L2TP in StarOS - implementatie op het ASR5k-type en probleemoplossing in L2TP-fase - L2TPTunnelDownPeerOnbereikbaar

Inhoud

Inleiding

Dit document beschrijft hoe Layer 2 Tunneling Protocol (L2TP) in StarOS wordt geïmplementeerd op de ASR5k- en Troubleshoot L2TP-koppeling - L2TPTunnelDownPeerUnbereikbaar.

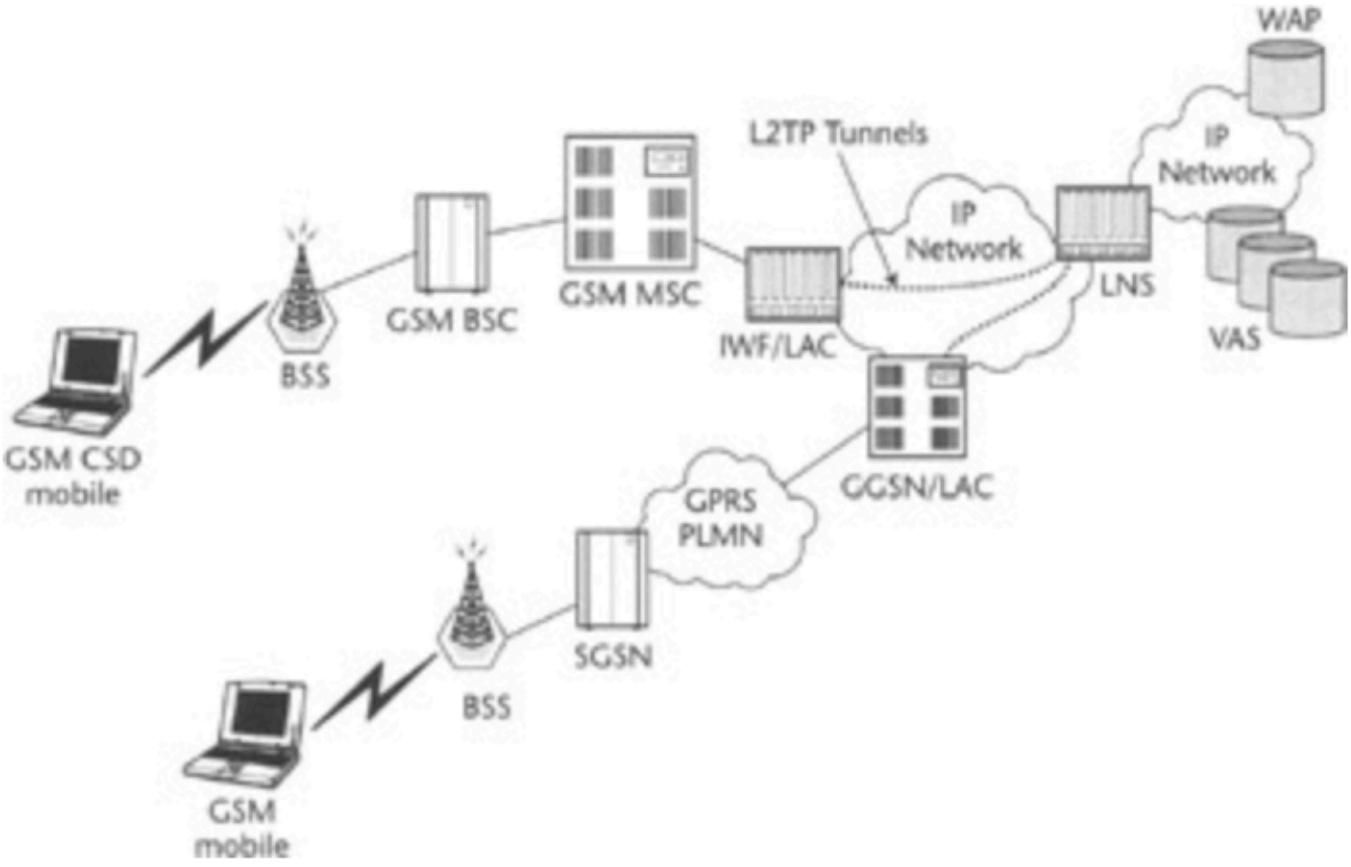
Wat is L2TP?

L2TP breidt de point-to-point aard van PPP uit. L2TP biedt een insluitingsmethode voor het verzenden van getunnelde PPP-frames, waardoor de PPP-endpoints via een pakketgeschakeld netwerk kunnen worden getunneld. L2TP wordt meestal ingezet in scenario's op afstand die het internet gebruiken om intranet-diensten aan te bieden. Het concept is van een Virtual Private Network (VPN).

De twee primaire fysieke elementen van L2TP zijn de L2TP Access Concentrator (LAC) en de L2TP Network Server (LNS):

- LAC: De LAC is een peer to the LNS die als één kant van het tunneleindpunt werkt. LAC beëindigt de externe PPP-verbinding en bevindt zich tussen de afstandsbediening en de LNS. Packets worden verzonden naar en van de externe verbinding via de PPP-verbinding. Packets naar en van de LNS worden doorgestuurd via de L2TP-tunnel.
- LNS: LNS is een peer to the LAC die als één kant van het tunneleindpunt fungeert. LNS is het eindpunt voor de LAC PPP verbonden sessies. Dit wordt gebruikt om de meerdere LAC-getunneled PPP sessies en ingangen naar het particuliere netwerk te bundelen.

Vereenvoudigde L2TP-instellingen in mobiel netwerk, zoals in deze afbeelding weergegeven.



Er zijn twee verschillende berichttypes die L2TP gebruikt:

- Besturingsberichten: L2TP passeert controle- en gegevensberichten via afzonderlijke controle- en gegevenskanalen. Het in-band controlekanaal passeert gesequentieerd beheer van de controle verbinding, vraagbeheer, foutmelding en sessiecontrole berichten. Het initiëren van de controleverbinding is niet specifiek voor de LAC of de LNS, maar eerder voor de tunnelinitiator en -ontvanger die relevant is voor de controleverbindingsinrichting. Er wordt een methode voor gedeelde geheime controle van de uitdagingen gebruikt tussen de eindpunten van de tunnel.
- Gegevensberichten: De gegevensberichten worden gebruikt om de PPP frames in te kapselen die in de L2TP-tunnel worden verstuurd.

De gedetailleerde telefoonstroom en tunnelvestiging wordt hier uitgelegd:

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

Waar gebruiken we het in mobiliteit?

De standaardinzet is voor zakelijke gebruikers waarbij de GGSN als LAC optreedt en beveiligde tunnels naar LNS instelt die in het bedrijfsnetwerk worden geëxploiteerd. Gedetailleerde callstromen zijn beschikbaar in de bijlage van de GGSN-configuratiegids die, per specifieke softwareversie, hier te vinden zijn:

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

Wat is ASR5x00 in deze instelling?

ASR5k kan LAC- en LNS-functies ondersteunen.

Ondersteuning van L2TP LAC

L2TP stelt L2TP-besturingtunnels tussen LAC en LNS in voordat u de abonnee PPP-verbindingen als L2TP-sessies tunnelt. De LAC-dienst is gebaseerd op dezelfde architectuur als het GGSN en profiteert van dynamische toewijzing van middelen en gedistribueerde berichten en gegevensverwerking. Dankzij dit ontwerp kan de LAC-service meer dan 4000 setup per seconde of een maximum van meer dan 3G doorvoersnelheid ondersteunen. Er kunnen maximaal 65535 sessies zijn in één tunnel en maximaal 500.000 L2TP-sessies met 32.000 tunnels per systeem.

Ondersteuning van L2TP LNS

Het systeem dat als Layer 2 Tunneling Protocol Network Server (LNS) wordt geconfigureerd ondersteunt de beveiligde Virtual Private Network (VPN)-tunnels tussen L2TP Access Concentrators (LAC's).

L2TP stelt L2TP-besturingtunnels tussen LAC en LNS in voordat u de abonnee PPP-verbindingen als L2TP-sessies tunnelt. Er kunnen maximaal 65535 sessies zijn in één tunnel en maximaal 500.000 sessies per LNS.

De LNS-architectuur is vergelijkbaar met de GGSN en gebruikt het concept van een de-multiplexer om op een intelligente manier nieuwe L2TP-sessies over de beschikbare software en hardwarebronnen op het platform toe te wijzen zonder tussenkomst van de exploitant.

Raadpleeg de GW/GGN-configuratiehandleidingen voor meer informatie.

Configuratie om de services op Cisco-apparaten op ASR5k in te schakelen

Configuratiemonster voor LAC op ASR5k

```
 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
```

Configuratievoorbeeld voor LNS op ASR5k

```
  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
}
```

> Opmerking: Meervoudige adressen op dezelfde IP interface kunnen aan verschillende LNS services worden gebonden. Elk adres kan echter aan slechts één LNS-service worden gebonden. Daarnaast kan de LNS-service niet aan dezelfde interface worden gebonden als andere services zoals een LAC-service.

Configuratievoorbeeld voor LNS op Cisco IOS-apparaat

Dit kan als ondersteunende configuratiesteekproef voor Cisco IOS worden gebruikt en is niet onderworpen aan dit artikel.

LNS-configuratie

```
 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
```

Onbereikbare gebeurtenis voor peer in probleemoplossing

In deze sectie worden enkele richtlijnen gegeven voor het oplossen van problemen met L2TPTunnelDownPeerUnbereikbaar gebeurtenis in het netwerk. Het wordt hier uitgelegd met betrekking tot PDSN gesloten RP maar de stappen voor probleemoplossing zijn hetzelfde wanneer er problemen worden opgelost met GSN/PGW.

Als herinnering wordt er een LAC naar LNS-tunnel gemaakt om abonnementssessies te bevatten terwijl de abonneeverbinding wordt uitgebreid van een PDSN/HA/GGSN/PGW naar de LNS waar deze wordt beëindigd en waar een IP-adres wordt opgegeven. Als er een StarOS-chassis is, krijgt de LNS een IP-adres uit een geconfigureerde IP-pool. Indien op een andere LNS, bijvoorbeeld in het klantengebouw, het IP-adres door de LNS daar wordt verstrekt. In het laatste scenario zou dit gebruikers in staat kunnen stellen om zich aan te sluiten op hun thuisnetwerk door middel van een LAC die op een roamingpartner loopt.

Eerst wordt een LAC LNS-tunnel aangemaakt omdat de eerste abonneesessie wordt geprobeerd te worden geïnstalleerd, en deze blijft omhoog zolang er sessies in de tunnel zitten.

Wanneer de laatste sessie eindigt voor een bepaalde tunnel, is die tunnel gesloten of gesloten. Er kan meer dan één tunnel worden ingericht tussen dezelfde LAC-LNS-peers.

Hier is een fragment van de output van de opdracht **toont l2tp tunnels** die dit in dit geval tonen de chassis zowel LAC als LNS services (TestLAC en TestLNS). Merk op dat de LAC en LNS tunnels allemaal sessies hebben, terwijl sommige gesloten RP tunnels geen sessies hebben.

```
 [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
```

Servicesconfiguratie kan worden bekeken met

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

Hier is een voorbeeld van de L2TPTunnelDownPeerOnbereikbare val met LAC-service 1.1.1.2 en LNS-service (peer) 1.1.1.1

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

Ontvang een telling van hoeveel keer deze val geactiveerd is (sinds herlading of laatste reset van statistieken) aan de **hand** van **de** opdracht **tongen statistieken**

De L2TPTunnelDownPeerOnbereikbare val wordt geactiveerd voor L2TP wanneer een tijdelijke versie van de tunnelinstallatie plaatsvindt of de (Hallo) pakketten in stand-houdt niet worden beantwoord. De oorzaak ligt meestal in het feit dat de LNS-peer niet reageert op verzoeken van de LAC of vervoerskwesties in een van beide richtingen.

Er is geen val om aan te geven dat de peer bereikbaar wordt, wat, indien niet wordt begrepen hoe verder onderzoek moet worden verricht, kan leiden tot verwarring over de vraag of er nog een probleem is of niet op het moment van onderzoek (verzoek om een kenmerk).

Om verder te gaan is het belangrijkste wat we nodig hebben het peer IP-adres. De eerste stap is om te verzekeren dat er IP connectiviteit is die met PING kan worden gecontroleerd. Als er connectiviteit is, kunt u met de diepten verder gaan

```
****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)
```

Opmerkingen:

l2TPr volgt specifieke setup-sessie

l2TP-tunnelbouw:

Hier volgt een voorbeeld van debug uit deze uitvoer

Gebruiksrechthoek: Eerste fout bij installatie van tunnelinstellingen door uitval opnieuw proberen

```
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
```

Hier is de resulterende SNMP-trap die is geactiveerd om de bovenstaande logs aan te passen op het moment dat het systeem de fout bepaalt

```
 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
```

Gebruiksrechthoek: Eerste fout bij installatie van tunnelinstellingen door tijdelijke uitval opnieuw proberen - Analyse

Wat we zien is dat de tunnel opkomt om 16:34 en het probeert de uitdaging vijf keer te sturen. Blijkbaar is er geen antwoord en uiteindelijk wordt de tunnel afgesloten.

Bekijk de standaardinstellingen voor de configuratie of de geconfigureerde waarden en zie

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

Deze configuratie moet worden geïnterpteerd als eerst na 1 seconde opnieuw verzenden, daarna exponentiële toename - elke keer verdubbeld: 1, 2, 4, 8, 8.

Merk op dat de term max-retransmissies (vijf) ook de eerste poging/transmissie omvat. retransmissie-timeout-max is maximale tijd tussen transmissies na (indien) deze limiet is bereikt retransmission-timeout-first is het beginpunt van hoe lang we moeten wachten voor de eerste retransmissie.

Dus, in het geval van de standaardparameters zou er een storing optreden na 1+ 2 + 4 + 8 seconden = 23 seconden, wat precies wordt gezien zoals in de output hieronder.

Gebruiksrechthoek: Eerste fout bij tunnelinstellingen vanwege keepaliën

De andere reden voor de L2TPTunnelDownPeerOnbereikbare val is geen reactie op de aanhoudende interval berichten. Deze worden gebruikt in periodes waarin geen controleberichten of gegevens over de tunnel worden verstuurd, om ervoor te zorgen dat het andere uiteinde nog in leven is. Als er sessies in de tunnel zijn, maar ze doen niets, dan zorgt deze opdracht ervoor dat de tunnel nog steeds goed werkt, omdat door hem in staat te stellen, er permanente berichten worden verstuurd na de geconfigureerde periode van geen pakketuitwisseling (d.w.z. 60 seconden), en er reacties worden verwacht. De frequentie van het verzenden van de keepaanval na het verzenden van de eerste en het niet krijgen van een antwoord is dezelfde als hierboven beschreven voor tunnelinstellingen. Dus na 23 seconden geen reactie te ontvangen op hallo (aanhoudende) berichten, zal de tunnel afgebroken worden. Zie Configureerbaar interval tussen beide (standaard= 60s).

Hier zijn voorbeelden van succesvolle Houd-levendige uitwisseling, zowel van monitor abonnee als houtkap. Noteer het interval van één minuut tussen berichten als gevolg van het feit dat er gedurende één minuut geen gebruikersgegevens worden verzonden. In dit voorbeeld bevinden de LAC - en LNS - diensten zich in hetzelfde chassis, in een context die respectievelijk **bestemming** en **lns** wordt genoemd.

```
 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)
```

Ten slotte, hier is een voorbeeld waar, voor een bestaande tunnel, hallo berichten niet worden beantwoord, en de vraag en tunnel worden afgebroken. Monitor Subscriber uitvoer:

```
 <<<<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)
```

Hier zijn de respectievelijke logs.

Merk op de tijdelijke versie van de uitvoertunnel - probeer het vijf, laatste-interval 8000 ms voor de mislukte pogingen.

```
 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
```

En corresponderende SNMP-trap

```
 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
```

Uitvoeroverwegingen weergeven

Het uitvoeren van de volgende opdracht zal aangeven of er problemen zijn met peer bereikbaarheid met een specifieke peer (of voor alle tunnels in een bepaalde lac/lns service)

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

```
service name>))
```
De Active Connections-teller komt overeen met het aantal bestaande tunnels voor die peer er meer dan één kan zijn, zoals gezien in de output van toont l2tp-tunnels vanuit vroeger.

De teller kan niet worden aangesloten en geeft aan hoeveel fouten in de tunnelinstellingen zijn opgetreden.
De Max Retry Overgegrepen teller is waarschijnlijk de belangrijkste teller, aangezien het op het falen wijst om te verbinden wegens een time-out (elke Retry overtrof resultaten in een L2TPTunnelDownPeerUnreachUnreach vangbare val). Deze informatie vertelt u alleen de frequentie van het probleem voor een bepaalde peer, het vertelt u niet waarom de time-out is opgetreden. Maar weten van de frequentie kan behulpzaam zijn bij het samenstellen van de onderdelen in het algehele proces voor het oplossen van problemen.

De sectie van de Sessies geeft details op het niveau van de abonneesessie (vs. tunnelniveau)
De actieve sessies komen overeen met de som van (als meer dan één tunnel voor een peer) de actieve uitvoer van de Sess kolom van tonen l2tp tunnels voor de bepaalde peer.
De teller kan niet worden aangesloten geeft aan hoeveel sessies niet zijn aangesloten. Merk op dat mislukte sessies NIET de L2TPTunnelDownPeerUnreach-val activeren, maar alleen mislukte tunnelinstellingen.

Er is ook een tellers versie van de show l2tp tunnels opdracht die nuttig kan zijn.

```
show l2tp tunnels counters peer-address <peer address>
```
Tenslotte kunnen op sessieniveau alle abonnees van een bepaalde peer worden bekeken.

```
show l2tp sessions peer-address <peer ip address>
```
Het aantal gevonden abonnees moet overeenkomen met het aantal actieve sessies zoals besproken.