

OpenVPN

PLUG North – Written/Presented by Michael Bevilacqua

What is a Virtual Private Network?

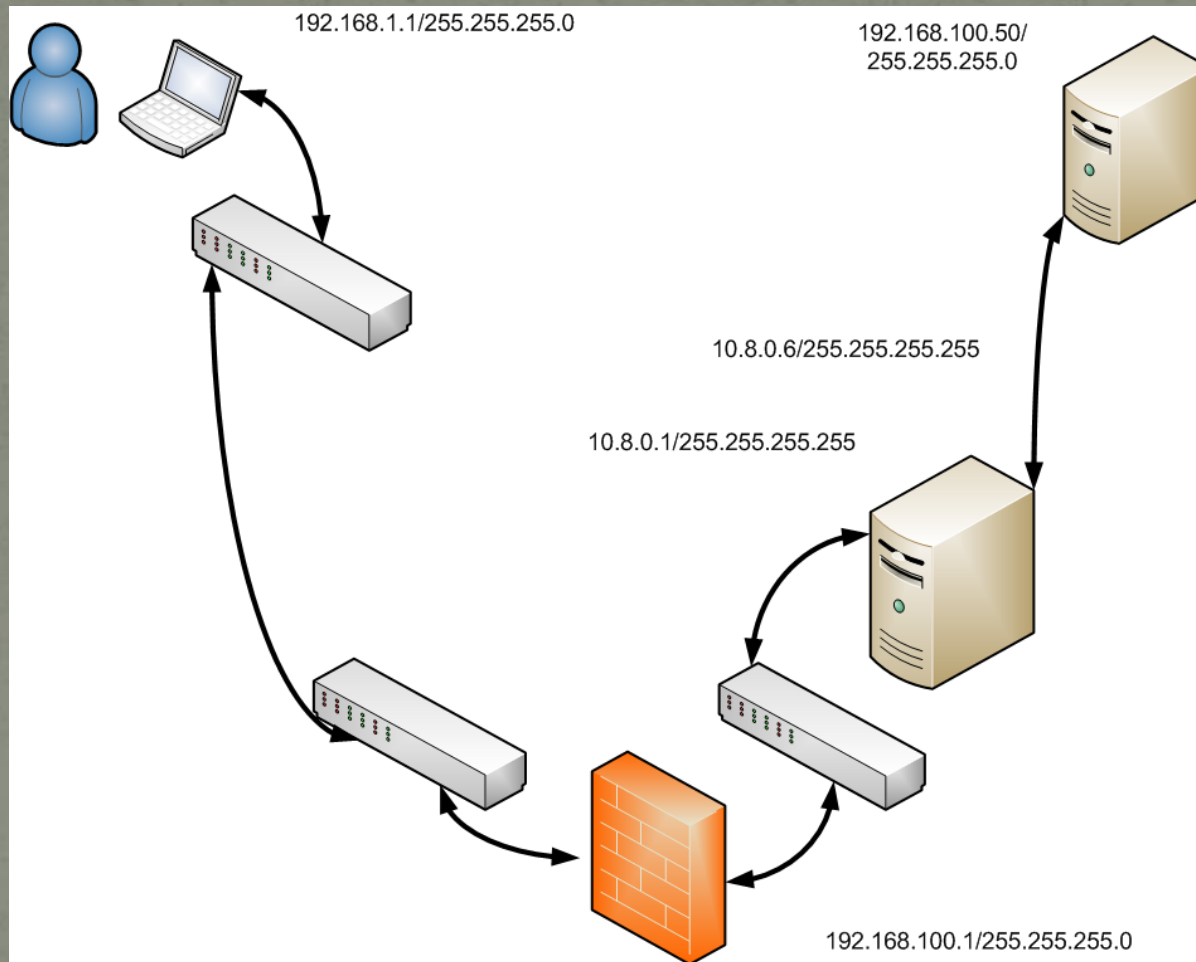
A *virtual private network* (VPN) is a computer network that is implemented in an additional software layer (overlay) on top of an existing larger network for the purpose of creating a private scope of computer communications or providing a secure extension of a private network into an insecure network such as the Internet.

Source: http://en.wikipedia.org/wiki/Virtual_private_network

The Uses of a VPN

- 1) Connecting mobile users securely to a private network
- 2) Masquerading the user's origin IP
- 3) Connecting together two or more private networks
- 4) Segregating mobile users from specific network resources (advanced)

The Client/Server Scenario



VPN Client/Server Setup Caveats

- 1) Requires that compatible software be installed on the client machine (Windows, Mac OS, Linux clients)
- 2) Server and client must be able to handle the encryption/decryption resource load
- 3) Destination must have enough bandwidth to handle client load
- 4) Origin and Destination subnets must be different unless you are bridging (advanced)

OpenVPN Server Setup (Debian)

```
apt-get install openvpn
```

Now build the server and client keys:

```
cd /usr/share/doc/openvpn/examples/easy-rsa/
```

```
cp -av 2.0 2.0.1
```

```
cd 2.0.1/
```

```
vim vars (edit the final SSL section)
```

Server and Client Key Build (con't)

```
./clean-all
```

```
./build-ca
```

```
./build-key-server server
```

```
./build-key client1
```

```
./build-dh
```

```
cd keys/
```

```
cp ca.key ca.crt dh1024.pem server.key server.crt /etc/openvpn
```

```
cd ..
```

```
cp -av keys ~/openvpn.keys
```

Write the /etc/openvpn/server.conf

```
local 192.168.100.6 # The Host's IP
port 1194 # The Host's Port
proto tcp # VPN Protocol
dev tun0 # Linux Device Driver
ca ca.crt # SSL Certs
cert server.crt
key server.key
dh dh1024.pem

server 10.8.0.0 255.255.255.0 # The VPN Subnet
ifconfig-pool-persist ipp.txt # client to virtIP map
push "route 192.168.100.0 255.255.255.0" # "PUSH" the route
push "redirect-gateway defl" # Route clients "through" the VPN
push "dhcp-option DNS 192.168.100.1" # Give them Internal DNS access
push "dhcp-option DNS 192.168.100.3"
push "dhcp-option DNS 192.168.100.7"
```

/etc/openvpn/server.conf (con't)

```
client-to-client # Allow client to client
keepalive 10 120 # Ping 10 / Timeout 120
comp-lzo # Compress using LZO
user nobody # Reduce Linux privs
group nogroup
persist-key # Needed for the above
persist-tun
status openvpn-status.log # Logging
log openvpn.log
verb 3
mute 20
management localhost 7505 # Management Console
```

Write the Masquerading (NAT) rule

In /etc/rc.local

```
iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o etho -j MASQUERADE  
exit 0
```

In /etc/sysctl.conf

```
net.ipv4.ip_forward=1
```

Reboot your server

```
init 6
```

The `ifconfig`

```
etho    Link encap:Ethernet HWaddr 00:06:5b:5c:fd:d4
        inet addr:192.168.100.6 Bcast:192.168.0.255 Mask:255.255.255.0

lo      Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0

tuno    Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
        inet addr:10.8.0.1 P-t-P:10.8.0.2 Mask:255.255.255.255
        UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500
        Metric:1
        RX packets:78 errors:0 dropped:0 overruns:0 frame:0
        TX packets:75 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:100
        RX bytes:6114 (6.1 KB) TX bytes:8043 (8.0 KB)
```

The `route -n`

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
10.8.0.2	0.0.0.0	255.255.255.255	UH	0	0	0	tuno
10.8.0.0	10.8.0.2	255.255.255.0	UG	0	0	0	tuno
192.168.100.0	0.0.0.0	255.255.255.0	U	0	0	0	etho
0.0.0.0	192.168.100.1	0.0.0.0	UG	100	0	0	etho

The `iptables -t nat -L`

Chain PREROUTING (policy ACCEPT)

target prot opt source destination

Chain POSTROUTING (policy ACCEPT)

target prot opt source destination

MASQUERADE all -- 10.8.0.0/24 anywhere

Chain OUTPUT (policy ACCEPT)

target prot opt source destination

Windows Vista Client Setup

- 1) You must disable User Account Control (UAC)
 - 1) Control Panel -> User Accounts -> Turn User Account Control On or Off
- 2) Install `openvpn-2.0.9-gui-1.0.3-install.exe`
- 3) From the `/root/openvpn.keys` directory, copy into `C:\Program Files\OpenVPN\config\`
 - 1) `ca.crt client1.crt client1.key`

Write the client.ovpn file

```
client # Designate client
dev-node mytap # Specify the Windows Device Driver Name
proto tcp # The VPN Protocol
dev tun # The Device Driver
remote office.digitalwave.com 1194 # The Remote Host and Port
persist-key # Keep the key open
persist-tun # Keep the dev open
ca ca.crt # Our Client SSL Certs
cert client1.crt
key client1.key
cipher aes-128-cbc
comp-lzo # LZO Compression
verb 3 # Logging options
mute 20
route-method exe # Windows specific routing method (route.exe)
route-delay 2 # Delay establishing route by 2 seconds
```

Windows Vista Client Setup (con't)

Change the name of the Network Interface to 'mytap':

- 1) Start -> Control Panel -> Network and Sharing Center -> Manage Network Connections
- 2) Device Name = TAP-Win32 Adapter V8
- 3) Change Name (not Device Name) to "mytap"
- 4) Reboot
- 5) Run OpenVPN from your System Tray Icon

Mac OS X Client Setup

- 1) Install Tunnelblick_3.ob14.dmg
- 2) Go to Applications and run Tunnelblick
- 3) Copy into ~/Library/openvpn the ca.crt, client1.crt and client1.key
- 4) Write the client1.ovpn file (next page)
- 5) Run Tunnelblick by clicking on the Tunnel Icon on the top right status bar. Click client1 to connect to the client1 VPN config.

Write the client1.ovpn file

```
client # Designate client
proto tcp # The VPN Protocol
dev tun # The Device Driver
remote office.digitalwave.com 1194 # The Remote Host and Port
persist-key # Keep the key open
persist-tun # Keep the dev open
ca ca.crt # Our Client SSL Certs
cert client1.crt
key client1.key
cipher aes-128-cbc
comp-lzo # LZO Compression
verb 3 # Logging options
mute 20
route-delay 2 # Delay establishing the route by 2 seconds
```

Ubuntu Linux OpenVPN Client

- 1) `apt-get install openvpn`
- 2) Copy into `/etc/openvpn/` the `ca.crt`, `client1.crt` and `client1.key`
- 3) Write your `client1.conf` (next page)
- 4) Manually start and stop OpenVPN's daemon:

```
/etc/init.d/openvpn start
```

```
/etc/init.d/openvpn stop
```

Write the client1.conf file

```
client # Designate client
proto tcp # The VPN Protocol
dev tun # The Device Driver
remote office.digitalwave.com 1194 # The Remote Host and Port
persist-key # Keep the key open
persist-tun # Keep the dev open
ca ca.crt # Our Client SSL Certs
cert client1.crt
key client1.key
cipher aes-128-cbc
comp-lzo # LZO Compression
verb 3 # Logging options
mute 20
route-delay 2 # Delay establishing the route by 2 seconds
```

Using the Management Console

- 1) ssh to the server running openvpn
- 2) apt-get install telnet
- 3) telnet localhost 7505
- 4) Type 'help' to see a list of available commands

Exercise #1:

Debug a live connection using the 'log on' command

References

- <http://www.openvpn.net/>
- <http://en.wikipedia.org/wiki/OpenVPN>
- http://en.wikipedia.org/wiki/Virtual_private_network
- http://en.wikipedia.org/wiki/Point-to-point_tunneling_protocol

Meeting Notes

Q) Why use TCP instead of the default UDP?

A) TCP has error checking and is able to traverse routers with greater ease. UDP is prone to timeouts.

Q) Can the 192.168.100.0 subnet interface with the 10.8.0.0 subnet?

A) No. There is no route to this subnet in this example.

Q) What about large scale key management?

A) A key server with a very regular rotation would be your best solution in this situation.