Openserve L2TP configuration guide

Last updated : 27 April 2022

Disclaimer: Use at own risk. Terminology in this guide will not be 100% accurate.

Introduction:

A few people have asked me how this is done so I thought instead of answering to each person individually I should put some notes down to help ISPs.
This guide will help you understand the Openserve L2TP solution.

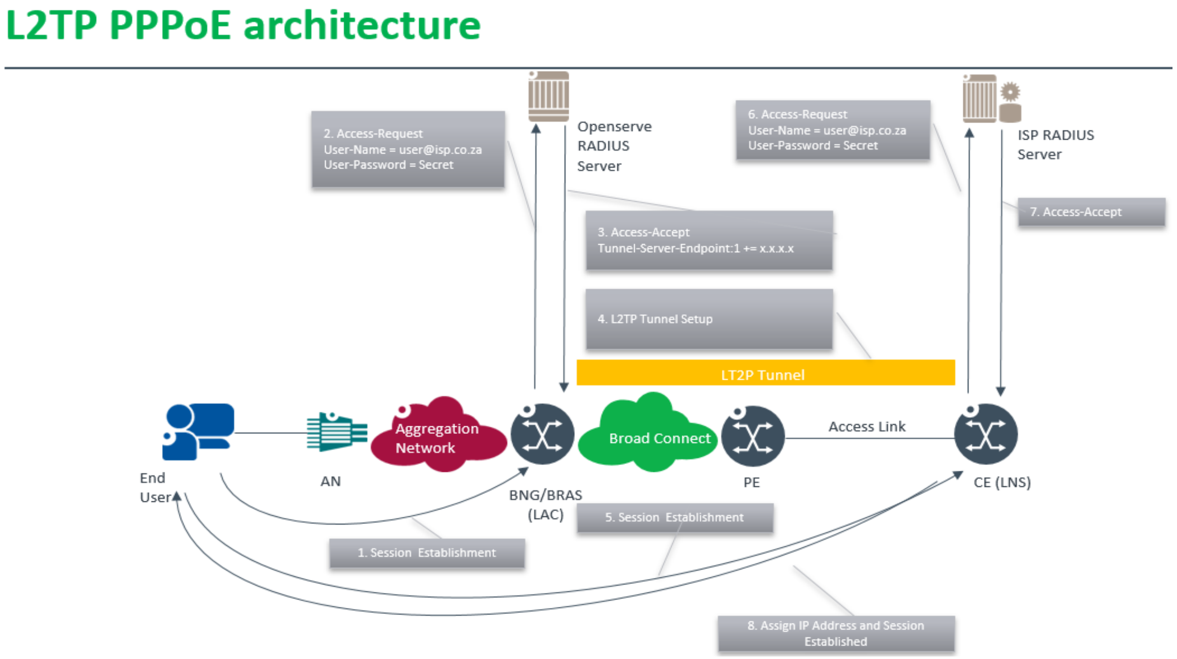
It is assumed you understand and have the current IPC setup with Openserve and have an IP Activator account and realm.

Thanks goes to Edrich de Lange for explaining it to me in the beginning and getting it going. Also thanks to the ZANOG community and the feedback on the mailing list.

Comments and corrections welcome. Email donald@networkstack.co.za

Overview

The below diagram is from the Openserve application explaining the architecture



L2TP PPPoE architecture

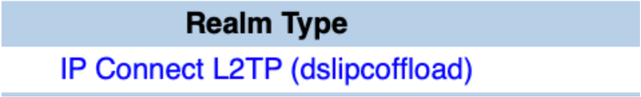
Customer end point will stay as is. They continue to do their PPPoE connection from the customer router with the same user/pass.

The customer pppoe connection is made to the Openserve BNG and then the Openserve BNG sends an Auth request to your LNS (L2TP Server) and if authentication passes it builds an L2TP tunnel/connection to your LNS

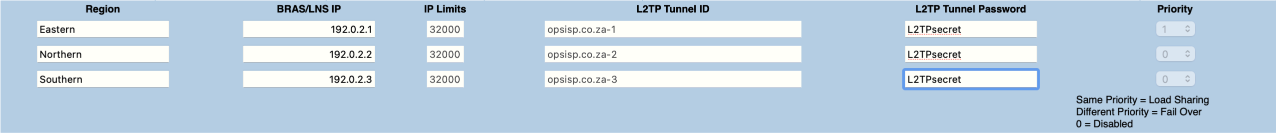
Setup

Step 1 - Realm Activation

A Realm on IP Activator is required and must be setup as “Realm Type” - IP Connect L2TP (dslipcoffload)



Realm Type
IP Connect L2TP (dslipcoffload)

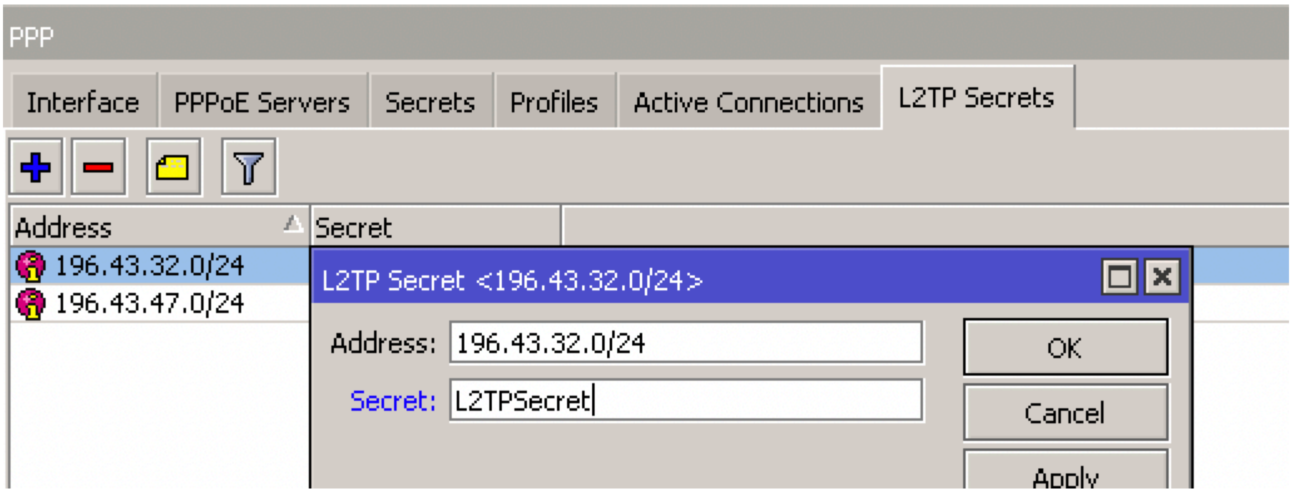


Region	BRAS/LNS IP	IP Limits	L2TP Tunnel ID	L2TP Tunnel Password	Priority
Eastern	192.0.2.1	32000	opsisp.co.za-1	L2TPsecret	1
Northern	192.0.2.2	32000	opsisp.co.za-2	L2TPsecret	0
Southern	192.0.2.3	32000	opsisp.co.za-3	L2TPsecret	0

Same Priority = Load Sharing
Different Priority = Fail Over
0 = Disabled

Step 2 - L2TP secret

Setup the LNS to receive the L2TP auth requests
Example here is for a Mikrotik



Address	Secret
196.43.32.0/24	
196.43.47.0/24	

L2TP Secret <196.43.32.0/24>

Address: 196.43.32.0/24

Secret: L2TPSecret

Step 3 - BGP filters

Adjust your BGP filters so that you accept the BNG addresses from openserve and you send them your LNS address

Openserve Upstream out
Send default route + your LNS address, like in the above example 192.0.2.1

Openserve Downstream In

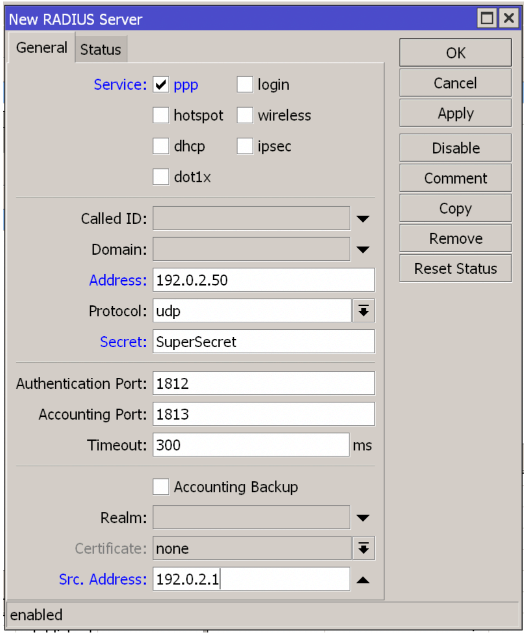
Accept their BNG addresses eg 196.43.32.0/24 le 32

The AAA session is not used because the authentication requests come directly from the openserve BNGs

Step 4 - User authentication

Setup your Mikrotik to send authentication to your management system

Authentication from openserve is standard RADIUS so you can setup your mikrotik to point to your RADIUS server where your user management resides


New RADIUS Server

General | Status

Service: ☑ ppp □ login □ hotspot □ wireless □ dhcp □ ipsec □ dot1x

Called ID:

Domain:

Address: 192.0.2.50

Protocol: udp

Secret: SuperSecret

Authentication Port: 1812

Accounting Port: 1813

Timeout: 300 ms

□ Accounting Backup

Realm:

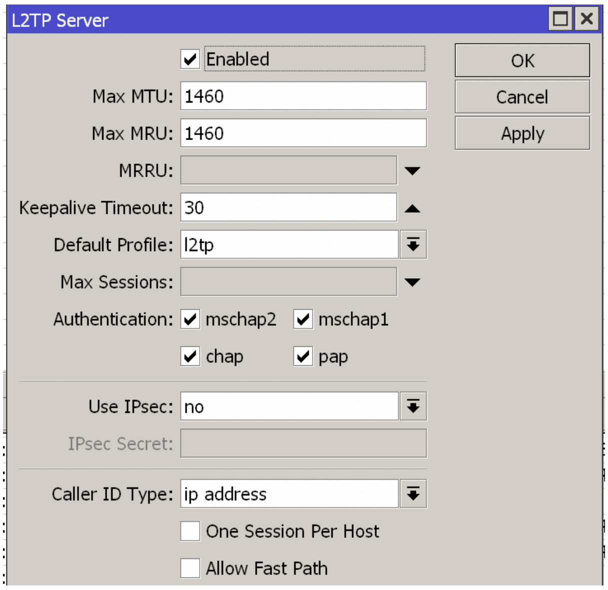
Certificate: none

Src. Address: 192.0.2.1

enabled

Step 5 - LNS setup

Setup your LNS by enabling the L2TP server


L2TP Server

☑ Enabled

Max MTU: 1460

Max MRU: 1460

MRRU:

Keepalive Timeout: 30

Default Profile: l2tp

Max Sessions:

Authentication: ☑ mschap2 ☑ mschap1 ☑ chap ☑ pap

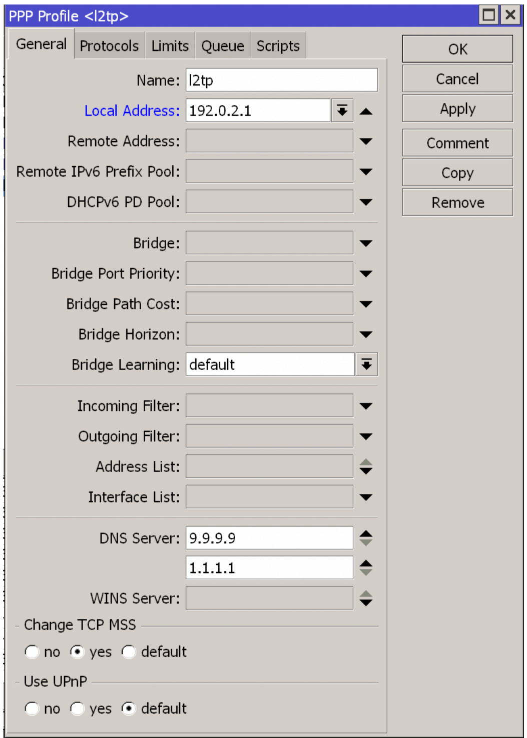
Use IPsec: no

IPsec Secret:

Caller ID Type: ip address

□ One Session Per Host

□ Allow Fast Path


PPP Profile <l2tp>

General | Protocols | Limits | Queue | Scripts

Name: l2tp

Local Address: 192.0.2.1

Remote Address:

Remote IPv6 Prefix Pool:

DHCPv6 PD Pool:

Bridge:

Bridge Port Priority:

Bridge Path Cost:

Bridge Horizon:

Bridge Learning: default

Incoming Filter:

Outgoing Filter:

Address List:

Interface List:

DNS Server: 9.9.9.9

1.1.1.1

WINS Server:

Change TCP MSS

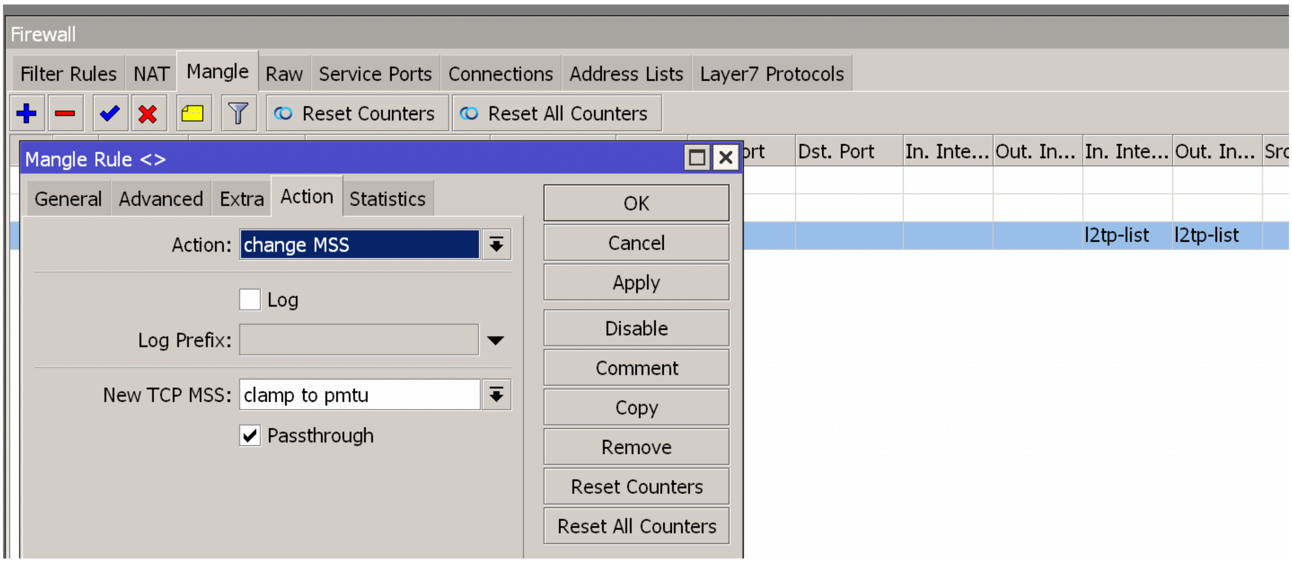
○ no ● yes ○ default

Use UPnP

○ no ○ yes ● default

I have setup a separate Default profile to use

For 1 of the setup’s I did I had to apply some clamping there is more than likely a better way to do this but this solved the speed issues.



Mangle Rule <>

Action: change MSS

☐ Log

Log Prefix:

New TCP MSS: clamp to pmtu

☑ Passthrough

Possible migration approach

IP Connect L2TP works on a per realm basis so you can test this using a test realm with a test username and password

-ends