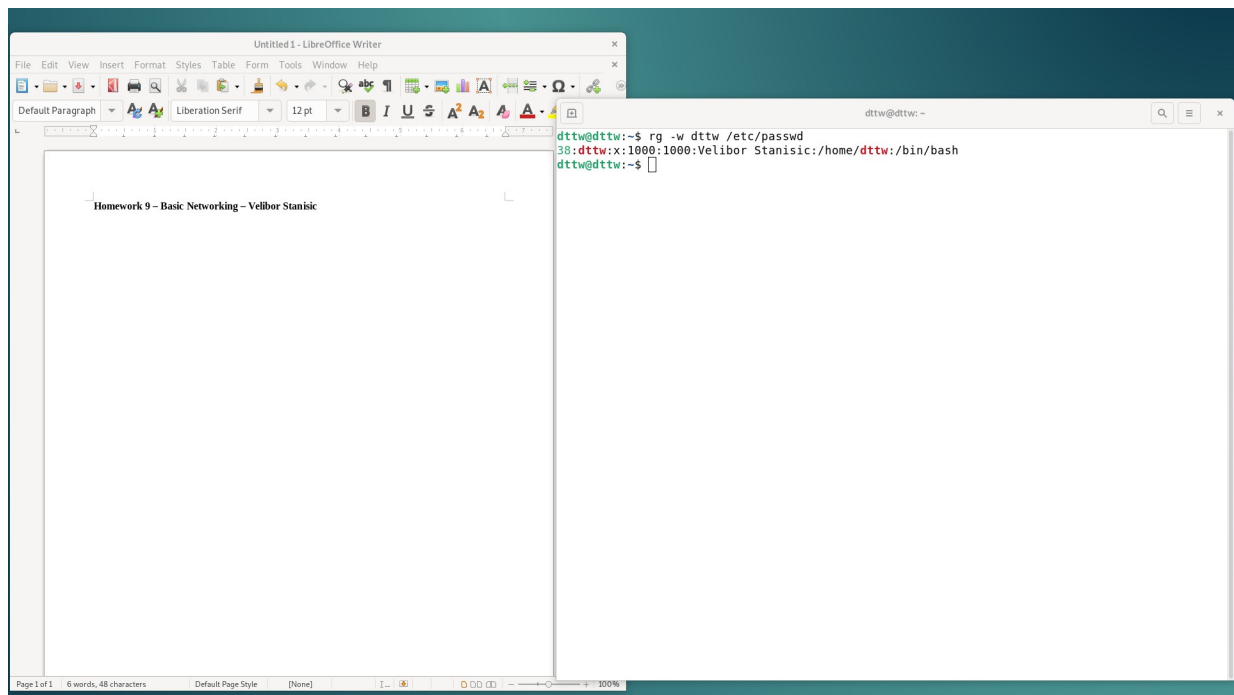


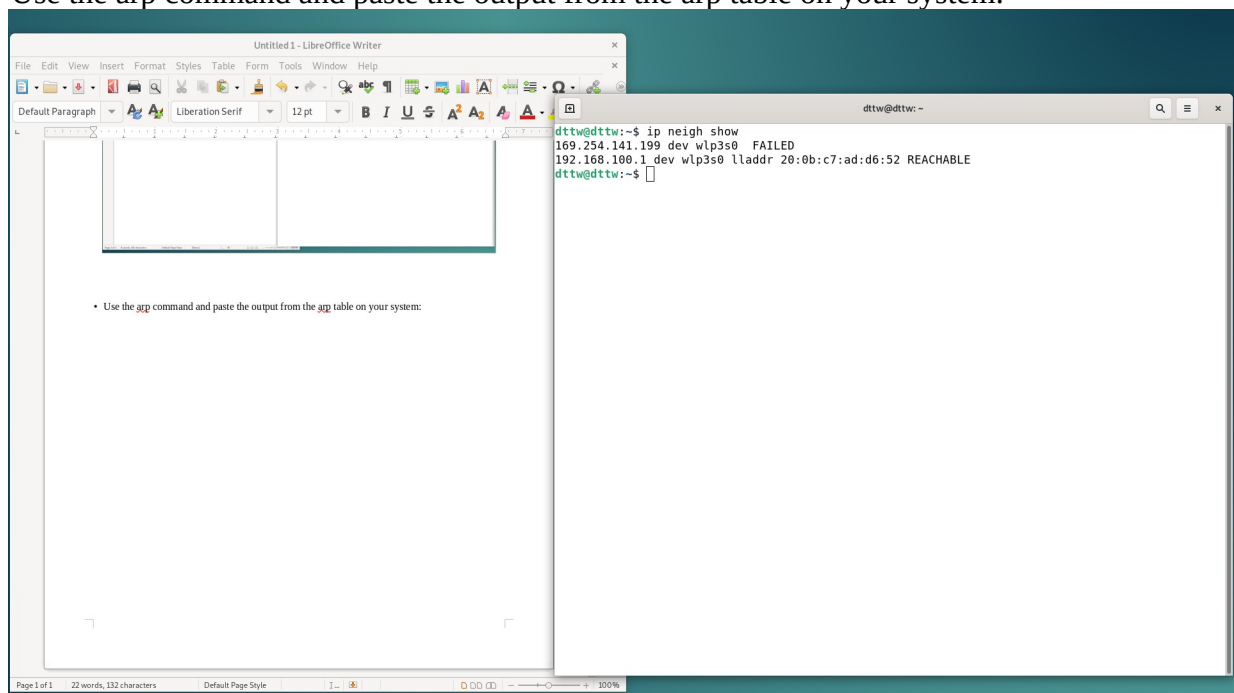
Homework 9 – Basic Networking – Velibor Stanisic



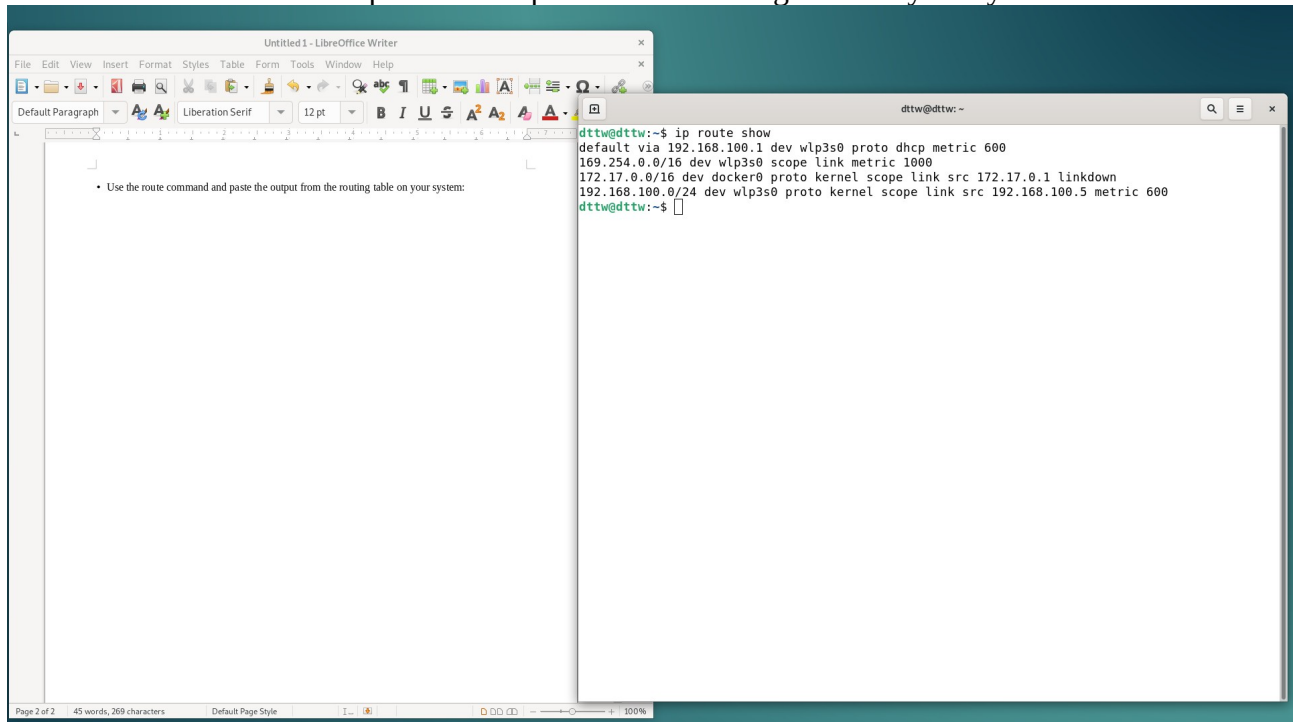
Exercise 1 – Basic network stuff

Difficulty: Easy

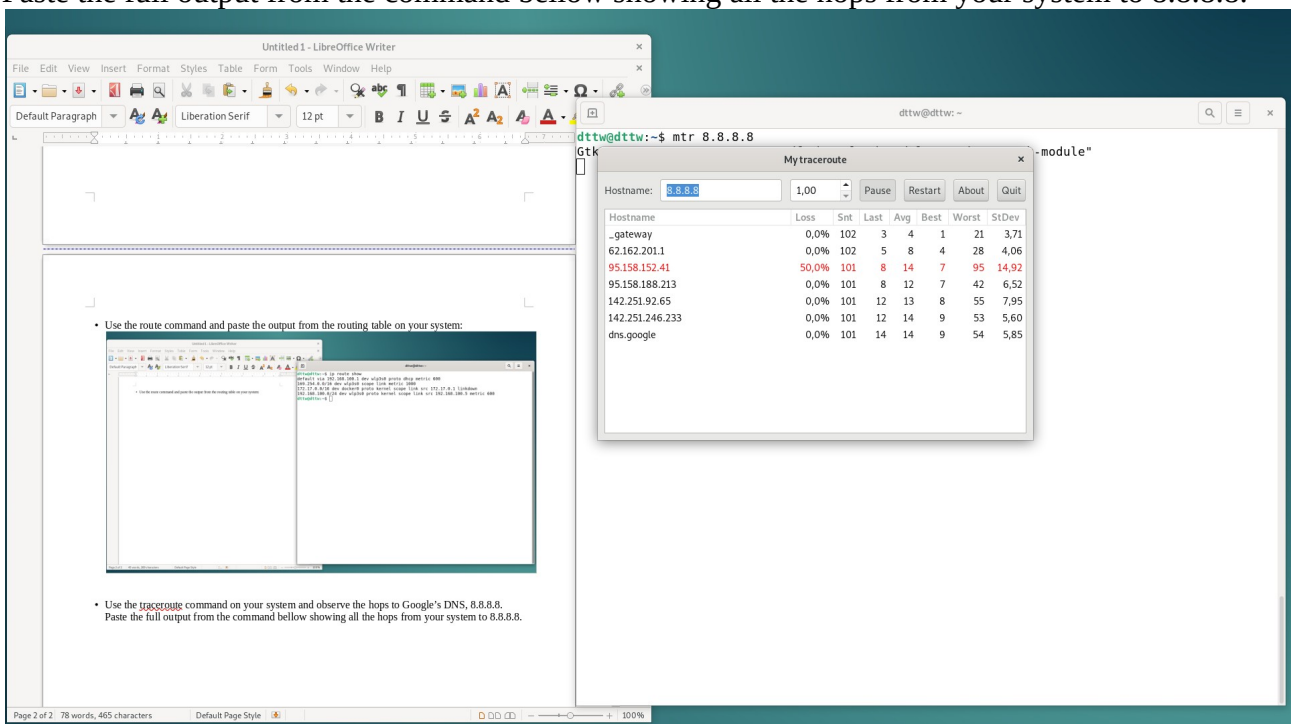
- Use the arp command and paste the output from the arp table on your system:



- Use the route command and paste the output from the routing table on your system:




- Use the traceroute command on your system and observe the hops to Google's DNS, 8.8.8.8. Paste the full output from the command below showing all the hops from your system to 8.8.8.8.

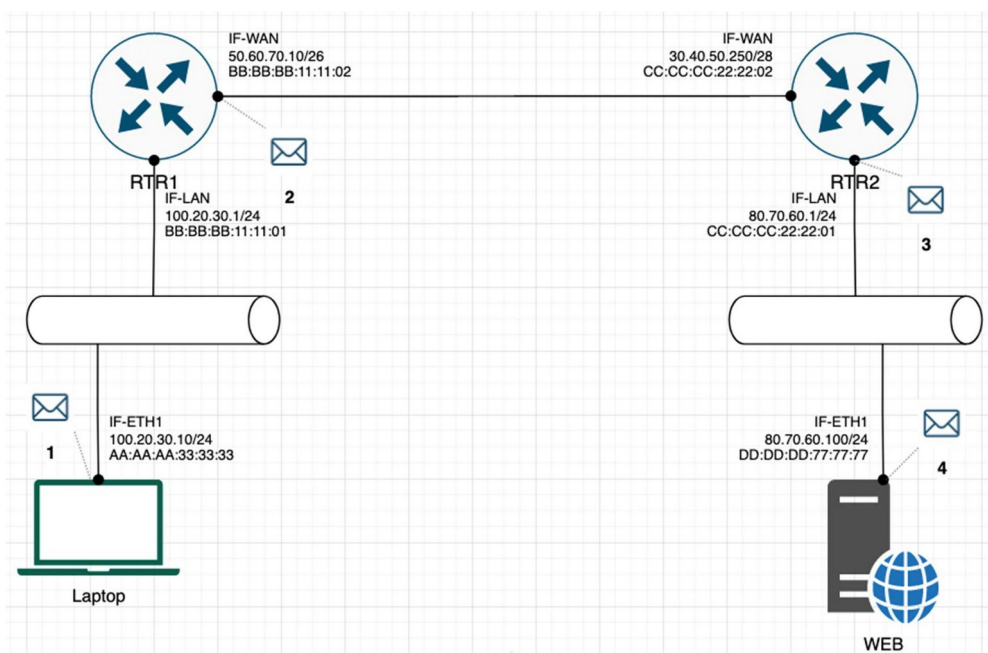


- Why would you need to use the ping command?
To test network connectivity, availability and latency between devices.
- Write down the TCP/UDP ports of the most commonly used services bellow in the form of TCP[PORT] or UDP[PORT].
HTTP – TCP80
SNMP – UDP161
HTTPS – TCP443
DNS client – TCP53, UDP53
DNS zone transfer – TCP53
SMTP – TCP25
SSH – TCP22
FTP – TCP21
Telnet – TCP23
MSSQL – TCP1443
MySQL – TCP3306
PostgreSQL – TCP5432
RDP (Remote Desktop Protocol) – TCP3389
NTP – UDP123
NFS – TCP2049, UDP2049

Exercise 2 – TCP/IP Basics

Difficulty: Medium

Refer to the exhibit and answer the questions below. The letter symbol , represents the IP packet as it travels across the network. In the example shown, the laptop attempts to communicate with the web server in question. During its travel the packet will be forwarded across the network nodes and will eventually end up across six network interfaces before it reaches the web server. Each packet as part of the TCP/IP Stack contains fields for the source and destination MAC Address, IP Address and the TCP/UDP Port.



For each of the packet locations shown, 1 to 4 write down the source and destination MAC addresses of the packet as it travels across the network interfaces.

1. The laptop initiates communication with the web server and prepares a packet. What would the packet look like at this stage?

- SRC IP – 100.20.30.10
- DST IP – 80.70.60.100
- SRC MAC – AA:AA:AA:33:33:33
- DST MAC – BB:BB:BB:11:11:01

2. RTR1 receives the packet on its IF-LAN interface, prepares it accordingly and forwards it out its IF-WAN. What would the packet look like at this stage?

- SRC IP – 100.20.30.10
- DST IP – 80.70.60.100
- SRC MAC – BB:BB:11:11:01
- DST MAC – CC:CC:CC:22:22:02

3. RTR2 receives the packet on its IF-WAN interface, prepares it accordingly and forwards it out via IF-LAN. What would the packet look like at this stage?

- SRC IP – 100.20.30.10
- DST IP – 80.70.60.100
- SRC MAC – CC:CC:CC:22:22:02
- DST MAC – CC:CC:CC:22:22:01

4. The web server receives the packet and prepares a response packet back. What would the packet look like at this stage?

- SRC IP – 80.70.60.100
- DST IP – 100.20.30.10
- SRC MAC – DD:DD:DD:77:77:77
- DST MAC – CC:CC:CC:22:22:01

Since we are talking about web traffic (www) in the example, which transport layer protocol will most probably be used?

- TCP
- UDP

If we do a traffic analysis with a network packet monitoring tool like WireShark, what can we expect to see for the source and destination ports when the laptop sends the packet?

SRC PORT: 1024 and above.

DST PORT:

Similarly, and vice versa, what can we expect to see as destination ports when the Web server sends a response packet back?

SRC PORT:

DST PORT: 1024 and above.

How many broadcast domains are there in the exhibit shown? _____

Exercise 3 – Traffic analysis and identifying the OSI layers of the network packets

Difficulty: Hard

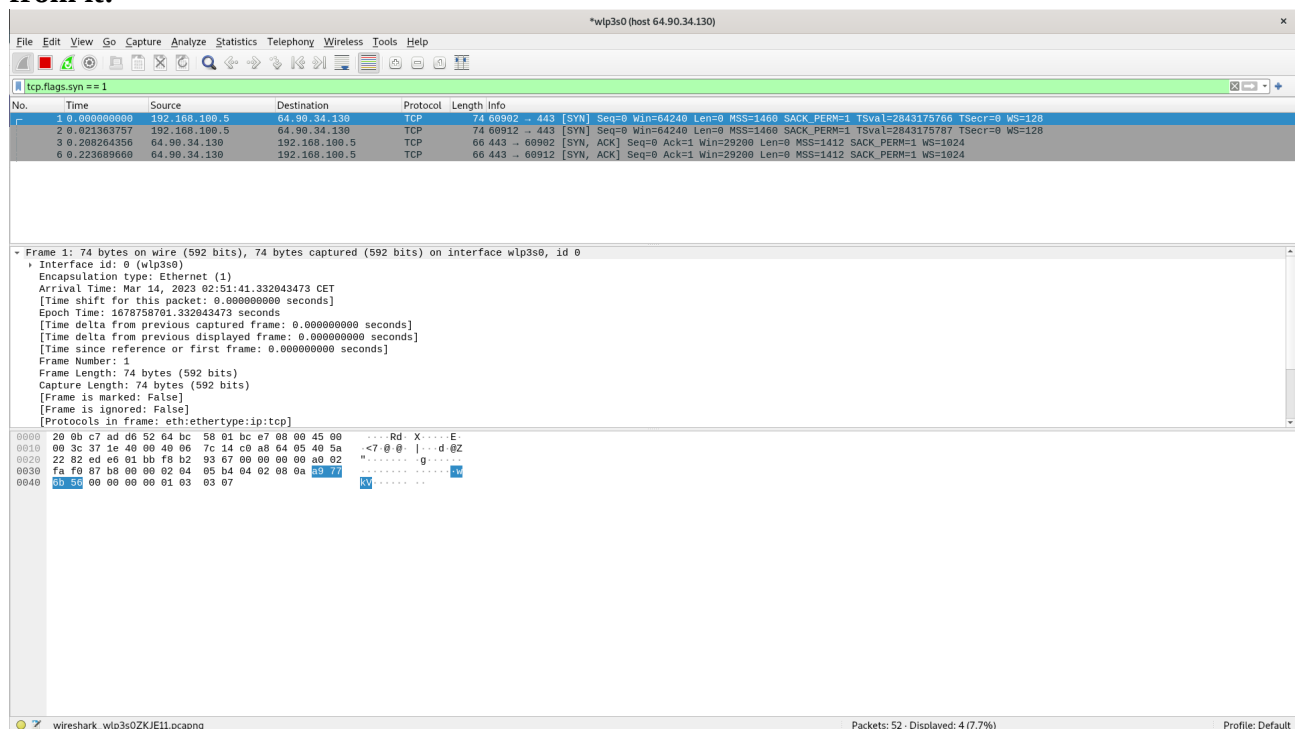
Prerequisite:

Search online and get familiar with the TCP's three-way handshake. Learn how to capture the three way handshake using Wireshark. Install Wireshark on your computer and use it to capture traffic against a website or a server or your choice. It is recommended that you capture traffic against a simple website. Name and the IP address of the website you plan to capture traffic:

Analyze the TCP's three-way handshake and using screenshots from the Wireshark window answer the questions below:

1. What is the source IP (of the initiating host): 192.168.100.5
2. What is the destination IP? (target website): 64.90.34.130

Identify the Network Interface (Layer 1 & 2) section of the SYN packet and paste a screenshot from it:



The screenshot shows the Wireshark interface with a capture on interface wlp3s0 (host 64.90.34.130). The packet list displays four packets related to a TCP three-way handshake:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.100.5	64.90.34.130	TCP	74	60902 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2843175766 TSecr=0 WS=128
2	0.021363757	192.168.100.5	64.90.34.130	TCP	74	60912 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2843175787 TSecr=0 WS=128
3	0.208264356	64.90.34.130	192.168.100.5	TCP	66	443 → 60902 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM=1 WS=1024
6	0.223689668	64.90.34.130	192.168.100.5	TCP	66	443 → 60912 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM=1 WS=1024

The packet details for frame 1 are expanded, showing the following sections:

- Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface wlp3s0, id 0
- Interface id: 0 (wlp3s0)
- Encapsulation type: Ethernet (1)
- Arrival Time: Mar 14, 2023 02:51:41.332043473 CET
- [Time shift for this packet: 0.000000000 seconds]
- Epoch Time: 1678750761.332043473 seconds
- [Time delta from previous captured frame: 0.000000000 seconds]
- [Time delta from previous displayed frame: 0.000000000 seconds]
- [Time since reference or first frame: 0.000000000 seconds]
- Frame Number: 1
- Frame Length: 74 bytes (592 bits)
- Capture Length: 74 bytes (592 bits)
- [Frame is marked: False]
- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:ip:tcp]

The packet bytes are displayed in hexadecimal and ASCII format:

```
0000 28 0b c7 ad d6 52 64 bc 58 01 bc e7 08 00 45 00 .....Rd: X.....E
0010 00 3c 37 1e 40 00 40 06 7c 14 c0 a8 64 05 40 5a ...<7 @ @: ]...d 0Z
0020 22 82 ed e6 61 bb f8 b2 93 67 00 00 00 00 a0 02 .....g.....
0030 fa f0 87 b8 00 00 02 04 05 04 04 02 00 0a 00 00 .....
0040 00 00 00 00 00 00 01 03 03 07 .....
```

Identify the Network Layer 3 section of the SYN/ACK packet and paste a screenshot from it:

The screenshot shows a Wireshark capture of a SYN/ACK packet. The packet list pane at the top shows a packet with the following details:

No.	Time	Source	Destination	Protocol	Length	Info
3	0.208264356	64.90.34.130	192.168.100.5	TCP	66	443 → 60902 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM=1 WS=1024
6	0.223689668	64.90.34.130	192.168.100.5	TCP	66	443 → 60912 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM=1 WS=1024

The packet details pane shows the following information:

- Frame 3: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface wlp3s0, id 0
- Ethernet II, Src: HuaweiEte-ad:d6:52 (28:0b:c7:ad:d6:52), Dst: IntelCor_01:bc:e7 (64:bc:58:01:bc:e7)
- Internet Protocol Version 4, Src: 64.90.34.130, Dst: 192.168.100.5
 - 0100 = Version: 4
 - 0101 = Header Length: 20 bytes (5)
 - Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
 - Total Length: 52
 - Identification: 0x0000 (0)
 - Flags: 0x40, Don't fragment
 - Fragment Offset: 0
 - Time to Live: 47
 - Protocol: TCP (6)
 - Header Checksum: 0xc43a [validation disabled]
 - [Header checksum status: Unverified]
 - Source Address: 64.90.34.130
 - Destination Address: 192.168.100.5
- Transmission Control Protocol, Src Port: 443, Dst Port: 60902, Seq: 0, Ack: 1, Len: 0

The packet bytes pane shows the raw data of the packet, including the Ethernet II header, Internet Protocol Version 4 header, and Transmission Control Protocol header.

Identify the Transport Layer 4 section of the ACK packet and paste a screenshot from it below:

The screenshot shows a Wireshark capture of an ACK packet. The packet list pane at the top shows a packet with the following details:

No.	Time	Source	Destination	Protocol	Length	Info
4	0.208288905	192.168.100.5	64.90.34.130	TCP	54	60902 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len=0
5	0.208460496	192.168.100.5	64.90.34.130	TLV1.3	571	Client Hello
6	0.223689668	64.90.34.130	192.168.100.5	TCP	66	443 → 60912 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM=1 WS=1024
7	0.223719908	192.168.100.5	64.90.34.130	TCP	54	60912 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len=0
8	0.223893121	192.168.100.5	64.90.34.130	TLV1.3	571	Client Hello
9	0.515558446	64.90.34.130	192.168.100.5	TCP	54	443 → 60902 [ACK] Seq=1 Ack=518 Win=30720 Len=0
10	0.525294474	64.90.34.130	192.168.100.5	TLV1.3	1466	Server Hello, Change Cipher Spec, Application Data
11	0.525343093	192.168.100.5	64.90.34.130	TCP	54	60902 → 443 [ACK] Seq=518 Ack=1413 Win=63104 Len=0
12	0.530077481	64.90.34.130	192.168.100.5	TCP	1466	443 → 60902 [ACK] Seq=1413 Ack=518 Win=30720 Len=1412 [TCP segment of a reassembled PDU]
13	0.530104427	192.168.100.5	64.90.34.130	TCP	54	60902 → 443 [ACK] Seq=518 Ack=2825 Win=63104 Len=0

The packet details pane shows the following information:

- Internet Protocol Version 4, Src: 192.168.100.5, Dst: 64.90.34.130
- Transmission Control Protocol, Src Port: 60902, Dst Port: 443, Seq: 1, Ack: 1, Len: 0
 - Source Port: 60902
 - Destination Port: 443
 - [Stream index: 0]
 - [TCP Segment Len: 0]
 - Sequence Number: 1 (relative sequence number)
 - Sequence Number (raw): 4172452712
 - [Next Sequence Number: 1 (relative sequence number)]
 - Acknowledgment Number: 1 (relative ack number)
 - Acknowledgment number (raw): 267193693
 - 0101 = Header Length: 20 bytes (5)
 - Flags: 0x010 (ACK)
 - Window: 562
 - [Calculated window size: 64256]
 - [Window size scaling factor: 128]
 - Checksum: 0x87a4 [unverified]
 - [Checksum Status: Unverified]
 - Urgent Pointer: 0

The packet bytes pane shows the raw data of the packet, including the Ethernet II header, Internet Protocol Version 4 header, and Transmission Control Protocol header.

Look closely at the L2 section of the three-way handshake packet details. Each of them shows the source and destination MAC address of the packets.

Who is the owner of the destination MAC address of the SYN packet?

20:0b:c7:ad:d6:52

Exercise 4 – Hacking mock-up (for Bonus points)

Difficulty: Very hard

Use Wireshark to capture the packet's application layer data and discover the implications of using unencrypted communication over a network.

It is recommended that you use your own Linux Virtual Machine on your system on which you need to configure a telnet server. From your own system try to login with a Telnet on the target VM all while capturing the traffic with a Wireshark. As a proof of competition for this exercise paste in bellow a screenshot of the application layer data containing visible username and password.

telnet -l username localhost

