Workshop 4 – Part 1 : Configuring a Virtual Private Network

What Is A VPN:

VPN stands for Virtual Private Network. It allows you to connect two or more remote networks securely over an insecure connection. The reason it is called virtual is that there is no physical connection between the two networks. Instead a non-dedicated secure tunnel is created through an insecure connection (like the Internet). This tunnel is generally encrypted and is only decrypted when it arrives at the destination host.

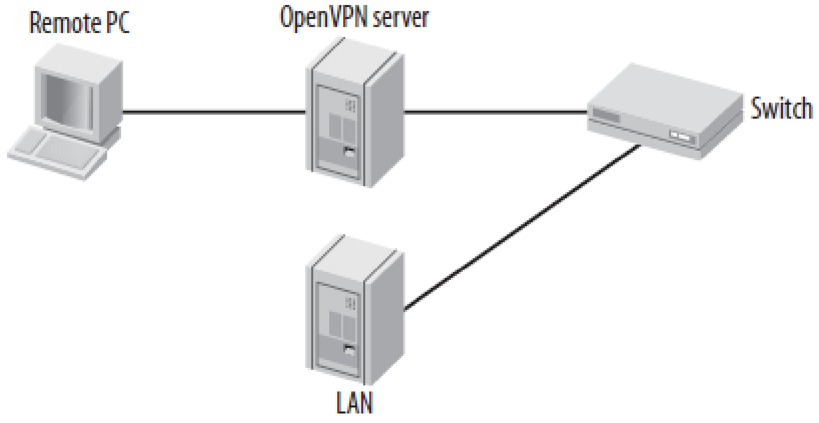
VPN Types:

There are many ways to implement a VPN. Many companies use proprietary software implementations while others use their routers because many routers have VPN support built in. These VPNs can be as simple as an SSH tunnel or very complicated. But all implementations create a virtual tunnel through an insecure network. Some VPN implementations include:

- IPSEC
- VPND
- SSH
- Many Cisco Routers (or other proprietary implementations)
- SSL/TLS

Here we will outline the implementations of OpenVPN.

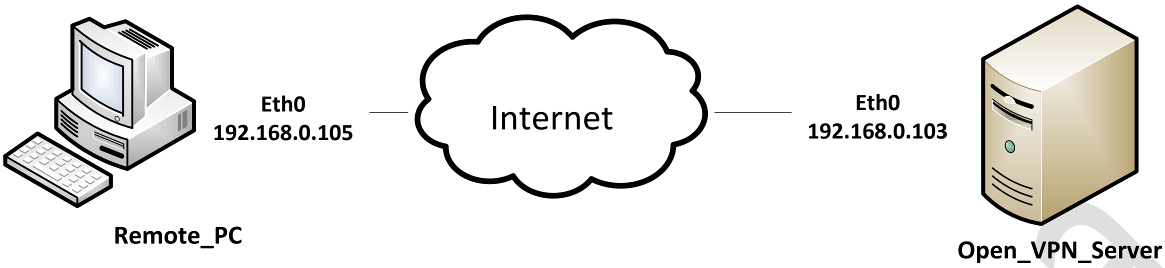
1- Open VPN test Lab:



Here we focus on two computers which one is Open VPN server (a Router) and the second is remote pc that could use open vpn server to connect to the LAN.

2- Set up machine:

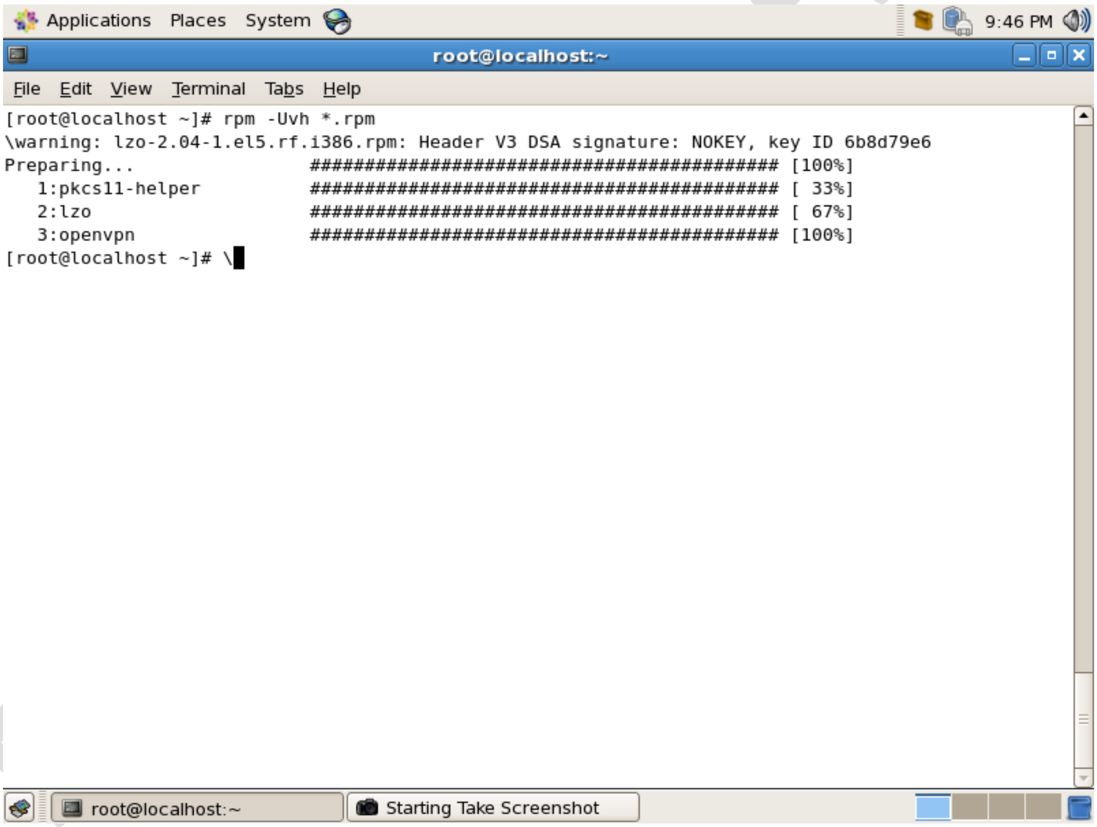
Now add two machine in to your system (Remote_PC) & (Open_VPN_Server) each with one bridged network interface on your virtual environment.



Note: IPs for this two node must be a valid internet addresses but we use CIDR range on this scenario.

3- Install Open_VPN packages:

Install open vpn packages on both server and client as below:

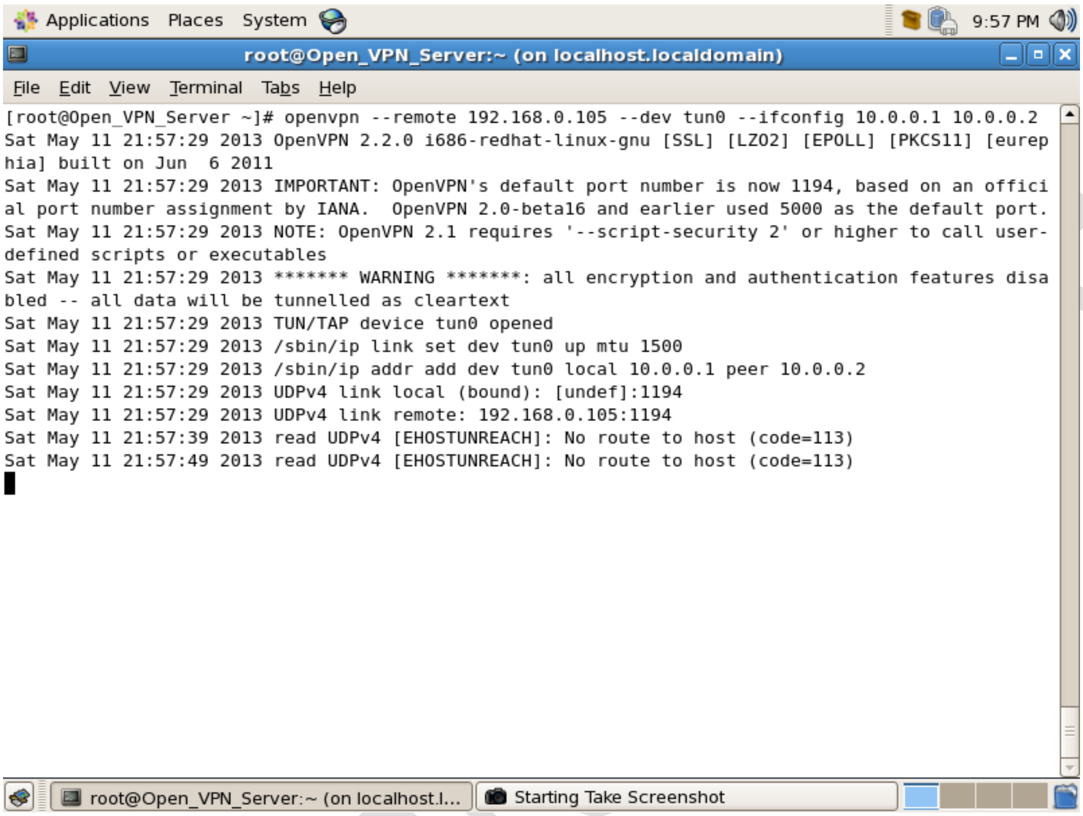


```
[root@localhost ~]# rpm -Uvh *.rpm
\warning: lzo-2.04-1.el5.rf.i386.rpm: Header V3 DSA signature: NOKEY, key ID 6b8d79e6
Preparing...                ########################################### [100%]
   1:pkcs11-helper          ########################################### [ 33%]
   2:lzo                    ########################################### [ 67%]
   3:openvpn                ########################################### [100%]
[root@localhost ~]# \
```

4- Starting and Testing OpenVPN:

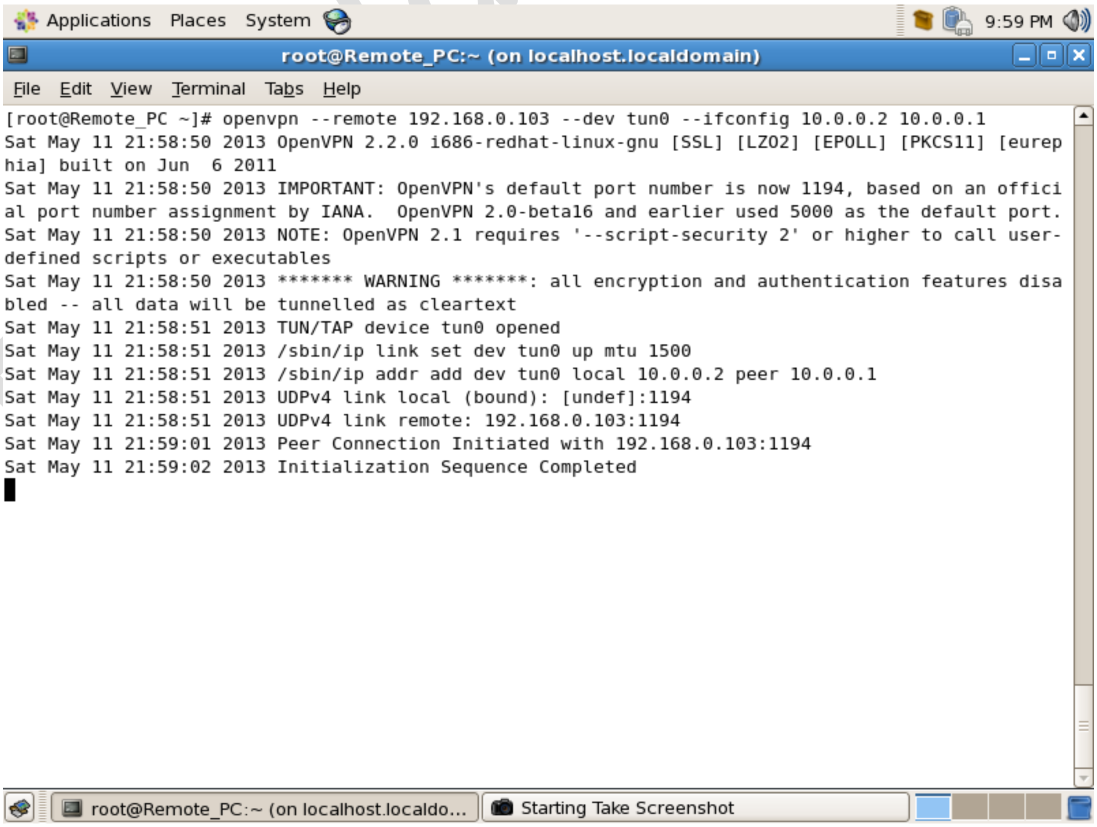
root@Open_Vpn_server :~# echo “1” > /proc/sys/net/ipv4/ip_forward

root@Open_Vpn_server :~# openvpn --remote 192.168.0.105 --dev tun0 --ifconfig 10.0.0.1 10.0.0.2


Applications Places System
9:57 PM
root@Open_VPN_Server:~ (on localhost.localdomain)
File Edit View Terminal Tabs Help
root@Open_VPN_Server:~ (on localhost.l...
Starting Take Screenshot

```
[root@Open_VPN_Server ~]# openvpn --remote 192.168.0.105 --dev tun0 --ifconfig 10.0.0.1 10.0.0.2
Sat May 11 21:57:29 2013 OpenVPN 2.2.0 i686-redhat-linux-gnu [SSL] [LZO2] [EPOLL] [PKCS11] [eurep
hia] built on Jun  6 2011
Sat May 11 21:57:29 2013 IMPORTANT: OpenVPN's default port number is now 1194, based on an offici
al port number assignment by IANA.  OpenVPN 2.0-beta16 and earlier used 5000 as the default port.
Sat May 11 21:57:29 2013 NOTE: OpenVPN 2.1 requires '--script-security 2' or higher to call user-
defined scripts or executables
Sat May 11 21:57:29 2013 ******* WARNING *******: all encryption and authentication features disa
bled -- all data will be tunnelled as cleartext
Sat May 11 21:57:29 2013 TUN/TAP device tun0 opened
Sat May 11 21:57:29 2013 /sbin/ip link set dev tun0 up mtu 1500
Sat May 11 21:57:29 2013 /sbin/ip addr add dev tun0 local 10.0.0.1 peer 10.0.0.2
Sat May 11 21:57:29 2013 UDPv4 link local (bound): [undef]:1194
Sat May 11 21:57:29 2013 UDPv4 link remote: 192.168.0.105:1194
Sat May 11 21:57:39 2013 read UDPv4 [EHOSTUNREACH]: No route to host (code=113)
Sat May 11 21:57:49 2013 read UDPv4 [EHOSTUNREACH]: No route to host (code=113)
```

root@Remote_Pc:~# openvpn --remote 192.168.0.103 --dev tun0 --ifconfig 10.0.0.2 10.0.0.1


Applications Places System
9:59 PM
root@Remote_PC:~ (on localhost.localdomain)
File Edit View Terminal Tabs Help
root@Remote_PC:~ (on localhost.localdo...
Starting Take Screenshot

```
[root@Remote_PC ~]# openvpn --remote 192.168.0.103 --dev tun0 --ifconfig 10.0.0.2 10.0.0.1
Sat May 11 21:58:50 2013 OpenVPN 2.2.0 i686-redhat-linux-gnu [SSL] [LZO2] [EPOLL] [PKCS11] [eurep
hia] built on Jun  6 2011
Sat May 11 21:58:50 2013 IMPORTANT: OpenVPN's default port number is now 1194, based on an offici
al port number assignment by IANA.  OpenVPN 2.0-beta16 and earlier used 5000 as the default port.
Sat May 11 21:58:50 2013 NOTE: OpenVPN 2.1 requires '--script-security 2' or higher to call user-
defined scripts or executables
Sat May 11 21:58:50 2013 ******* WARNING *******: all encryption and authentication features disa
bled -- all data will be tunnelled as cleartext
Sat May 11 21:58:51 2013 TUN/TAP device tun0 opened
Sat May 11 21:58:51 2013 /sbin/ip link set dev tun0 up mtu 1500
Sat May 11 21:58:51 2013 /sbin/ip addr add dev tun0 local 10.0.0.2 peer 10.0.0.1
Sat May 11 21:58:51 2013 UDPv4 link local (bound): [undef]:1194
Sat May 11 21:58:51 2013 UDPv4 link remote: 192.168.0.103:1194
Sat May 11 21:59:01 2013 Peer Connection Initiated with 192.168.0.103:1194
Sat May 11 21:59:02 2013 Initialization Sequence Completed
```

5- Checking the connectivity:

Now, open some new terminals, and try *pinging* your new virtual IP addresses:

```
root@Open_Vpn_server :~#  ping 10.0.0.2

root@Remote_Pc:~# ping 10.0.0.1
```

You may also specify which interface for *ping* to use:

```
root@Open_Vpn_server :~#  ping –I tun0 10.0.0.2

root@Remote_Pc:~# ping –I tun0 10.0.0.1
```

Hit Ctrl-C to shut down OpenVPN and close the tunnels.

Workshop 4 – Part 2: Testing Encryption with Static Keys:

Now you want to test using encryption keys with OpenVPN.

1- Create the shared static key on the Open- VPN server:

```
root@Open_Vpn_server :~#  openvpn --genkey --secret static.key
```

2- Copy the key to the client PC:

```
root@Open_Vpn_server :~#  scp static.key 192.168.0.105:/etc/openvpn/keys/
```

3- Create the server configuration file:

Call it /etc/openvpn/server1.conf; you can call it anything you like. Use IP addresses that are on a different subnet than your server. Open_VPN_Server is at 192.168.3.10, so let’s make it’s tunnel endpoint address 10.0.0.1:

```
## openvpn server1.conf
dev tun
ifconfig 10.0.0.1 10.0.0.2
secret /etc/openvpn/keys/static.key
local 192.168.0.103
```

4- Create the client configuration file:

Call it /etc/openvpn/client1.conf.

```
## openvpn client1.conf
remote 192.168.0.103
dev tun
ifconfig 10.0.0.2 10.0.0.1
secret /etc/openvpn/keys/static.key
```

5 – Start the connection:

```
root@Open_Vpn_server :~#  openvpn /etc/openvpn/server1.conf
root@Remote_Pc:~# openvpn /etc/openvpn/client1.conf
```

6- Test the connectivity:

Just like in the previous recipe, you’ll see Initialization Sequence completed when the tunnel is completed, and both machines can *ping* each other:

```
root@Open_Vpn_server :~#  ping 10.0.0.2
root@Remote_Pc:~# ping 10.0.0.1
```

Hit Ctrl-C on both tunnel endpoints to shut it down.

7- Discussion:

Watch your messages when you establish the tunnels. When you set up the unencrypted tunnel, the warning:

```
******* WARNING *******: all encryption and authentication features disabled -- all
data will be tunnelled as cleartext
```

was displayed. That should be gone now.