Aalto University

School of Electrical Engineering
Department of Communications and Networking
ELEC-E7330 - Laboratory Course in Internet Technologies

Lab 28

Virtual Private Networks

Student's instructions
Heikki Salokanto
15 September 2008

Updated by Juho Kaivosoja
10 June 2009
7 January 2011
25 May 2011

Updated by Tahir Jemal
29 August 2018

Preliminary report
Lab work
Final Report

Preliminary Report

Grading

Your final lab grade consists of the preliminary report score (max 15 pts) and final report score (max 30 pts). You will have to get at least 50 % of each to pass.
Preliminary report gives you two points per question except for PR5, in which each sub-question is worth one point. Please answer briefly and accurately for maximum points.

PR1.

What is a virtual private network and what is it used for? How does it differ from a VLAN?

PR2.

Briefly explain features and differences of VPN "technologies" PPTP, IPSec and OpenVPN.

PR3.

OpenVPN can be deployed in "routing" or "bridging" mode, and it also supports both UDP and TCP based data transmission. Figure out in what circumstances should you use these different methods.

PR4.

Explain how TLS works with OpenVPN.

PR5.

Find out the Linux commands for the following:

1. Bring up interface eth1 with ip 192.168.1.1 and netmask 255.255.255.0
2. Modify routing table so that requests to 10.38.40.0 (netmask 255.255.255.0) are routed to 10.38.0.10
3. Delete the previous entry from the routing table
4. Establish an unsecured OpenVPN connection between you (10.1.2.3) and a remote host (10.1.2.4), assigning addresses 10.4.5.6 and 10.4.5.7 for the computers on the new link. (Hint: you probably only need switches --remote, --dev and --ifconfig)
5. Using tcpdump, capture full UDP packets over eth0 on port 1194

PR6.

Your employer has blocked outward network access to all ports but TCP 80, 443 and 8080. However, you start feeling an irresistible urge to chat on IRC, so how would you go around the limitation at the company network?

Please note that there is no single correct answer for several of these questions.
If you feel uncertain about SSH, scp, Emacs (or nano or vi), Wireshark or Linux networking in general, you really should spend a few moments around these before entering the lab. You will surely need that knowledge later in your life as well, so now is a good time to study it.
Reading "Setting up a CA in OpenVPN" at
`http://openvpn.net/index.php/documentation/howto.html#pki` and
`https://help.ubuntu.com/lts/serverguide/openvpn.html.en`, are strongly recommended, so during the lab you can just go through a similar document quickly.

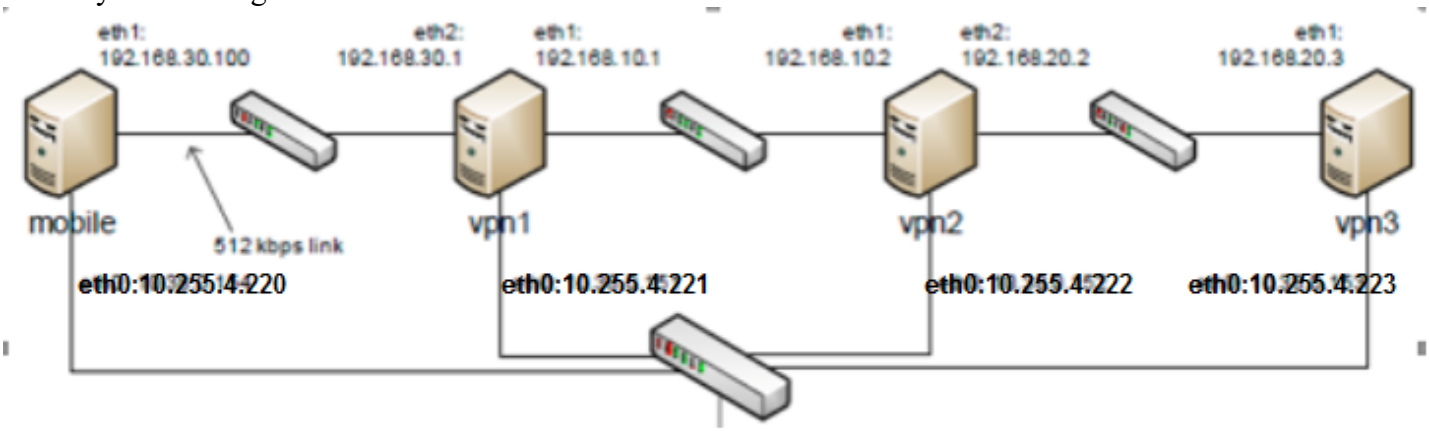
Try the following sources:

- OpenVPN documentation
- 'man route', 'man ifconfig', 'man tcpdump'
- if you Google for something like “linux networking”, the Net3-4-HOWTO will pop up among the first hits, but be aware it is quite badly outdated (last update in 1999) just like the Net-Concepts-HOWTO (updated in 2000). Those how-to’s discuss Linux 2.2 and quite a few things have changed since.

Lab work

Before the launch

The network setup has to be as follows (all netmasks /24).The vm are running inside vmware-esxi and they are running Ubuntu 14.04 LTS server.



VM running inside VMWARE-ESXI

Figure 1. VM running in VMWARE environment

Route configuration: see Table 3. It adheres to the following table 1 below:
Table 1.

	to mobile	to vpn1	to vpn2	to vpn3
from mobile	local	eth1	no	no
from vpn1	eth2	local	eth1	route via vpn2
from vpn2	route via vpn1	eth1	local	eth2
from vpn3	route via vpn2	route via vpn2	eth1	local

If the running configuration does not comply with the diagram, you must take necessary actions (ifconfig, route, etc.) to make it so. Your next tasks depend on this, so be careful.
Please do not install any additional software on the machines – at least not without asking the assistant first!
The Linux virtual machines can be reached using the management interface as shown in the table 2 below.

Table 2. SSH Access to the machines

vm machine	**Management IP address**	**Password**
Mobile	lab@10.255.4.220	eeeeee
vpn1	lab@10.255.4.221	eeeeee
vpn2	lab@10.255.4.222	eeeeee
vpn3	lab@10.255.4.223	eeeeee

Table 3 Routing tables.

Mobile

Destination	Gateway	Metric	Interface

192.168.30.0 / 24	local	0	eth1

VPN1

Destination	Gateway	Metric	Interface
192.168.20.0 / 24	192.168.10.2	0	eth1
192.168.10.0 / 24	local	0	eth1
192.168.30.0 / 24	local	0	eth2

VPN2

Destination	Gateway	Metric	Interface
192.168.20.0 / 24	local	0	eth2
192.168.10.0 / 24	local	0	eth1
10.10.10.0 / 24	192.168.10.1	0	eth1

VPN3

Destination	Gateway	Metric	Interface
192.168.20.0 / 24	local	0	eth1
192.168.10.0 / 24	192.168.20.2	0	eth1
10.10.10.0 / 24	192.168.20.2	0	eth1

All machines

Mighty have their routing tables (in addition to the above):

Destination	Gateway	Metric	Interface
10.38.0.0 / 24	local	0	eth0
0.0.0.0 / 0	10.38.0.254	100	eth0

As you can see from these tables, host Mobile does not have a route to VPN2 or VPN3. This is intentional, because a mobile user is not aware of a company's intranet servers until he opens a VPN connection to the intranet. In order to ease up the things a little bit, there is a pre-configured route in VPN2 and VPN3 for the VPN tunnel in ex. 3 (10.10.10 / 24).
The routing is automatically set up the way described above when the machines are booted and the interfaces brought up. The configuration resides in `/etc/network/interfaces` in each host, and essentially consists of the following (per interface): This is already preconfigured on each interfaces.

```
iface eth1 inet static
address 192.168.10.1
netmask 255.255.255.0
up route add -net 192.168.20.0/24 gw 192.168.10.2 dev eth1
down route del -net 192.168.20.0/24 gw 192.168.10.2 dev eth1
```

Host name resolution

Manual configuration only. Each machine has the following in /etc/hosts:

```
127.0.0.1               localhost
192.168.30.100          mobile
192.168.10.1            vpn1
```

```
192.168.10.2            vpn2
192.168.20.3            vpn3
```

Note that these are only for convenience when opening SSH connections and stuff - all VPN and routing related stuff should be bound to IP addresses instead of host names in order to attach them to correct interfaces.

ARP

```
arp -s 192.168.20.3 <VPN3's eth1 MAC address>
```

Enabling IP forwarding

Added the following command in `/etc/init.d/networking` on VPN1 and VPN2 (those two need to forward packets):

```
echo 1 >/proc/sys/net/ipv4/ip_forward
```

1. Unsecured P2P VPN

Tagline: Establish an unsecured OpenVPN link between hosts VPN1 and VPN3.

Your tasks

1. Start capturing packets going through VPN2.
   - Use tcpdump. By default, it only captures the first 68 bytes of a packet, so increase this number to 1500 so that you are going to get whole packets.
2. Open the web page served by VPN3 in a web browser on VPN1.
   - Use lynx. Be careful to open the web page so that your traffic moves through VPN2 and not through the maintenance network (10.38.0/24) - otherwise you are going to capture nothing.
3. Establish an unsecured point-to-point OpenVPN link between VPN1 and VPN3 through VPN2.
   - This may sound a little bit tricky, but you only need to issue one command with a couple of parameters:
     - Use the TUN (tunnel) device (--dev tun1).
     - Define the VPN endpoints (--ifconfig). Preferably use something not within any existing subnets, for example VPN1 = 10.0.0.1 and VPN3 = 10.0.0.2.
     - Use high enough verbosity level so that you see what's happening (for example --verb 5).
4. Open the web page served by VPN3 in a web browser on VPN1 through the VPN tunnel.
   - Remember to go through the VPN - in this case there is no firewall forcing you to do so
5. Stop the capture and save the results.
6. (For final report) Measure round-trip-time (in ms) and TCP throughput (in MiB/s) between VPN1 and VPN3 without any VPN tunnel.
   - Use `iperf` for throughput measurements
   - If you have plenty of time, you can also measure UDP throughput (iperf handles this as well), or try using OpenVPN over TCP
7. (For final report) Measure round-trip-time (in ms) and TCP throughput (in MiB/s) between VPN1 and VPN3 over your VPN.
8. Stop the VPN tunnels.

9. (For final report) Compare the captured results of 2. and 4. Make sure you both understand your findings. There is Wireshark installed on the host computer; you probably want to ensure right away that you captured the right thing.

2. Secured P2P VPN

Tagline: Establish a secured (pre-shared key) OpenVPN link between hosts VPN1 and VPN3.

Your tasks

1. Generate a pre-shared key and share it to the other machine.
   - `openvpn --genkey`
2. Capture traffic going through VPN2.
3. Establish a secured VPN tunnel between VPN1 and VPN3 using the PSK.
   - Use the same command you used last time, but add a switch for the encryption.
4. Open the web page served by VPN3 through the VPN.
5. Stop the capture.
6. (For final report) Measure round-trip-time (in ms) and TCP throughput (in MiB/s) between VPN1 and VPN3 over your encrypted VPN.
7. Shut down the VPN.
8. (For final report) Compare the captured results to the results when no security was applied. You probably want to try it out right away with Wireshark.

3. Mobile user to a company network

Tagline: A VPN gateway server with TLS security
Setup: MOBILE acts as a mobile user connecting to the company's VPN server running on VPN1. The mobile user wants to open an intranet page on web server VPN3, so packets coming from the VPN must be routed to the intranet (192.168.30.0 / 24). Authentication will be taken care of by creating and issuing a certificate to the mobile user. The mobile user must receive a dynamically assigned IP address for the tunnel.

Your tasks

NB! Do **NOT** modify any example files; copy them as instructed in the How-to document on the host computer.

1. Install a certificate authority (CA) on VPN1.
   - see "Setting up your own CA" in the OpenVPN HOWTO (`howto-pki.html`)on the host computer desktop (or at `http://openvpn.net/index.php/documentation/howto.html#pki`) and `https://help.ubuntu.com/lts/serverguide/openvpn.html.en`
     ```
     sudo cp -r /usr/share/easy-rsa /etc/openvpn
     ```
   - You don't necessarily need to edit the `vars` file - you can enter the parameters on-the-fly
   - One client key is enough
   - Issue and sign the certificates when asked
   - Transfer the files indicated by the table to MOBILE
2. Start capturing on VPN2.
3. Configure and start the OpenVPN server in VPN1. There must be a pool of IP addresses that can be assigned to connecting clients.
   - Unlike previous exercises, you'd better use a configuration file instead of typing all the parameters on the command line
   - Sample configuration files are found in `/usr/share/doc/openvpn/examples/sample-config-files` @ VPN1
   - Transfer the client configuration file to MOBILE
   - Ensure that you are using the TAP device both at the server and at the client
   - Invoke `openvpn` with the `--config` switch to use a specified configuration file
4. Connect the mobile user to the VPN server
   - Try to ping the VPN first
   - If the VPN is ok, you need to add a route entry for "the intranet" (192.168.20 / 24).
5. Open a web page served by VPN3 through the VPN
6. Stop the capture.
7. (For final report) Measure round-trip-time and TCP throughput between MOBILE and VPN1 without a VPN
8. (For final report) Measure round-trip-time and TCP throughput between MOBILE and VPN1 over the secured VPN
9. Shut down the VPN.
10. Show your OpenVPN configuration files and the working setup to the assistant.

Final Report

Discuss the following questions in your final report. Please keep your answers brief and stick to the matter – the points are awarded for reasoned observations and factual comments – but do cover all aspects of the questions. All citations must cite their sources.

FR1. (6 points)

Compare the throughput and round-trip-time values for different use cases (measured in 1.6, 1.7 and 2.6). Make **a)** a table and **b)** a graph of both throughput and RTT for easy comparison and **c)** explain the differences (what causes it, how big an effect there is) in throughput and RTT, between the cases.

FR2. (6 points)

Compare the mobile network results (measured in 3.7 and 3.8) like in question FR1, items **a** through **c**.

FR3. (6 points)

a) What differences are there between the mobile and fixed-network cases?
b) How and why are the effects of the VPN different in fixed and mobile networks?
c) Estimate how all your measurements would change if made in a physical network instead of a virtual setup.

FR4. (6 points)

a) What security threats are there when an employer connects to a company network from outside?
b) VPN reduces a lot of the risks, but what risks are left?
c) How would you go about making the network and its usage even safer?

FR5. (6 points)

Find real life use cases for the three different VPN setups and describe how (if) you would need to modify them to make them useful. The cases are **a)** an unsecured P2P VPN connection, **b)** secured P2P VPN connection and **c)** a client-server secured VPN setup ("mobile user to a company network").