SWINBURNE UNIVERSITY OF TECHNOLOGY

TNE20003 – Internet and Cybersecurity for Engineering Applications

Portfolio Task – Lab 9 Pass Task

Aims:

- To observe and investigate tunnelling in a network.

Preparation:

- View "Introduction to Cybersecurity" & "Cybersecurity"

Due Date:

- All tasks in this lab are to be completed and demonstrated to your Lab instructor preferably during or at the end of the current lab, but if you do not complete the tasks you may demonstrate it at the beginning of your next lab class.

Task 1

Get an understanding of the lab.

The purpose of this lab is to investigate tunnelling, in particular to consider how this and similar software can be used to subvert a security program. We will be examining tunnelling of telnet but the general principles can be applied to other protocols.

Telnet is used because it is a simple protocol with messages transmitted without encryption ("in clear text"). Of course, this also makes it a very insecure protocol and you should never use telnet when there are alternatives. Do not under any circumstances interpret the use of telnet in this lab as a recommendation for it to be used anywhere else.

You will use the Wireshark packet sniffing system and open source http tunnel software. You will use Wireshark to examine the packets.

This work is to be carried out in the VMWARE environment. We will be running two Linux Ubuntu virtual machines.

You may be asked for a password. All passwords are **user**

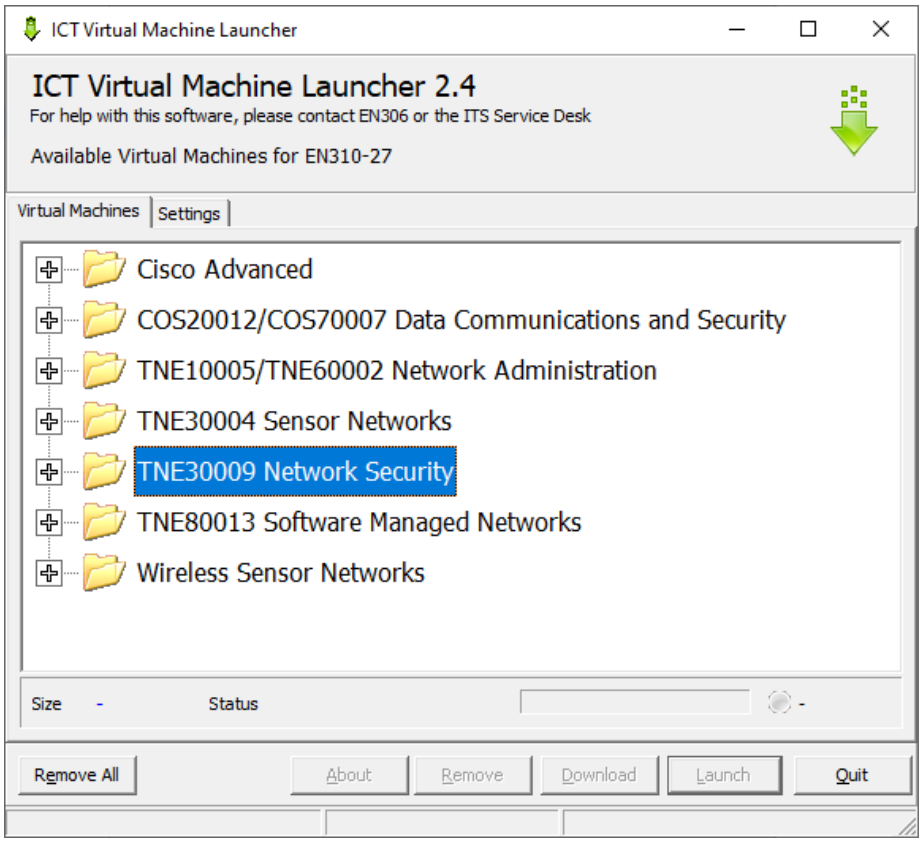
This lab uses the GNU http tunnelling software. It provides two tunnelling programs: **hts** and **htc**. **hts** is installed on the server system to accept tunnels from a specified port number and transfer them to another port number. **htc** is used to communicate to **hts** to set up a tunnel.

Assessment of this lab is through you demonstrating the working tunnel and wireshark output to the lab demonstrator and answering some questions (listed at the end of the lab) that the demonstrator will ask you.

Task 2

This lab uses Linux Virtual Machines (VMs) running in via VMWARE. If you are unfamiliar with Linux please spend some time before the lab learning some of its basic commands.
The VMs are both available via the start menu. To get them started do the following:

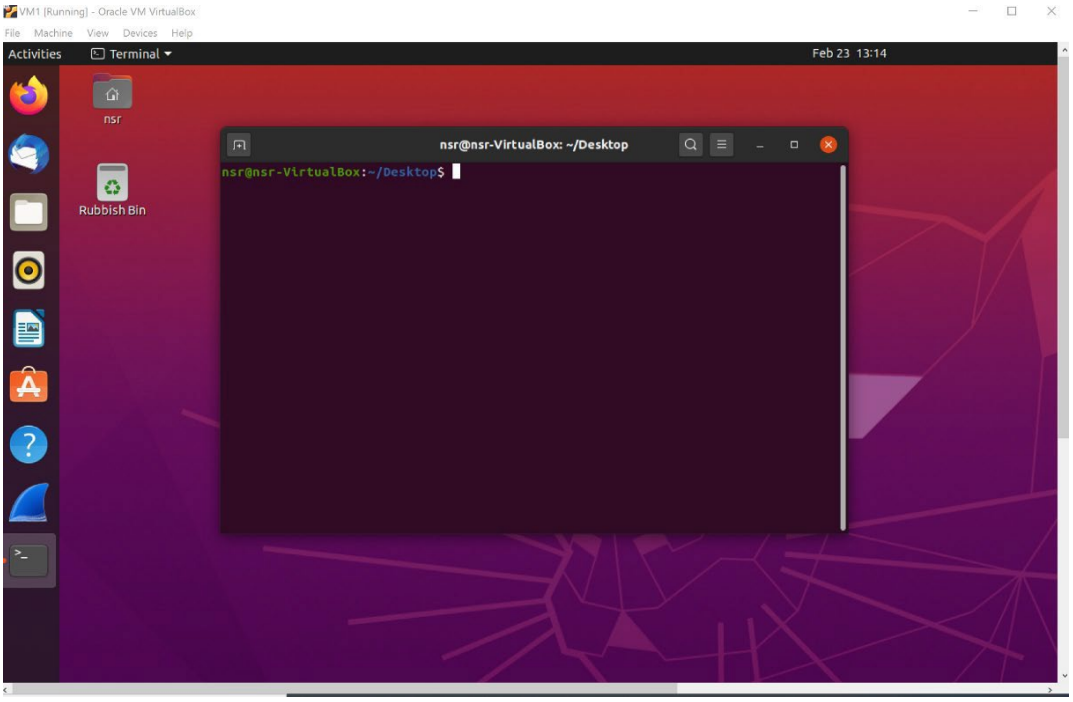
1. Go to Start menu > Virtual Machine Launcher. You will be presented with the following window:



Choose the TNE30009 Network Security tab. You will be presented with three VMs. Download Ubuntu 1 and Ubuntu 2. This may take a few minutes.

Launch the VMs. You may be asked to update the VMWARE software or the Ubuntu version. Ignore these.

2. You should now have two Ubuntu VMs running.



3. You will need to work within the terminal. Right click your mouse and choose "**Open in Terminal**". You can determine your IP address with the command **ifconfig**. The interface you will be monitoring is **ens33**. It should have an IP address assigned from 192.168.0.0/16 with both hosts on the same /24 subnet.
4. Once both VMs are running determine their IP addresses and confirm they are on the same subnet. Use

```
Ifconfig
```

5. You should now be able to ping between the two VMs. Make sure you can.

```
ping <ip address>
```

Task 3

To set up the tunnel, choose one machine to be the tunnel server and the other to be the tunnel client. You will telnet via the tunnel from the tunnel client to the tunnel server. You are to determine the addresses and port numbers to use in the command below.

On the server machine use the following command to accept tunnelled messages from port 80. Messages are to be forwarded to the telnet server listening on port 23.

```
sudo hts -F <IP address to forward to>:<forwarding port number>
<receiving port number>
```

On the client machine use the following command to initiate a tunnel to the server machine over port 80.

Use 2323 as the receiving port number.

```
sudo htc -F <receiving port number> <IP address to forward
to>:<forwarding port number>
```

Task 4

To monitor the traffic start Wireshark. Monitor the ethernet port which will usually be **ens3.**

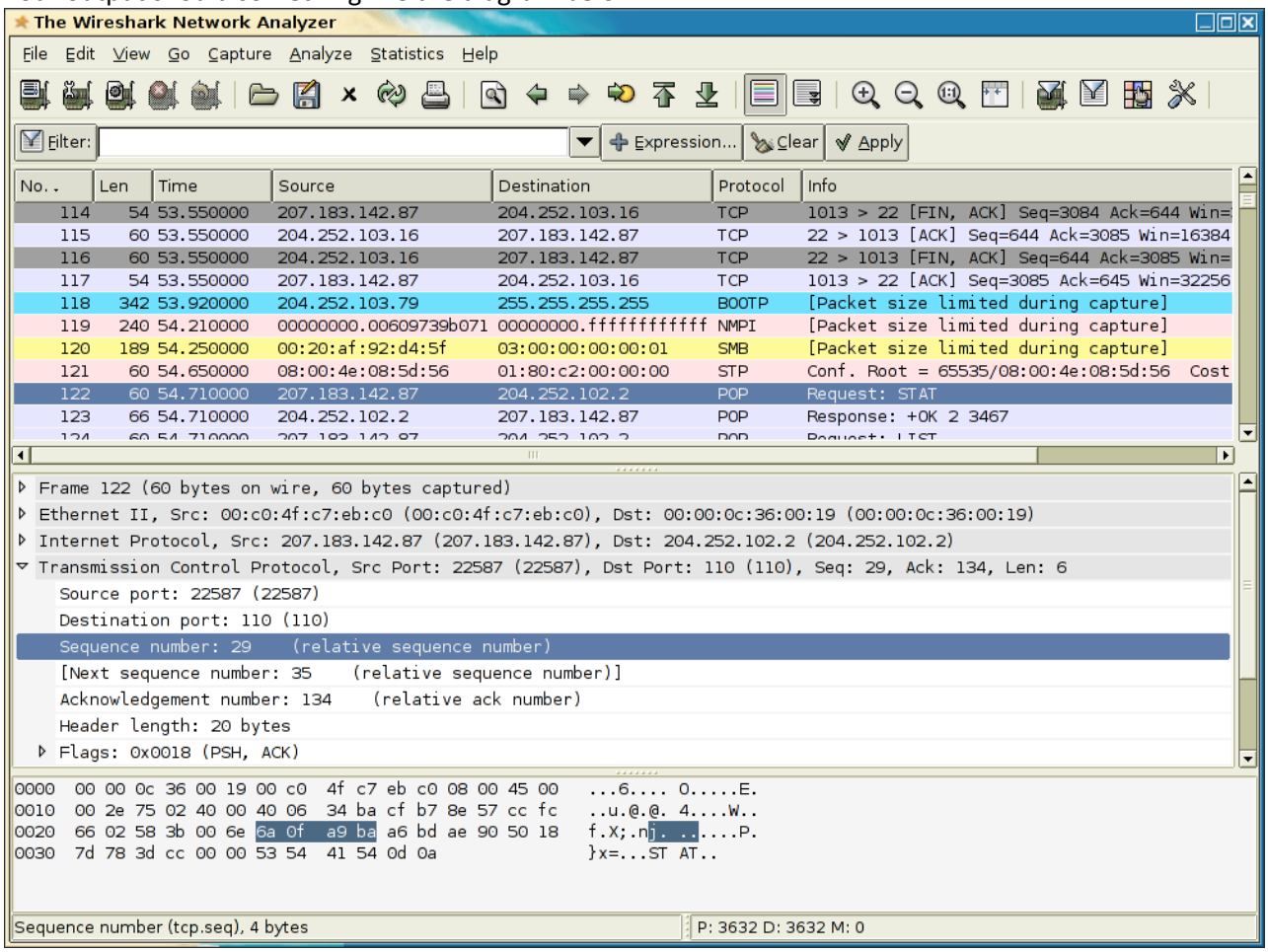
Initiate a telnet session via the tunnel from the client. The format of the telnet command is:

```
telnet <IP address of tunnel client> <receiving port number>
```

You should receive a telnet login request. Login with user **nsr** and password **user**.

Using Wireshark you should be able to see the transmitted traffic. Note how the telnet traffic is tunnelled.

Your output should something like the diagram below:



Task 5

Assessment of this lab

This lab will be assessed with a discussion between you and the lab supervisor. You will need to demonstrate that you have successfully completed the lab. He or she will also ask you the following:

1. What command is needed on the tunnel server to accept tunnelled messages on port 80 and send them to port 23 on the local host? Include the IP addresses that you used.
2. What command is needed on the tunnel client to tunnel messages to and from port 2323 to and from port 80?
3. What did Wireshark interpret the tunnelled traffic as?
4. How did the tunnelling software attempt to hide the traffic?
5. How might tunnelling of this kind be used to subvert security policy?

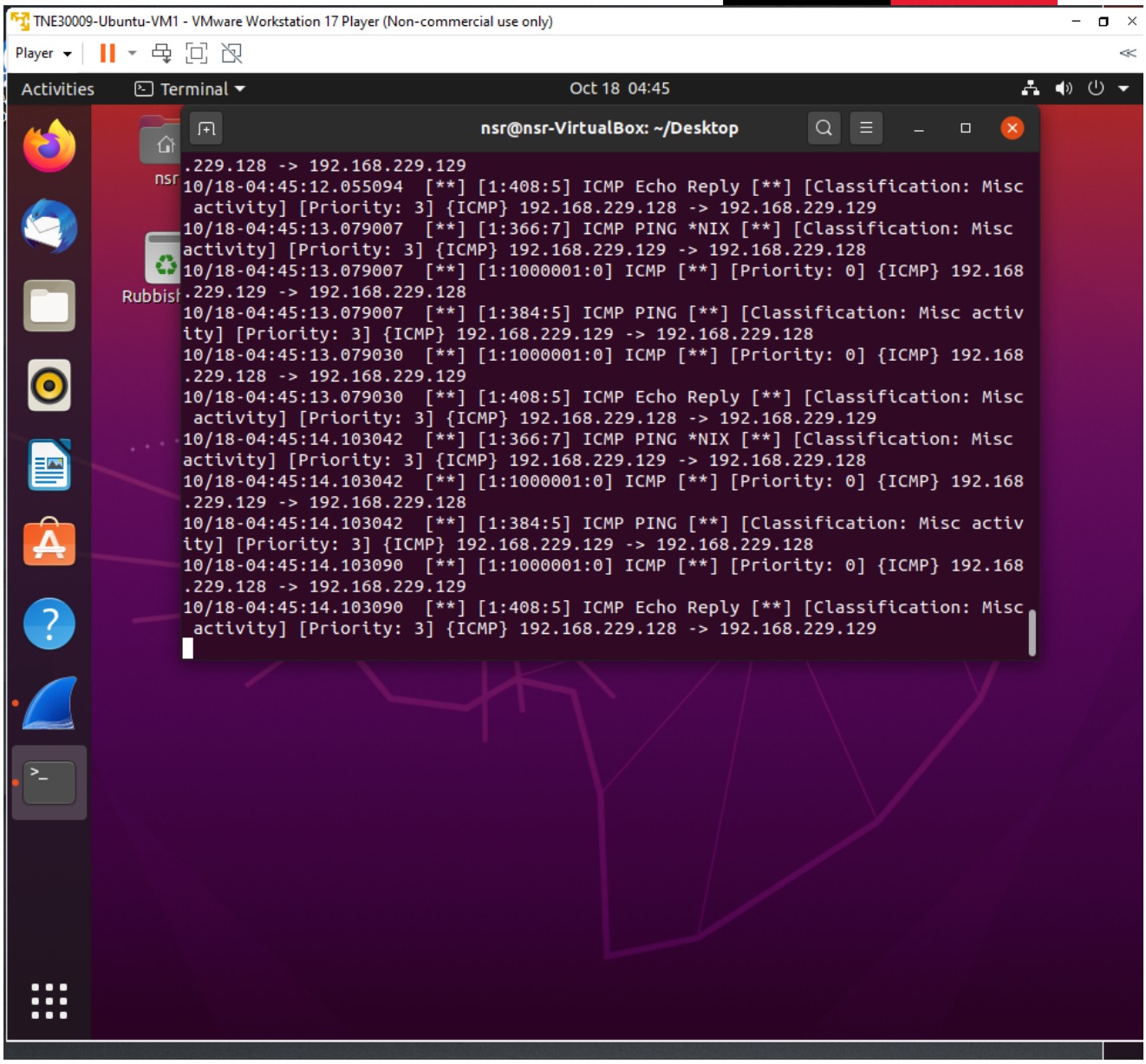
1. sudo hts –F 192.168.2.0:23 80

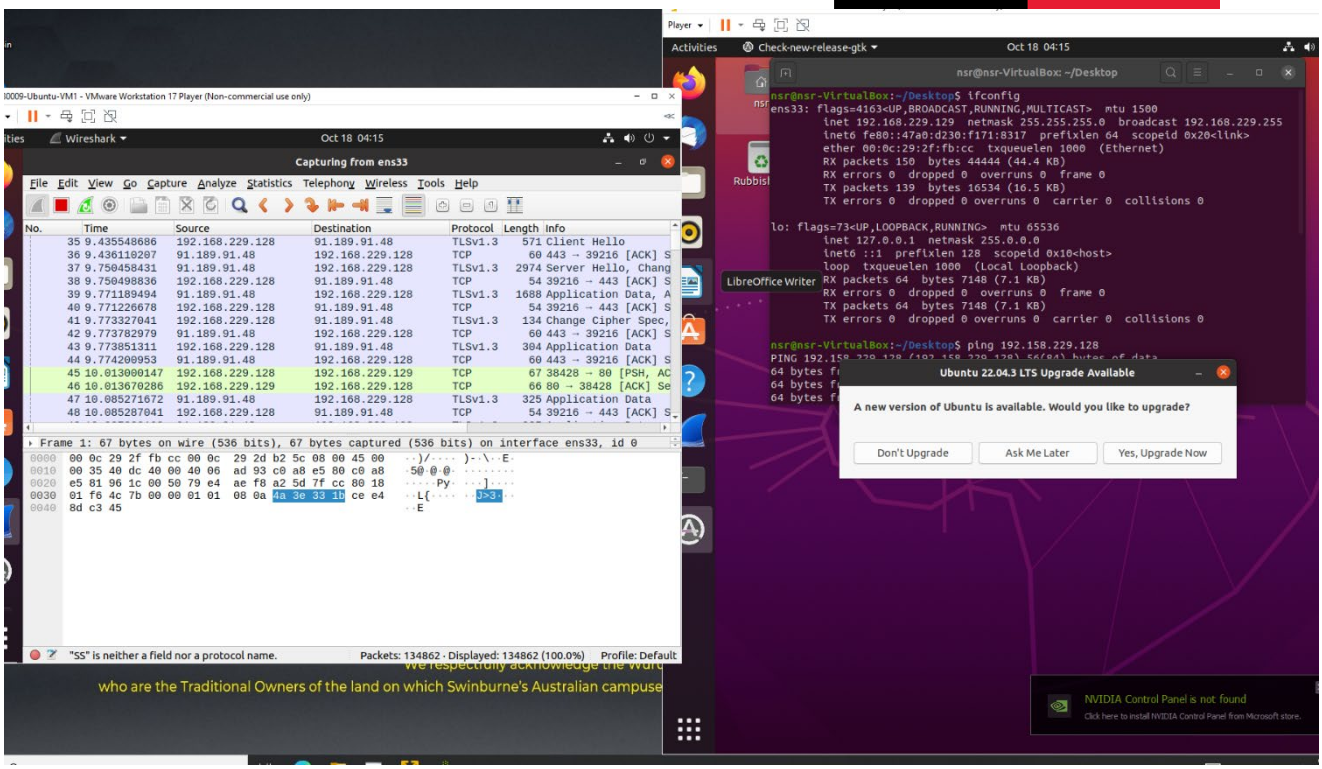
2. sudo htc –F 23 192.168.1.0:80

3. Wireshark is capable of interpreting it based on the protocols it supports. Wireshark can interpret tunnelled traffic by analyzing the encapsulating and encapsulated protocols separately.

4. encapsulating it within another protocol

5. Bypassing Firewall Rules, Evading Content Filtering,


TNE30009-Ubuntu-VM1 - VMware Workstation 17 Player (Non-commercial use only)
Player
Activities
Terminal
Oct 18 04:45
nsr@nsr-VirtualBox: ~/Desktop
.229.128 -> 192.168.229.129
10/18-04:45:12.055094 [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.128 -> 192.168.229.129
10/18-04:45:13.079007 [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.129 -> 192.168.229.128
10/18-04:45:13.079007 [**] [1:1000001:0] ICMP [**] [Priority: 0] {ICMP} 192.168.229.129 -> 192.168.229.128
10/18-04:45:13.079007 [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.129 -> 192.168.229.128
10/18-04:45:13.079030 [**] [1:1000001:0] ICMP [**] [Priority: 0] {ICMP} 192.168.229.128 -> 192.168.229.129
10/18-04:45:13.079030 [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.128 -> 192.168.229.129
10/18-04:45:14.103042 [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.129 -> 192.168.229.128
10/18-04:45:14.103042 [**] [1:1000001:0] ICMP [**] [Priority: 0] {ICMP} 192.168.229.129 -> 192.168.229.128
10/18-04:45:14.103042 [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.129 -> 192.168.229.128
10/18-04:45:14.103090 [**] [1:1000001:0] ICMP [**] [Priority: 0] {ICMP} 192.168.229.128 -> 192.168.229.129
10/18-04:45:14.103090 [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.229.128 -> 192.168.229.129


Capturing from ens33
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
No. Time Source Destination Protocol Length Info
35 9.435548686 192.168.229.128 91.189.91.48 TLSv1.3 571 Client Hello
36 9.436110207 91.189.91.48 192.168.229.128 TCP 60 443 → 39216 [ACK] S
37 9.750458431 91.189.91.48 192.168.229.128 TLSv1.3 2974 Server Hello, Chang
38 9.750498836 192.168.229.128 91.189.91.48 TCP 54 39216 → 443 [ACK] S
39 9.771189494 91.189.91.48 192.168.229.128 TLSv1.3 1688 Application Data, A
40 9.771226678 192.168.229.128 91.189.91.48 TCP 54 39216 → 443 [ACK] S
41 9.773327041 192.168.229.128 91.189.91.48 TLSv1.3 134 Change Cipher Spec,
42 9.773782979 91.189.91.48 192.168.229.128 TCP 60 443 → 39216 [ACK] S
43 9.773851311 192.168.229.128 91.189.91.48 TLSv1.3 304 Application Data
44 9.774200953 91.189.91.48 192.168.229.128 TCP 60 443 → 39216 [ACK] S
45 10.013000147 192.168.229.128 192.168.229.129 TCP 67 38428 → 80 [PSH, AC
46 10.013670286 192.168.229.129 192.168.229.128 TCP 66 80 → 38428 [ACK] Se
47 10.085271672 91.189.91.48 192.168.229.128 TLSv1.3 325 Application Data
48 10.085287041 192.168.229.128 91.189.91.48 TCP 54 39216 → 443 [ACK] S
Frame 1: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface ens33, id 0
"SS" is neither a field nor a protocol name.
Packets: 134862 · Displayed: 134862 (100.0%)
Profile: Default
who are the Traditional Owners of the land on which Swinburne's Australian campuse
nsr@nsr-VirtualBox:~/Desktop$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.229.129 netmask 255.255.255.0 broadcast 192.168.229.255
inet6 fe80::47a0:d230:f171:8317 prefixlen 64 scopeid 0x20<link>
ether 00:0c:29:2f:fb:cc txqueuelen 1000 (Ethernet)
RX packets 150 bytes 44444 (44.4 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 139 bytes 16534 (16.5 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 64 bytes 7148 (7.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 64 bytes 7148 (7.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
nsr@nsr-VirtualBox:~/Desktop$ ping 192.168.229.128
Ubuntu 22.04.3 LTS Upgrade Available
A new version of Ubuntu is available. Would you like to upgrade?
Don't Upgrade
Ask Me Later
Yes, Upgrade Now
NVIDIA Control Panel is not found