StarOS中的L2TP - ASR5k上的實施和排除L2TP對等故障 — L2TP隧道中斷對等體無法訪問

目錄

[簡介](#)
[什麼是L2TP?](#)
[在移動性中，我們在哪裡使用它？](#)
[在此設定中什麼是ASR5x00?](#)
[L2TP LAC支援](#)
[L2TP LNS支援](#)
[在ASR5k上的思科裝置上啟用服務的配置](#)
[ASR5k上的LAC配置示例](#)
[ASR5k上的LNS配置示例](#)
[Cisco IOS裝置上的LNS配置示例](#)
[對對等體無法到達事件進行故障排除](#)
[使用案例:由於重試超時而導致初始隧道設定失敗](#)
[使用案例:由於keepalive而導致的初始隧道設定失敗](#)
[顯示輸出注意事項](#)

簡介

本檔案將介紹StarOS中的第2層通道通訊協定(L2TP)如何在ASR5k上實作，以及L2TP對等疑難排解— L2TPTunnelDownPeerUnreachable。

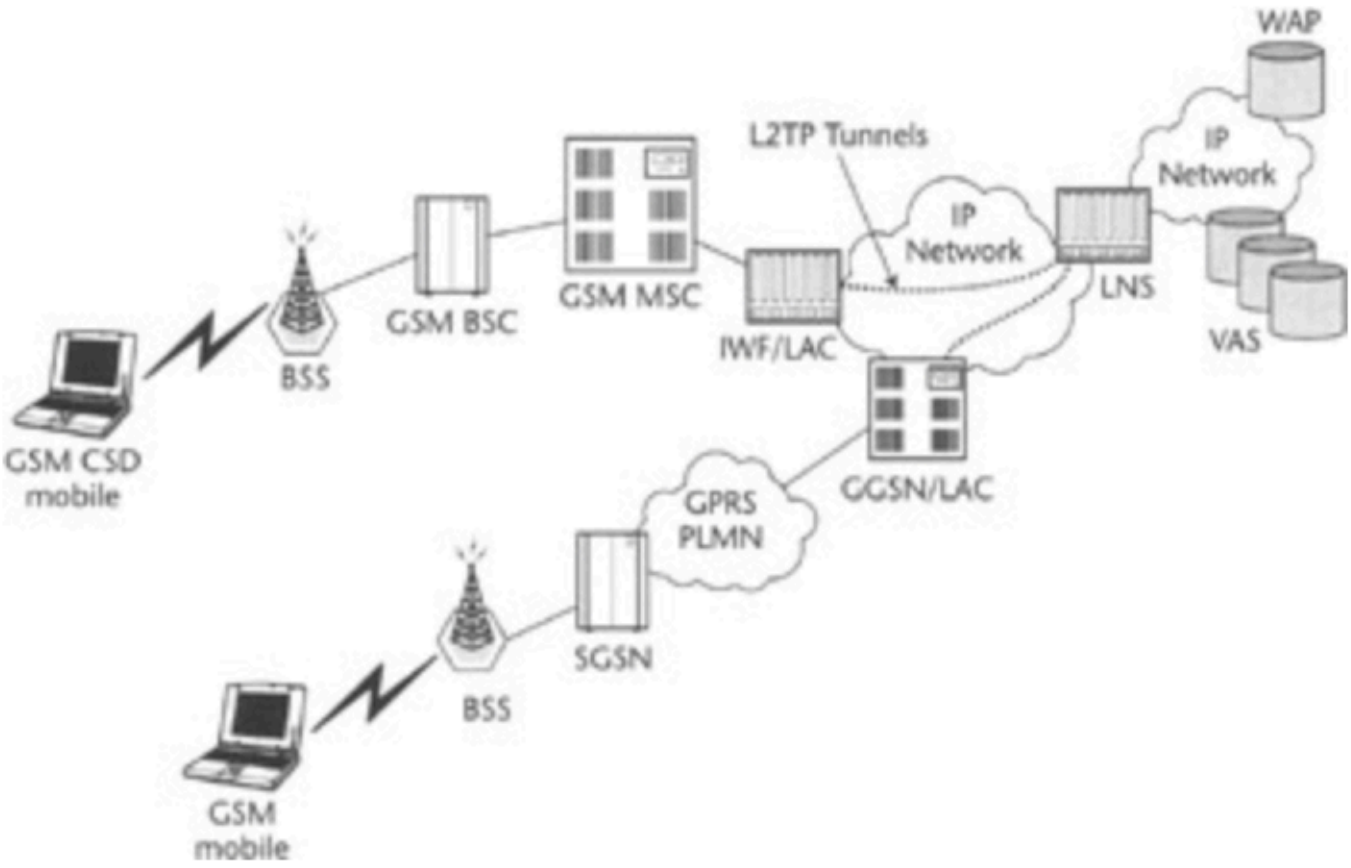
什麼是L2TP?

L2TP擴展了PPP的點對點性質。L2TP提供用於傳輸經隧道傳輸的PPP幀的封裝方法，允許通過分組交換網路傳輸PPP終端。L2TP最常部署在遠端訪問型別場景中，這些場景使用Internet提供Intranet型別的服務。其概念為虛擬私人網路(VPN)。

L2TP的兩個主要物理元素是L2TP訪問集中器(LAC)和L2TP網路伺服器(LNS):

- LAC:LAC是作為隧道端點一側的LNS的對等裝置。LAC終止遠端PPP連線並位於遠端和LNS之間。資料包通過PPP連線轉發到遠端連線或從遠端連線轉發。通過L2TP隧道轉發來往於LNS的資料包。
- LNS:LNS是作為隧道端點一側的LAC的對等裝置。LNS是LAC PPP隧道會話的終止點。這用於匯聚多個LAC隧道化PPP會話和進入專用網路。

簡化行動網路中的L2TP設定，如下圖所示。



L2TP使用兩種不同的消息型別：

- 控制消息：L2TP通過單獨的控制和資料通道傳遞控制和資料消息。帶內控制通道傳遞順序控制連線管理、呼叫管理、錯誤報告和會話控制消息。控制連線的啟動並不特定於LAC或LNS，而是特定於與控制連線建立相關的隧道發起方和接收方。在隧道端點之間使用共用金鑰質詢身份驗證方法。
- 資料消息：資料消息用於封裝傳送到L2TP隧道的PPP幀。

詳細的呼叫流程和隧道建立解釋如下：

http://www.cisco.com/c/en/us/support/docs/dial-access/virtual-private-dialup-network-vpdn/23980-l2tp-23980.html

在移動性中，我們在哪裡使用它？

典型的部署適用於企業使用者，其中GGSN充當LAC並建立通向企業網路中運行的LNS的安全隧道。GGSN配置指南的附錄中提供了詳細的呼叫流程，您可以在以下特定軟體版本中找到：

http://www.cisco.com/c/en/us/support/wireless/asr-5000-series/products-installation-and-configuration-guides-list.html

在此設定中什麼是ASR5x00?

ASR5k可支援LAC和LNS功能。

L2TP LAC支援

L2TP在LAC和LNS之間建立L2TP控制隧道，然後將使用者PPP連線作為L2TP會話建立隧道。LAC服務基於與GGSN相同的架構，並且受益於動態資源分配以及分散式消息和資料處理。此設計允許LAC服務支援每秒超過4000個設定，或最大吞吐量超過3G。單個通道中最多可以有65535個作業階段，而每個系統使用32,000個通道時，最多可以有500,000個L2TP作業階段。

L2TP LNS支援

設定為第2層通道通訊協定網路伺服器(LNS)的系統支援來自L2TP存取集中器(LAC)之間的終端安全虛擬私人網路(VPN)通道。

L2TP在LAC和LNS之間建立L2TP控制隧道，然後將使用者PPP連線作為L2TP會話建立隧道。在單個隧道中最多可以有65535個會話，每個LNS最多可以有500,000個會話。

LNS體系結構與GGSN類似，並利用解複用器的概念在平台上跨可用軟體和硬體資源智慧地分配新的L2TP會話，無需操作員干預。

有關詳細資訊，請參閱PGW/GGSN配置指南。

在ASR5k上啟用思科設備上的服務的配置

ASR5k上的LAC配置示例

```
 apn test-apn
accounting-mode none
     aaa group AAA
     authentication msisdn-auth
     ip context-name destination
     tunnel l2tp peer-address 1.1.1.1 local-hostname lac_l2tp

 configure
context destination-gi
lac-service l2tp_service
     allow called-number value apn
     peer-lns 1.1.1.1 encrypted secret pass
     bind address 1.1.1.2
```

ASR5k上的LNS配置示例

```
  configure
context destination-gi
lns-service lns-svc
bind address 1.1.1.1
authentication { { [ allow-noauth | chap < pref > | mschap < pref > | | pap < pref > | msid-auth
}
```

> **附註**：同一IP介面上的多個地址可以繫結到不同的LNS服務。但是，每個地址只能繫結到一個LNS服務。此外，LNS服務不能與其他服務（例如LAC服務）繫結到同一介面。

Cisco IOS裝置上的LNS配置示例

這可以作為Cisco IOS配置的支援配置示例，不受本文限制。

LNS配置

```
 aaa group server radius AAA
server 2.2.2.2 auth-port 1812 acct-port 1813
ip radius source-interface GigabitEthernet0/1
!

 aaa authentication login default local
aaa authentication ppp AAA group AAA
aaa authorization network AAA group AAA
aaa accounting network default
action-type start-stop
group radius

 vpdn-group vpdn
accept-dialin
protocol l2tp
virtual-template 10
l2tp tunnel password pass

 interface Virtual-Template10
ip unnumbered GigabitEthernet0/1
peer default ip address pool AAA
ppp authentication pap chap AAA
ppp authorization AAA
```

對對等體無法到達事件進行故障排除

本節將提供有關如何對網路中的L2TPTunnelDownPeerUnreachable事件進行故障排除的一些准則。此處介紹的是關於PDSN關閉的RP的資訊，但故障排除步驟與GGSN/PGW故障排除步驟相同。

作為提醒，建立了LAC到LNS隧道以包含使用者會話，同時它將使用者連線從PDSN/HA/GGSN/PGW擴展到LNS，LNS在終止處提供IP地址。如果在StarOS機箱上，LNS將從配置的IP池獲取IP地址。如果在某些其他LNS上（例如在客戶駐地），則IP地址由那裡的LNS提供。在後一種情況下，這可以有效地允許使用者通過漫遊合作夥伴上運行的LAC連線到其家庭網路。

LAC LNS隧道在嘗試設定第一個使用者會話時首先建立，並且只要隧道中存在會話，就會保持運行。

指定通道的最後一個作業階段結束時，該通道會關閉或關閉。可以在同一LAC-LNS對等體之間建立多個隧道。

以下是**show l2tp tunnels**命令的輸出片段，該片段在此案例中顯示，機箱同時承載LAC和LNS服務（TestLAC和TestLNS）。 請注意，LAC和LNS隧道全部具有會話，而某些封閉的RP隧道沒有會話。

```
 [local]1X-PDSN# show l2tp tunnels all | more
|+----State: (C) - Connected       (c) - Connecting
|            (d) - Disconnecting   (u) - Unknown
|
|
```

```
v  LocTun ID  PeerTun ID Active Sess Peer IPAddress  Service Name   Uptime
--- ---------- ---------- ---------- -------------- ------------   ---------
.............
C  30         1          511         214.97.107.28   TestLNS         00603h50m
C  31         56         468         214.97.107.28   TestLNS         00589h31m
C  10         105        81          79.116.237.27   TestLAC         00283h53m
C  29         16         453         79.116.231.27   TestLAC         00521h32m
C  106        218        63          79.116.231.27   TestLAC         00330h10m
C  107        6          464         79.116.237.27   TestLAC         00329h47m
C  30         35         194         214.97.107.28   TestLNS         00596h06m
```

服務配置可通過檢視

```
  show (lac-service | lns-service) name <lac or lns service name>
```

以下是使用LAC服務1.1.1.2和LNS服務（對等點）1.1.1的L2TPTunnelDownPeerUnreachable陷阱的範例

```
 Internal trap notification 92 (L2TPTunnelDownPeerUnreachable)  context destination service lac peer address 1.1.1.1 local address 1.1.1.2
```

使用**show snmp trap statistics**命令獲取觸發此陷阱的次數（自重新載入或上次重置統計資料以來）

當隧道設定超時或未響應保持連線(Hello)資料包時，會為L2TP觸發L2TPTunnelDownPeerUnreachable陷阱。其原因通常是由於LNS對等體未響應來自LAC的請求或任一方向的傳輸問題所致。

沒有陷阱指示對等體可訪問，如果不知道如何進一步調查，這可能導致調查時是否仍然出現問題（提交功能請求）的混亂。

要繼續操作，我們需要最重要的部分是對等IP地址。第一步是確儲存在可通過PING檢查的IP連線。如果存在連線，您可以繼續進行調試

```
****THIS IS TO BE RUN CAREFULLY and UPON verification of TAC/BU****


Active logging (exec mode) – logs written to terminal window

logging filter active facility l2tpmgr level debug
logging filter active facility l2tp-control level debug
logging active

To stop logging:

no logging active

Runtime logging (global config mode) – logs saved internally

logging filter runtime facility l2tpmgr level debug
logging filter runtime facility l2tp-control level debug

To view logs:

show logs (and/or check the syslog server if configured)
```

附註：

l2tpmgr跟蹤特定使用者會話設定

l2tp-control跟蹤隧道建立：

以下是此輸出的偵錯範例

使用案例:由於重試超時而導致初始隧道設定失敗

```
16:34:00.017 [l2tpmgr 48140 debug] [7/0/555 <l2tpmgr:1> l2tpmgr_call.c:591] [callid 4144ade2]
[context: destination, contextID: 3]  [software internal system] L2TPMgr-1  msid 0000012345
username laclnsuser  service <lac> - IPSEC tunnel does not exist
16:34:00.018 [l2tp-control 50069 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_fsm.c:105] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user] l2tp fsm: state
L2TPSNX_STATE_OPEN event L2TPSNX_EVNT_APP_NEW_SESSION

---------------------
16:34:00.018 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13660 to 1.1.1.1:1701 (138)
l2tp:[TLS](0/0)Ns=0,Nr=0 *MSGTYPE(SCCRQ) *PROTO_VER(1.0) *FRAMING_CAP(AS) *BEARER_CAP(AD)
TIE_BREAKER(0706050403020100) FIRM_VER(256) *HOST_NAME(lac) VENDOR_NAME(StarentNetworks)
*ASSND_TUN_ID(10) *RECV_WIN_SIZE(16) *CHALLENGE(dbed79cdc497f266bd374d427607cd52)
16:34:00.928 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13660 to 1.1.1.1:1701 (138)
l2tp:[TLS](0/0)Ns=0,Nr=0 *MSGTYPE(SCCRQ) *PROTO_VER(1.0) *FRAMING_CAP(AS) *BEARER_CAP(AD)
TIE_BREAKER(0706050403020100) FIRM_VER(256) *HOST_NAME(lac) VENDOR_NAME(StarentNetworks)
*ASSND_TUN_ID(10) *RECV_WIN_SIZE(16) *CHALLENGE(dbed79cdc497f266bd374d427607cd52)
16:34:02.943 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13660 to 1.1.1.1:1701 (138)
l2tp:[TLS](0/0)Ns=0,Nr=0 *MSGTYPE(SCCRQ) *PROTO_VER(1.0) *FRAMING_CAP(AS) *BEARER_CAP(AD)
TIE_BREAKER(0706050403020100) FIRM_VER(256) *HOST_NAME(lac) VENDOR_NAME(StarentNetworks)
*ASSND_TUN_ID(10) *RECV_WIN_SIZE(16) *CHALLENGE(dbed79cdc497f266bd374d427607cd52)
16:34:06.870 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13660 to 1.1.1.1:1701 (138)
l2tp:[TLS](0/0)Ns=0,Nr=0 *MSGTYPE(SCCRQ) *PROTO_VER(1.0) *FRAMING_CAP(AS) *BEARER_CAP(AD)
TIE_BREAKER(0706050403020100) FIRM_VER(256) *HOST_NAME(lac) VENDOR_NAME(StarentNetworks)
*ASSND_TUN_ID(10) *RECV_WIN_SIZE(16) *CHALLENGE(dbed79cdc497f266bd374d427607cd52)
16:34:14.922 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13660 to 1.1.1.1:1701 (138)
l2tp:[TLS](0/0)Ns=0,Nr=0 *MSGTYPE(SCCRQ) *PROTO_VER(1.0) *FRAMING_CAP(AS) *BEARER_CAP(AD)
TIE_BREAKER(0706050403020100) FIRM_VER(256) *HOST_NAME(lac) VENDOR_NAME(StarentNetworks)
*ASSND_TUN_ID(10) *RECV_WIN_SIZE(16) *CHALLENGE(dbed79cdc497f266bd374d427607cd52)
-------------------

 16:34:22.879 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13660 to 1.1.1.1:1701 (38)
l2tp:[TLS](0/0)Ns=1,Nr=0 *MSGTYPE(StopCCN) *RESULT_CODE(2/0) *ASSND_TUN_ID(10)
16:34:22.879 [l2tp-control 50069 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_fsm.c:105] [callid
4144ade2] [context: destination, contextID: 3]  [software internal user] l2tp fsm: state
L2TPSNX_STATE_WAIT_TUNNEL_ESTB event L2TPSNX_EVNT_PROTO_TUNNEL_DISCONNECTED
```

以下是觸發的SNMP陷阱，以匹配系統確定故障那一刻的上述日誌

```
 16:34:22 2009 Internal trap notification 92 (L2TPTunnelDownPeerUnreachable)  context
destination service lac peer address 1.1.1.1 local address 1.1.1.2
```

使用案例:由於重試超時而導致初始隧道設定失敗 — 分析

我們看到隧道在16:34出現，它嘗試傳送挑戰五次。顯然，沒有應答，隧道最終會斷開。

檢視配置預設值或配置值，然後檢視

```
 max-retransmission 5
retransmission-timeout-first 1
retransmission-timeout-max 8
```

此配置將被解釋為在1秒後第一次重新傳輸，然後指數增長 — 每次增加一倍：1、2、4、8、8。

請注意，術語max-retransmissions(5)包括第一次嘗試/傳輸。
retransmission-timeout-max是達到此限制後(if)兩次傳輸之間的最大時間量
retransmission-timeout-first是第一個重傳之前等待時間的起點。

因此，進行數學運算時，在預設引數的情況下，在1+ 2 + 4 + 8 + 8秒= 23秒後就會發生故障，如下面的輸出所示。

使用案例:由於keepalive而導致的初始隧道設定失敗

L2TPTunnelDownPeerUnreachable陷阱的另一個原因是沒有對keepalive-interval消息的響應。當沒有通過隧道傳送控制消息或資料時，會使用這些命令，以確保另一端仍然處於活動狀態。如果通道中有作業階段，但沒有任何動作，此指令可確保通道仍可正常運作，因為透過啟用，keepalive訊息會在設定的no封包交換期間（即60秒）後傳送，而且應該會有回應。在傳送第一個keepalive資料包後未收到響應的傳送頻率與上面介紹的隧道設定頻率相同。因此，在23秒內未收到對hello(keepalive)消息的響應後，隧道將被關閉。請參閱可設定的keepalive-interval（預設值為60s）。

下面是從監控使用者和日誌記錄成功進行保持連線交換的示例。請注意由於一分鐘內未傳輸任何使用者資料，消息集之間的間隔為一分鐘。在本示例中，LAC和LNS服務位於同一機箱中，分別位於名為**destination**和**lns**的情景中。

```
 INBOUND>>>>>  12:54:35:660 Eventid:50000(3)
L2TP Rx PDU, from 1.1.1.1:13660 to 1.1.1.2:13661 (20)
l2tp:[TLS](5/0)Ns=19,Nr=23 *MSGTYPE(HELLO)

<<<<OUTBOUND  12:54:35:661 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13660 (12)
l2tp:[TLS](1/0)Ns=23,Nr=20 ZLB

<<<<OUTBOUND  12:55:35:617 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13660 (20)
l2tp:[TLS](1/0)Ns=23,Nr=20 *MSGTYPE(HELLO)

INBOUND>>>>>  12:55:35:618 Eventid:50000(3)
L2TP Rx PDU, from 1.1.1.1:13660 to 1.1.1.2:13661 (12)
l2tp:[TLS](5/0)Ns=20,Nr=24 ZLB

 12:54:35.660 [l2tp-control 50001 debug] [7/0/555 <l2tpmgr:1> l2tpsnx_proto.c:1474] [callid
106478e8] [context: lns, contextID: 11]  [software internal user outbound protocol-log] L2TP Tx
PDU, from 1.1.1.1:13660 to 1.1.1.2:13661 (20) l2tp:[TLS](5/0)Ns=19,Nr=23 *MSGTYPE(HELLO)

12:55:35.618 [l2tp-control 50000 debug] [7/0/555 <l2tpmgr:1> l2tp.c:13050] [callid 106478e8]
[context: lns, contextID: 11]  [software internal user inbound protocol-log] L2TP Rx PDU, from
1.1.1.2:13661 to 1.1.1.1:13660 (20) l2tp:[TLS](1/0)Ns=23,Nr=20 *MSGTYPE(HELLO)
```

最後，以下範例顯示對於EXISTING通道，hello訊息沒有回應，且呼叫和通道已關閉。監控使用者

輸出：

```
 <<<<OUTBOUND  14:06:21:406 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)

<<<<OUTBOUND  14:06:22:413 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)

<<<<OUTBOUND  14:06:24:427 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)

<<<<OUTBOUND  14:06:28:451 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)

<<<<OUTBOUND  14:06:36:498 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)

<<<<OUTBOUND  14:06:44:446 Eventid:50001(3)
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (38)
l2tp:[TLS](2/0)Ns=5,Nr=2 *MSGTYPE(StopCCN) *RESULT_CODE(2/0) *ASSND_TUN_ID(6)
```

以下是各自的日誌。

請注意輸出Control tunnel timeout - retry-attempted five， last-interval 8000 ms，用於失敗的嘗試。

```
 14:06:21.406 [l2tp-control 50001 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_proto.c:1474] [callid
42c22625] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)
14:06:22.413 [l2tp-control 50001 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_proto.c:1474] [callid
42c22625] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)
14:06:24.427 [l2tp-control 50001 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_proto.c:1474] [callid
42c22625] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HE LLO)
14:06:28.451 [l2tp-control 50001 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_proto.c:1474] [callid
42c22625] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)
14:06:36.498 [l2tp-control 50001 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_proto.c:1474] [callid
42c22625] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (20)
l2tp:[TLS](2/0)Ns=4,Nr=2 *MSGTYPE(HELLO)
14:06:44.446 [l2tp-control 50068 warning] [7/0/9133 <l2tpmgr:2> l2tp.c:14841] [callid 42c22625]
[context: destination, contextID: 3]  [software internal user] L2TP (Local[svc: lac]: 6
Remote[1.1.1.1]: 2): Control tunnel timeout -  retry-attempted 5 , last-interval 8000 ms, Sr 2,
Ss 5, num-pkt-not-acked 1, Sent-Q-len 1, tun-recovery-flag 0, instance-recovery-flag 0, msg-type
Hello
14:06:44.446 [l2tp-control 50001 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_proto.c:1474] [callid
42c22625] [context: destination, contextID: 3]  [software internal user outbound protocol-log]
L2TP Tx PDU, from 1.1.1.2:13661 to 1.1.1.1:13661 (38)
l2tp:[TLS](2/0)Ns=5,Nr=2 *MSGTYPE(StopCCN) *RESULT_CODE(2/0) *ASSND_TUN_ID(6)
14:06:44.447 [l2tp-control 50069 debug] [7/0/9133 <l2tpmgr:2> l2tpsnx_fsm.c:105] [callid
```

```
42c22625] [context: destination, contextID: 3]  [software internal user] l2tp fsm: state
L2TPSNX_STATE_CONNECTED event L2TPSNX_EVNT_PROTO_SESSION_DISCONNECTED
```

和相應的SNMP陷阱

```
 14:06:44 2009 Internal trap notification 92 (L2TPTunnelDownPeerUnreachable)  context
destination service lac peer address 1.1.1.1 local address 1.1.1.2
```

顯示輸出注意事項

運行以下命令將指示特定對等體（或特定lac/lns服務中的所有隧道）是否存在對等體可達性問題

```
 show l2tp statistics (peer-address <peer ip address> | ((lac-service | lns-service) <lac or lns
service name>))
```

Active Connections計數器與可能存在多個對等點的現有隧道數匹配，如從較早版本的show l2tp tunnels輸出所示。

Failed to Connect計數器將指示發生了多少隧道安裝故障。
Max Retry Exceeded計數器可能是最重要的計數器，因為它表示由於超時而連線失敗（每個Retry exceeded都會導致L2TPTunnelDownPeerUnreachable陷阱）。 此資訊只告訴您給定對等體的問題發生頻率，而不告訴您發生超時的原因。但是，瞭解頻率有助於在整個故障排除過程中整合各個部分。

「會話」部分提供了訂戶會話級別（與隧道級別相比）的詳細資訊
Active Sessions計數器與特定對等體的show l2tp tunnels輸出的Active Sess列總和匹配（如果對等體有多個隧道）。
Failed to Connect計數器表示連線失敗的會話數。請注意，失敗的會話設定不會觸發L2TPTunnelDownPeerUnreachable陷阱，只有失敗的隧道設定才會觸發。

此外，還可以使用show l2tp tunnels命令的計數器版本。

```
 show l2tp tunnels counters peer-address <peer address>
```

最後，在會話級別，可以檢視給定對等體的所有訂戶。

```
 show l2tp sessions peer-address <peer ip address>
```

找到的訂閱者數量應與所討論的活動會話數量相匹配。