New package for Layer 2 Peer-to-Peer VPN was added in Switchfin. It opens series of new features of an encrypted (twofish based) speech communication. In addition it can simplify the network configuration in case the PBX is behind symmetric or asymmetric NAT. To confirm the basic functionality I have done a simple test. Let me describe it here.

For establishing of the simplest possible VPN tunnel apart of the two communicating **Edge Nodes**
one extra
Super Node
is needed.

Initial tests was done using three BR4-Appliances manufactured by SwitchVoice Ltd . An example is shown below. The prompts of the BR4-Appliane devices was annotated so it is clear where the specific command is executed.

Let's start the **Supernode** service on one of the BR4-Appliance. The IP of this Appliance is **192.168.1.99**
and the
Supernode
service is listening on TCP/UDP port
20
. For this simple test all of the Appliances are in the local network. In reality
Edge Nodes
should have connectivity to the listening port of the
Supernode

root@br4_supernode:~> supernode -l 20
06/Sep/2009 11:33:15 [supernode.c: 477] Supernode ready: listening on port 20 [TCP/UDP]

Let’s start creating the VPN network.

root@br4_edgenode1:~> edge -a 10.1.2.1 -c mynetwork -k encryptme -l 192.168.1.99:20 &
06/Sep/2009 07:35:20 [edge.c:1138] Using supernode 192.168.1.99:20
06/Sep/2009 07:35:20 [tuntap_linux.c: 38] Interface edge0 has MAC 66:D7:A3:3E:BE:94
06/Sep/2009 07:35:20 [edge.c: 670] Registering with supernode
06/Sep/2009 07:35:20 [edge.c:1370]
06/Sep/2009 07:35:20 [edge.c:1371] Ready
06/Sep/2009 07:35:20 [edge.c:1037] Received REGISTER_ACK from remote peer [ip=192.168.1.99:20]
06/Sep/2009 07:35:20 [edge.c:1437] STATUS: pending=0, operational=0

In the **edge** command above **mynetwork** is the name of the created VPN nethwork and **encryptme** is the encryption key associated with this **Edge Node**.

New virtual Ethernet interface **edge0** is created.

root@br4_edgenode1:~> ifconfig
edge0 Link encap:Ethernet HWaddr 5A:9C:0C:2B:ED:08
inet addr:10.1.2.1 Bcast:10.1.2.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1400 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:500

eth0 Link encap:Ethernet HWaddr 00:09:45:56:21:A0
inet addr:192.168.1.101 Bcast:192.168.1.255 Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:86 errors:0 dropped:0 overruns:0 frame:0
TX packets:60 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0

Lets do the same for the second **Edge Node** but this time will give different VPN IP address 10.1.2.2

root@br4_edgenode2:~> edge -a 10.1.2.2 -c mynetwork -k encryptme -l 192.168.1.99:20 &

Now we can ping one of the **Edge Node** from the other through the new VPN tunnel, and vice versa.

root@br4_edgenode1:~> ping 10.1.2.2
PING 10.1.2.2 (10.1.2.2): 56 data bytes
64 bytes from 10.1.2.2: seq=0 ttl=64 time=3.237 ms
64 bytes from 10.1.2.2: seq=1 ttl=64 time=3.010 ms
64 bytes from 10.1.2.2: seq=2 ttl=64 time=2.980 ms
…

root@br4_edgenode2:~> ping 10.1.2.1
PING 10.1.2.2 (10.1.2.1): 56 data bytes
64 bytes from 10.1.2.1: seq=0 ttl=64 time=3.237 ms
64 bytes from 10.1.2.1: seq=1 ttl=64 time=3.010 ms
…

The new package can be used for easy creation of a secure voice channels.
For more information about the actual VPN implementation please take a look at n2n

Dimitar Penev