnel L2F Removing resend packet L2F Removing resend packet L2F Closed tunnel structure L2F Closed tunnel structure L2F Deleted inactive tunnel	Router is performing normal cleanup operations when the tunnel is being brought down.
%LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-Access1, changed state to down	Line protocol is down; virtual access interface 1 cannot be used.

### Debugging Errors on the NAS—L2F Error Conditions

The following is sample output from the **debug vpdn errors** command on a NAS when the L2F tunnel is not set up:

```
Router# debug vpdn errors

%LINEPROTO-5-UPDOWN: Line protocol on Interface Async1, changed state to down
%LINK-5-CHANGED: Interface Async1, changed state to reset
%LINK-3-UPDOWN: Interface Async1, changed state to down
%LINK-3-UPDOWN: Interface Async1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async1, changed state to up
VPDN tunnel management packet failed to authenticate
VPDN tunnel management packet failed to authenticate
```

Table 266 describes the significant fields shown in the display.

**Table 266** debug vpdn error Field Descriptions for the NAS

Field	Description
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async1, changed state to down	Line protocol on the asynchronous interface went down.
%LINK-5-CHANGED: Interface Async1, changed state to reset	Asynchronous interface 1 was reset.
%LINK-3-UPDOWN: Interface Async1, changed state to down %LINK-3-UPDOWN: Interface Async1, changed state to up	Link from asynchronous interface 1 link went down and then came back up.
%LINEPROTO-5-UPDOWN: Line protocol on Interface Async1, changed state to up	Line protocol on the asynchronous interface came back up.
VPDN tunnel management packet failed to authenticate	Tunnel authentication failed. This is the most common VPDN error.  <b>Note</b> Verify the password for the NAS and the tunnel server name.  If you store the password on an AAA server, you can use the <b>debug aaa authentication</b> command.

The following is sample output from the **debug vpdn l2x-errors** command:

```
Router# debug vpdn l2x-errors

%LINK-3-UPDOWN: Interface Async1, changed state to up
L2F Out of sequence packet 0 (expecting 0)
L2F Tunnel authentication succeeded for cisco.com
L2F Received a close request for a non-existent mid
L2F Out of sequence packet 0 (expecting 0)
L2F packet has bogus1 key 1020868 D248BA0F
L2F packet has bogus1 key 1020868 D248BA0F
```

Table 267 describes the significant fields shown in the display.

**Table 267** *debug vpdn l2x-errors Field Descriptions*

Field	Description
%LINK-3-UPDOWN: Interface Async1, changed state to up	The line protocol on the asynchronous interface came up.
L2F Out of sequence packet 0 (expecting 0)	Packet was expected to be the first in a sequence starting at 0, but an invalid sequence number was received.
L2F Tunnel authentication succeeded for cisco.com	Tunnel was established from the NAS to the tunnel server, cisco.com.
L2F Received a close request for a non-existent mid	Multiplex ID was not used previously; cannot close the tunnel.
L2F Out of sequence packet 0 (expecting 0)	Packet was expected to be the first in a sequence starting at 0, but an invalid sequence number was received.
L2F packet has bogus l key 1020868 D248BA0F	Value based on the authentication response given to the peer during tunnel creation. This packet, in which the key does not match the expected value, must be discarded.
L2F packet has bogus l key 1020868 D248BA0F	Another packet was received with an invalid key value. The packet must be discarded.

**Debugging L2F Control Packets for Complete Information**

The following is sample output from the **debug vpdn l2x-packets** command on a NAS. This example displays a trace for a **ping** command:

```
Router# debug vpdn l2x-packets

L2F SENDING (17): D0 1 1 10 0 0 0 4 0 11 0 0 81 94 E1 A0 4
L2F header flags: 53249 version 53249 protocol 1 sequence 16 mid 0 cid 4
length 17 offset 0 key 1701976070
L2F RECEIVED (17): D0 1 1 10 0 0 0 4 0 11 0 0 65 72 18 6 5
L2F SENDING (17): D0 1 1 11 0 0 0 4 0 11 0 0 81 94 E1 A0 4
L2F header flags: 53249 version 53249 protocol 1 sequence 17 mid 0 cid 4
length 17 offset 0 key 1701976070
L2F RECEIVED (17): D0 1 1 11 0 0 0 4 0 11 0 0 65 72 18 6 5
L2F header flags: 57345 version 57345 protocol 2 sequence 0 mid 1 cid 4
length 32 offset 0 key 1701976070
L2F-IN Output to Async1 (16): FF 3 C0 21 9 F 0 C 0 1D 41 AD FF 11 46 87
L2F-OUT (16): FF 3 C0 21 A F 0 C 0 1A C9 BD FF 11 46 87
L2F header flags: 49153 version 49153 protocol 2 sequence 0 mid 1 cid 4
length 32 offset 0 key -2120949344
L2F-OUT (101): 21 45 0 0 64 0 10 0 0 FF 1 B9 85 1 0 0 3 1 0 0 1 8 0 62 B1
0 0 C A8 0 0 0 0 11 E E0 AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD
AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB
CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD
L2F header flags: 49153 version 49153 protocol 2 sequence 0 mid 1 cid 4
length 120 offset 3 key -2120949344
L2F header flags: 49153 version 49153 protocol 2 sequence 0 mid 1 cid 4
length 120 offset 3 key 1701976070
L2F-IN Output to Async1 (101): 21 45 0 0 64 0 10 0 0 FF 1 B9 85 1 0 0 1 1 0
0 3 0 0 6A B1 0 0 C A8 0 0 0 0 11 E E0 AB CD AB CD AB CD AB CD AB CD AB CD
AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB
CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD
```

Table 268 describes the significant fields shown in the display.

**Table 268** *debug vpdn l2x-packets Field Descriptions*

Field	Description
L2F SENDING (17)	Number of bytes being sent. The first set of “SENDING”...“RECEIVED” lines displays L2F keepalive traffic. The second set displays L2F management data.
L2F header flags:	Version and flags, in decimal.
version 53249	Version.
protocol 1	Protocol for negotiation of the point-to-point link between the NAS and the tunnel server is always 1, indicating L2F management.
sequence 16	Sequence numbers start at 0. Each subsequent packet is sent with the next increment of the sequence number. The sequence number is thus a free running counter represented modulo 256. There is a distinct sequence counter for each distinct MID value.
mid 0	Multiplex ID, which identifies a particular connection within the tunnel. Each new connection is assigned a MID currently unused within the tunnel.
cid 4	Client ID used to assist endpoints in demultiplexing tunnels.
length 17	Size in octets of the entire packet, including header, all fields pre-sent, and payload. Length does not reflect the addition of the checksum, if pre-sent.
offset 0	Number of bytes past the L2F header at which the payload data is expected to start. If it is 0, the first byte following the last byte of the L2F header is the first byte of payload data.
key 1701976070	Value based on the authentication response given to the peer during tunnel creation. During the life of a session, the key value serves to resist attacks based on spoofing. If a packet is received in which the key does not match the expected value, the packet must be silently discarded.
L2F RECEIVED (17)	Number of bytes received.
L2F-IN Otput to Async1 (16)	Payload datagram. The data came in to the VPDN code.
L2F-OUT (16):	Payload datagram sent out from the VPDN code to the tunnel.
L2F-OUT (101)	Ping payload datagram. The value 62 in this line is the ping packet size in hexadecimal (98 in decimal). The three lines that follow this line show ping packet data.

### Debugging an L2TPv3 Xconnect Session—Normal Operations

The following example shows output from the **debug vpdn** command for an L2TP version 3 (L2TPv3) xconnect session on an Ethernet interface:

```
Router# debug vpdn l2x-events
```

```
23:31:18: L2X: l2tun session [1669204400], event [client request], old state [open], new
state [open]
23:31:18: L2X: L2TP: Received L2TUN message <Connect>
23:31:18: Tnl/Sn58458/28568 L2TP: Session state change from idle to wait-for-tunnel
23:31:18: Tnl/Sn58458/28568 L2TP: Create session
23:31:18: Tnl58458 L2TP: SM State idle
23:31:18: Tnl58458 L2TP: O SCCRQ
23:31:18: Tnl58458 L2TP: Control channel retransmit delay set to 1 seconds
23:31:18: Tnl58458 L2TP: Tunnel state change from idle to wait-ctl-reply
23:31:18: Tnl58458 L2TP: SM State wait-ctl-reply
23:31:18: Tnl58458 L2TP: I SCCRP from router
23:31:18: Tnl58458 L2TP: Tunnel state change from wait-ctl-reply to established
23:31:18: Tnl58458 L2TP: O SCCCN to router tnlid 8012
23:31:18: Tnl58458 L2TP: Control channel retransmit delay set to 1 seconds
23:31:18: Tnl58458 L2TP: SM State established
23:31:18: Tnl/Sn58458/28568 L2TP: O ICRQ to router 8012/0
23:31:18: Tnl/Sn58458/28568 L2TP: Session state change from wait-for-tunnel to wait-reply
23:31:19: Tnl58458 L2TP: Control channel retransmit delay set to 1 seconds
23:31:20: %LINK-3-UPDOWN: Interface Ethernet2/1, changed state to up
23:31:21: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet2/1, changed state to
up
23:31:25: L2X: Sending L2TUN message <Connect OK>
23:31:25: Tnl/Sn58458/28568 L2TP: O ICCN to router 8012/35149
23:31:25: Tnl58458 L2TP: Control channel retransmit delay set to 1 seconds
23:31:25: Tnl/Sn58458/28568 L2TP: Session state change from wait-reply to established
23:31:25: L2X: l2tun session [1669204400], event [server response], old state [open], new
state [open]
23:31:26: Tnl58458 L2TP: Control channel retransmit delay set to 1 seconds
```

#### Related Commands

Command	Description
<b>debug aaa authentication</b>	Displays information on AAA/TACACS+ authentication.
<b>debug acircuit</b>	Displays events and failures related to attachment circuits.
<b>debug pppoe</b>	Display debugging information for PPPoE sessions.
<b>debug vpdn pppoe-data</b>	Displays data packets of PPPoE sessions.
<b>debug vpdn pppoe-error</b>	Displays PPPoE protocol errors that prevent a session from being established or errors that cause an established sessions to be closed.
<b>debug vpdn pppoe-events</b>	Displays PPPoE protocol messages about events that are part of normal session establishment or shutdown.
<b>debug vpdn pppoe-packet</b>	Displays each PPPoE protocol packet exchanged.
<b>debug xconnect</b>	Displays errors and events related to an xconnect configuration.

# debug vpdn pppoe-data

To display data packets of PPP over Ethernet (PPPoE) sessions, use the **debug vpdn pppoe-data** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug vpdn pppoe-data**

**no debug vpdn pppoe-data**

**Syntax Description** This command has no arguments or keywords.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.1(1)T	This command was introduced.
	12.2(13)T	This command was replaced by the <b>debug pppoe</b> command.

**Usage Guidelines** The **debug vpdn pppoe-data** command displays a large number of debug messages and should generally be used only on a debug chassis with a single active session.

**Examples** The following is sample output from the **debug vpdn pppoe-data** command:

```
Router# debug vpdn pppoe-data

6d20h:%LINK-3-UPDOWN:Interface Virtual-Access1, changed state to up
6d20h:PPPoE:OUT
  contiguous pak, size 19
    FF 03 C0 21 01 01 00 0F 03 05 C2 23 05 05 06 D3
    FF 2B DA
6d20h:PPPoE:IN
  particle pak, size 1240
    C0 21 01 01 00 0A 05 06 39 53 A5 17 00 00 00 00
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
    00 00 00 00 00 00 00 00
6d20h:PPPoE:OUT
  contiguous pak, size 14
    FF 03 C0 21 02 01 00 0A 05 06 39 53 A5 17
6d20h:PPPoE:OUT
  contiguous pak, size 19
    FF 03 C0 21 01 02 00 0F 03 05 C2 23 05 05 06 D3
    FF 2B DA
6d20h:PPPoE:IN
  particle pak, size 1740
    C0 21 02 02 00 0F 03 05 C2 23 05 05 06 D3 FF 2B
    DA 00 80 C2 00 07 00 00 00 10 7B 01 2C D9 00 B0
    C2 EB 10 38 88 64 11 00
6d20h:PPPoE:OUT
  contiguous pak, size 30
    FF 03 C2 23 01 06 00 1A 10 99 1E 6E 8F 8C F2 C6
    EE 91 0A B0 01 CB 89 68 13 47 61 6E 67 61
```

```

6d20h:PPPoE:IN
  particle pak, size 3840
    C2 23 02 06 00 24 10 E6 84 FF 3A A4 49 19 CE D7
    AC D7 D5 96 CC 23 B3 41 6B 61 73 68 40 63 69 73
    63 6F 2E 63 6F 6D 00 00
6d20h:PPPoE:OUT
  contiguous pak, size 8
    FF 03 C2 23 03 06 00 04
6d20h:PPPoE:OUT
  contiguous pak, size 14
    FF 03 80 21 01 01 00 0A 03 06 65 65 00 66
6d20h:PPPoE:IN
  particle pak, size 1240
    80 21 01 01 00 0A 03 06 00 00 00 00 49 19 CE D7
    AC D7 D5 96 CC 23 B3 41 6B 61 73 68 40 63 69 73
    63 6F 2E 63 6F 6D 00 00
6d20h:PPPoE:OUT
  contiguous pak, size 14
    FF 03 80 21 03 01 00 0A 03 06 65 65 00 67
6d20h:PPPoE:IN
  particle pak, size 1240
    80 21 02 01 00 0A 03 06 65 65 00 66 00 04 AA AA
    03 00 80 C2 00 07 00 00 00 10 7B 01 2C D9 00 B0
    C2 EB 10 38 88 64 11 00
6d20h:PPPoE:IN
  particle pak, size 1240
    80 21 01 02 00 0A 03 06 65 65 00 67 49 19 CE D7
    AC D7 D5 96 CC 23 B3 41 6B 61 73 68 40 63 69 73
    63 6F 2E 63 6F 6D 00 00
6d20h:PPPoE:OUT
  contiguous pak, size 14
    FF 03 80 21 02 02 00 0A 03 06 65 65 00 67
6d20h:%LINEPROTO-5-UPDOWN:Line protocol on Interface Virtual-Access1,
changed state to up
6d20h:PPPoE:OUT
  contiguous pak, size 16
    FF 03 C0 21 09 01 00 0C D3 FF 2B DA 4C 4D 49 A4
6d20h:PPPoE:IN
  particle pak, size 1440
    C0 21 0A 01 00 0C 39 53 A5 17 4C 4D 49 A4 AA AA
    03 00 80 C2 00 07 00 00 00 10 7B 01 2C D9 00 B0
    C2 EB 10 38 88 64 11 00
6d20h:PPPoE:IN
  particle pak, size 1440
    C0 21 09 01 00 0C 39 53 A5 17 00 00 00 00 00 00
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
    00 00 00 00 00 00 00 00

```

Table 269 describes the significant fields shown in the display.

**Table 269** debug vpdn pppoe-data Field Descriptions

Field	Descriptions
6d20h:%LINK-3-UPDOWN:Interface Virtual-Access1, changed state to up	Virtual access interface 1 came up.
6d20h:PPPoE:OUT	The host delivered a PPPoE session packet to the access concentrator.
6d20h:PPPoE:IN	The access concentrator received a PPPoE session packet.

**Table 269** *debug vpdn pppoe-data Field Descriptions (continued)*

Field	Descriptions
6d20h:%LINEPROTO-5-UPDOWN:Line protocol on Interface Virtual-Access1, changed state to up	Line protocol is up; the line can be used.
contiguous pak, size 19	Size 19 contiguous packet.
particle pak, size 1240	Size 1240 particle packet.

**Related Commands**

Command	Description
<b>debug pppoe</b>	Displays debugging information for PPPoE sessions.
<b>debug vpdn pppoe-error</b>	Displays PPPoE protocol errors that prevent a session from being established or errors that cause an established session to be closed.
<b>debug vpdn pppoe-events</b>	Displays PPPoE protocol messages about events that are part of normal session establishment or shutdown.
<b>debug vpdn pppoe-packet protocol (VPDN)</b>	Displays each PPPoE protocol packet exchanged. Specifies the L2TP that the VPDN subgroup will use.
<b>show vpdn</b>	Displays information about active L2F protocol tunnel and message identifiers in a VPDN.
<b>vpdn enable</b>	Enables virtual private dialup networking on the router and informs the router to look for tunnel definitions in a local database and on a remote authorization server (home gateway), if one is pre-sent.

## debug vpdn pppoe-error

To display PPP over Ethernet (PPPoE) protocol errors that prevent a session from being established or errors that cause an established sessions to be closed, use the **debug vpdn pppoe-error** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug vpdn pppoe-error**

**no debug vpdn pppoe-error**

**Syntax Description** This command has no arguments or keywords.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.1(1)T	This command was introduced.
	12.2(13)T	This command was replaced by the <b>debug pppoe</b> command.

**Examples** The following is a full list of error messages displayed by the **debug vpdn pppoe-error** command:

```

PPPOE:pppoe_acsys_err cannot grow packet
PPPoE:Cannot find PPPoE info
PPPoE:Bad MAC address:00b0c2eb1038
PPPOE:PADI has no service name tag
PPPoE:pppoe_handle_padi cannot add AC name/Cookie.
PPPoE:pppoe_handle_padi cannot grow packet
PPPoE:pppoe_handle_padi encap failed
PPPoE cannot create virtual access.
PPPoE cannot allocate session structure.
PPPoE cannot store session element in tunnel.
PPPoE cannot allocate tunnel structure.
PPPoE cannot store tunnel
PPPoE:VA221:No Session, Packet Discarded
PPPOE:Tried to shutdown a null session
PPPoE:Session already open, closing
PPPoE:Bad cookie:src_addr=00b0c2eb1038
PPPoE:Max session count on mac elem exceeded:mac=00b0c2eb1038
PPPoE:Max session count on vc exceeded:vc=3/77
PPPoE:Bad MAC address - dropping packet
PPPoE:Bad version or type - dropping packet

```

[Table 270](#) describes the significant fields shown in the display.

**Table 270** *debug vpdn pppoe-error Field Descriptions*

<b>Field</b>	<b>Descriptions</b>
PPPOE:pppoe_acsys_err cannot grow packet	Asynchronous PPPoE packet initialization error.
PPPoE:Cannot find PPPoE info	The access concentrator sends a PADO to the host.
PPPoE:Bad MAC address:00b0c2eb1038	The host was unable to identify the Ethernet MAC address.
PPPOE:PADI has no service name tag	PADI requires a service name tag.
PPPoE:pppoe_handle_padi cannot add AC name/Cookie.	pppoe_handle_padi could not append AC name.
PPPoE:pppoe_handle_padi cannot grow packet	pppoe_handle_padi could not append packet.
PPPoE:pppoe_handle_padi encap failed	pppoe_handle_padi could not specify PPPoE on ATM encapsulation.
PPPoE cannot create virtual access.	PPPoE session unable to verify virtual access interface.
PPPoE cannot allocate session structure.	PPPoE session unable to allocate Stage Protocol.
PPPoE cannot store session element in tunnel.	PPPoE tunnel cannot allocate session element.
PPPoE cannot allocate tunnel structure.	PPPoE tunnel unable to allocate Stage Protocol.
PPPoE cannot store tunnel	PPPoE configuration settings unable to initialize a tunnel.
PPPoE:VA221:No Session, Packet Discarded	No sessions created. All packets dropped.
PPPOE:Tried to shutdown a null session	Null session shutdown.
PPPoE:Session already open, closing	PPPoE session already open.
PPPoE:Bad cookie:src_addr=00b0c2eb1038	PPPoE session unable to append new cookie.
PPPoE:Max session count on mac elem exceeded:mac=00b0c2eb1038	The maximum number of sessions exceeded the Ethernet MAC address.
PPPoE:Max session count on vc exceeded:vc=3/77	The maximum number of sessions exceeded the PVC connection.
PPPoE:Bad MAC address - dropping packet	The host was unable to identify the MAC address. Packet dropped.
PPPoE:Bad version or type - dropping packet	The host was unable to identify the encapsulation type.

**Related Commands**

<b>Command</b>	<b>Description</b>
<b>debug pppoe</b>	Displays debugging information for PPPoE sessions.
<b>debug vpdn pppoe-data</b>	Displays data packets of PPPoE sessions.
<b>debug vpdn pppoe-events</b>	Displays PPPoE protocol messages about events that are part of normal session establishment or shutdown.
<b>debug vpdn pppoe-packet</b>	Displays each PPPoE protocol packet exchanged.
<b>protocol (VPDN)</b>	Specifies the L2TP that the VPDN subgroup will use.
<b>show vpdn</b>	Displays information about active L2F protocol tunnel and message identifiers in a VPDN.
<b>vpdn enable</b>	Enables virtual private dialup networking on the router and informs the router to look for tunnel definitions in a local database and on a remote authorization server (home gateway), if one is pre-sent.

# debug vpdn pppoe-events

To display PPP over Ethernet (PPPoE) protocol messages about events that are part of normal session establishment or shutdown, use the **debug vpdn pppoe-events** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug vpdn pppoe-events**

**no debug vpdn pppoe-events**

**Syntax Description** This command has no arguments or keywords.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.1(1)T	This command was introduced.
	12.2(13)T	This command was replaced by the <b>debug pppoe</b> command.

**Examples** The following is sample output from the **debug vpdn pppoe-events** command:

```
1w5d:IN PADI from PPPoE tunnel
1w5d:OUT PADO from PPPoE tunnel
1w5d:IN PADR from PPPoE tunnel
1w5d:PPPoE:VPN session created.
1w5d:%LINK-3-UPDOWN:Interface Virtual-Access2, changed state to up

1w5d:%LINEPROTO-5-UPDOWN:Line protocol on Interface Virtual-Access2, changed state to up
```

[Table 271](#) describes the significant fields shown in the display.

**Table 271** *debug vpdn pppoe-events* Field Descriptions

Field	Descriptions
1w5d:IN PADI from PPPoE tunnel	The access concentrator receives an Active Discovery Initiation (PADI) packet from the PPPoE tunnel.
1w5d:OUT PADO from PPPoE tunnel	The access concentrator sends an Active Discovery Offer (PADO) to the host.
1w5d:IN PADR from PPPoE tunnel	The host sends a single Active Discovery Request (PADR) to the access concentrator that it has chosen.
1w5d:PPPoE:VPN session created.	The access concentrator receives the PADR packet and creates a virtual private network (VPN) session.

**Table 271** *debug vpdn pppoe-events Field Descriptions (continued)*

Field	Descriptions
1w5d:%LINK-3-UPDOWN:Interface Virtual-Access2, changed state to up	Virtual access interface 2 came up.
1w5d:%LINEPROTO-5-UPDOWN:Line protocol on Interface Virtual-Access2, changed state to up	Line protocol is up. The line can be used.

**Related Commands**

Command	Description
<b>debug pppoe</b>	Displays debugging information for PPPoE sessions.
<b>debug vpdn pppoe-data</b>	Displays data packets of PPPoE sessions.
<b>debug vpdn pppoe-error</b>	Displays PPPoE protocol errors that prevent a session from being established or errors that cause an established session to be closed.
<b>debug vpdn pppoe-packet protocol (VPDN)</b>	Displays each PPPoE protocol packet exchanged. Specifies the L2TP that the VPDN subgroup will use.
<b>show vpdn</b>	Displays information about active L2F protocol tunnel and message identifiers in a VPDN.
<b>vpdn enable</b>	Enables virtual private dialup networking on the router and informs the router to look for tunnel definitions in a local database and on a remote authorization server (home gateway), if one is pre-sent.

# debug vpdn pppoe-packet

To display each PPP over Ethernet (PPPoE) protocol packet exchanged, use the **debug vpdn pppoe-packet** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

```
debug vpdn pppoe-packet
```

```
no debug vpdn pppoe-packet
```

**Syntax Description** This command has no arguments or keywords.

**Command Modes** Privileged EXEC

Command History	Release	Modification
	12.1(1)T	This command was introduced.
	12.2(13)T	This command was replaced by the <b>debug pppoe</b> command.

**Usage Guidelines** The **debug vpdn pppoe-packet** command displays a large number of debug messages and should generally only be used on a debug chassis with a single active session.

**Examples** The following is sample output from the **debug vpdn pppoe-packet** command:

```
PPPoE control packets debugging is on

1w5d:PPPoE:discovery packet
contiguous pak, size 74
  FF FF FF FF FF FF 00 10 7B 01 2C D9 88 63 11 09
  00 00 00 04 01 01 00 00 00 00 00 00 00 00 00
  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ...
1w5d:OUT PADO from PPPoE tunnel
contiguous pak, size 74
  00 01 09 00 AA AA 03 00 80 C2 00 07 00 00 00 10
  7B 01 2C D9 00 90 AB 13 BC A8 88 63 11 07 00 00
  00 20 01 01 00 00 01 02 00 04 41 67 6E 69 01 ...
1w5d:PPPoE:discovery packet
contiguous pak, size 74
  00 90 AB 13 BC A8 00 10 7B 01 2C D9 88 63 11 19
  00 00 00 20 01 01 00 00 01 02 00 04 41 67 6E 69
  01 04 00 10 B7 4B 86 5B 90 A5 EF 11 64 A9 BA ...
```

Table 272 describes the significant fields shown in the display.

**Table 272** *debug vpdn pppoe-packet Field Descriptions*

Field	Descriptions
PPPoE control packets debugging is on	PPPoE debugging of packets is enabled.
1w5d:PPPoE:discovery packet	The host performs a discovery to initiate a PPPoE session.
1w5d:OUT PADO from PPPoE tunnel	The access concentrator sends a PADO to the host.
1w5d:PPPoE:discovery packet	The host performs a discovery to initiate a PPPoE session.
contiguous pak, size 74	Size 74 contiguous packet.

#### Related Commands

Command	Description
<b>debug pppoe</b>	Displays debugging information for PPPoE sessions.
<b>debug vpdn pppoe-data</b>	Displays data packets of PPPoE sessions.
<b>debug vpdn pppoe-error</b>	Displays PPPoE protocol errors that prevent a session from being established or errors that cause an established session to be closed.
<b>debug vpdn pppoe-events</b>	Displays PPPoE protocol messages about events that are part of normal session establishment or shutdown.
<b>protocol (VPDN)</b>	Specifies the L2TP that the VPDN subgroup will use.
<b>show vpdn</b>	Displays information about active L2F protocol tunnel and message identifiers in a VPDN.
<b>vpdn enable</b>	Enables virtual private dialup networking on the router and informs the router to look for tunnel definitions in a local database and on a remote authorization server (home gateway), if one is pre-sent.

# debug vpm all

To enable all voice port module (VPM) debugging, use the **debug vpm all** command in privileged EXEC mode. To disable debugging output, use the **no** form of this command.

**debug vpm all**

**no debug vpm all**

**Syntax Description** This command has no arguments or keywords.

**Defaults** Disabled

**Command Modes** Privileged EXEC

## Command History

Release	Modification
11.3(1)T	This command was introduced for the Cisco 3600 series.
12.0(7)XK	This command was updated for the Cisco 2600, 3600, and Cisco MC3810 series devices.
12.1(2)T	This command was integrated into Cisco IOS Release 12.1(2)T.

## Usage Guidelines

Use the **debug vpm all** command to enable the complete set of VPM debugging commands: **debug vpm dsp**, **debug vpm error**, **debug vpm port**, **debug vpm spi**, and **debug vpm trunk\_sc**.

Execution of **no debug all** will turn off all port level debugging. It is usually a good idea to turn off all debugging and then enter the **debug** commands you are interested in one by one. This will help to avoid confusion about which ports you are actually debugging.

## Examples

For sample outputs, refer to the documentation of the other **debug vpm** commands.

## Related Commands

Command	Description
<b>debug vpm port</b>	Limits the <b>debug vpm all</b> command to a specified port.
<b>show debug</b>	Displays which debug commands are enabled.
<b>debug vpm error</b>	Enables DSP error tracing.
<b>debug vpm voaal2 all</b>	Enables the display of trunk conditioning supervisory component trace information.