Practicals Document

ICMP, ARP TCP/UDP Analysis using Wireshark

Learning Objectives

Upon completion of this practical, you will be able to:

- Use Ping utility working for diagnosing network connectivity issues in the Network layer
- Use Tracert utility to display the path to reach the destination host by sending ICMP echo request/echo reply messages
- Use Wireshark to examine the UDP header and TCP header
- Use Wireshark to examine the Address Resolution Protocol (ARP)

Scenario

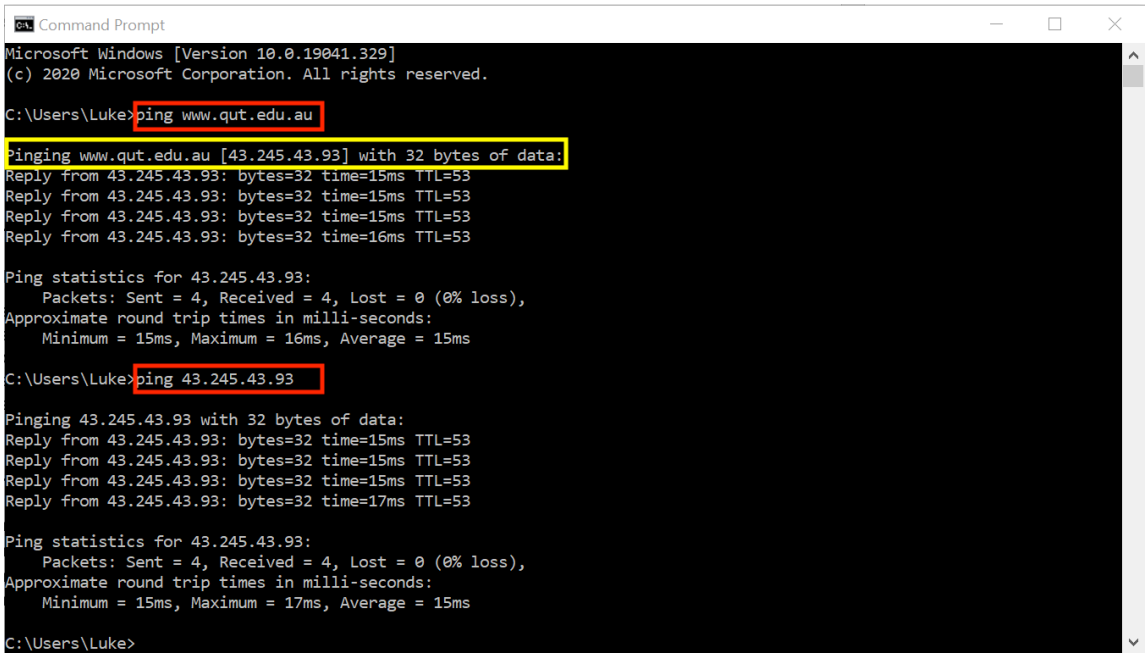
- **Lab Activity 1**: Use Ping and Tracert utilities
- **Lab Activity 2**: Examine ICMP messages
- **Lab Activity 3**: Examine UDP and TCP header fields and the TCP 3-way handshake operation
- **Lab Activity 4**: Examine ARP process

Lab Activity 1

Two utilities that are indispensable when testing TCP/IP network connectivity are **ping** and **tracert**. The ping utility is available on Windows, Linux and Cisco IOS to test network connectivity. The **tracert** utility is available on Windows, and a similar utility **traceroute** is available on Linux and Cisco IOS. Both **ping** and **tracert** operate by sending Internet Control Message Protocol (ICMP) messages to the destination host. ICMP is a TCP/IP Network layer protocol, first defined in RFC 792.

Task 1: Use the ping Command to Verify Simple TCP/IP Network Connectivity

The **ping** command operates by sending ICMP echo request messages to the destination host and then the destination host responds with echo reply messages. It can also be used to test domain name services (DNS) server functionality with an IP address or fully qualified domain name of the destination. For example,



Note: If you are completing this activity on campus, QUT's network has been setup to block ICMP messages. You may be able to send and receive ICMP probe messages to network hosts within the QUT network. If you are sending ICMP probe message to external hosts, you will not receive ICMP messages from external hosts.

Step 1: Verify TCP/IP Network layer connectivity on the LAN

1. In Windows, start the command prompt (cmd.exe).
2. Use the **ipconfig** command to display the TCP/IP information of your computer. Use the following table to record the information.

   From the command prompt, type **ipconfig** and then press the Enter key. **(Answers will vary)**

TCP/IP Information	Value
IPv4 Address	131.181.110.50
Subnet Mask	255.255.255.128
Default Gateway	131.181.110.1

Use the **ping** command to verify connectivity to the default gateway.

From the command prompt, ping your default gateway address.

By default, four ping request messages are sent to the destination host and then reply messages are returned. The output messages should look similar to **Figure 1**.

```
C:\>ping 131.181.110.1                                    ❶

Pinging 131.181.110.1 with 32 bytes of data:
Reply from 131.181.110.1: bytes=32 time<1ms TTL=255        ❷
Reply from 131.181.110.1: bytes=32 time<1ms TTL=255
Reply from 131.181.110.1: bytes=32 time<1ms TTL=255
Reply from 131.181.110.1: bytes=32 time<1ms TTL=255

Ping statistics for 131.181.110.1:                        ❸
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:            ❹
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figure 1: Output of the ping command.

❶ Destination address: the IP address or fully qualified domain name of the destination host.

❷ Reply information:

bytes – size of the ICMP packet.
time – elapsed time between transmission and reply.
TTL – default TTL value of the destination device, minus the number of routers in the path. The maximum TTL value is 255; the default TTL value of Windows machines is 128, Cisco IOS 255, and Linux computer 64.

❸ Summary information about the reply messages:

Packet Sent – number of packets transmitted.
Packet Received – number of packets received.
Packet Lost – difference between number of packets sent and received.

❹ Information about the delay in replies, measured in milliseconds.

Note: The round-trip times are all calculated as 0 ms. This is because of the low-resolution timer available to the program.

3. Fill in the results of the ping command on your computer: **(Answer may vary)**

Field	Value
Size of packet	32 bytes
Number of packets sent	4
Number of replies	4
Number of lost packets	0
Minimum delay	0ms
Maximum delay	0ms
Average delay	0ms

Step 2: Verify TCP/IP Network layer connectivity to a remote network

1. Use the **ping** command to verify connectivity to the QUT Website.

   From your command prompt, type **ping 43.245.43.93**

2. Fill in the results of the ping command on your computer: **(Answer may vary)**

Field	Value
Size of packet	32 bytes
Number of packets sent	4
Number of replies	4
Number of lost packets	0
Minimum delay	0ms
Maximum delay	0ms
Average delay	0ms

Note: You will not be able to ping the QUT Web site from the outside of the QUT's campus network. This is because the ICMP messages are blocked from the QUT's firewall for security reasons.

Task 2: Use the tracert Command to Verify TCP/IP Connectivity

The **tracert** command is useful for learning about network latency and path information. With Linux and Cisco IOS, the equivalent command is **traceroute**.

1. Use the **tracert** command to verify connectivity to reach the QUT Web site.

   From your Windows command prompt, type **tracert 43.245.43.93**

   Output should look similar to that shown in **Figure 2**. (answer may vary)

```
C:\>tracert 43.245.43.93

Tracing route to qut.squizedge.net [43.245.43.93]
over a maximum of 30 hops:

1     1 ms    1 ms    1 ms  R1 [10.10.10.1]
2     4 ms    3 ms    3 ms  10.20.22.127
3    16 ms   15 ms   16 ms  bri-pow-que-crt3-be-100.tpg.com.au [60.240.241.1]
4    19 ms   15 ms   16 ms  syd-apt-ros-crt1-be-50.tpg.com.au [203.219.107.74]
5    15 ms   15 ms   15 ms  syd-apt-ros-wgw1-be-10.tpgi.com.au [203.29.134.7]
6    15 ms   15 ms   16 ms  be300.sglebbrdr11.aapt.net.au [203.219.107.198]
7    16 ms   15 ms   16 ms  202.10.14.202
8    20 ms   15 ms   16 ms  pvadom.precisionglobal.asia [59.100.201.110]
9    16 ms   16 ms   15 ms  ae-14.r00.sydnau04.au.bb.gin.ntt.net [129.250.9.242]
10   15 ms   15 ms   24 ms  ae-0.r21.sydnau03.au.bb.gin.ntt.net [129.250.3.176]
11   15 ms   16 ms   15 ms  ae-7-63.r21.sydnau03.au.ce.gin.ntt.net [202.68.67.134]
12   17 ms   15 ms   17 ms  qut.squizedge.net [43.245.43.93]

Trace complete.
```

Figure 2: Output of the tracert command.

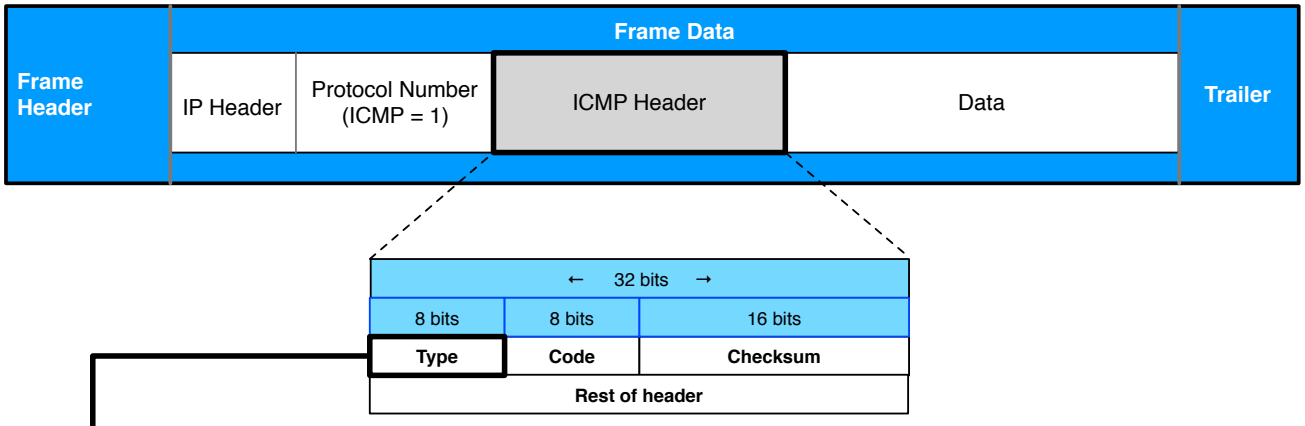
2. Fill in the results of the tracert command on your computer: **(Answers will vary)**

Field	Value
Maximum number of hops	30
First router IP address	10.10.10.1
Second router IP address	10.20.22.127
Third router IP address	60.240.241.1
Fourth router IP address	203.219.107.74
Destination reached?	Yes / No

Lab Activity 2

Internet Control Message Protocol (ICMP) is an error reporting and diagnostic protocol. It is an essential part of any IP implementation. Therefore, understanding ICMP and knowing what can possibly generate a specific type of ICMP is very useful in diagnosing network problems.

Task 1: Understand the Format of ICMP Messages



Type Value	Meaning
0	Echo Reply
3	Destination Unreachable
8	Echo Request

Figure 3: ICMP Message Format.

Figure 3 shows the ICMP message format and also lists out three commonly used ICMP types 0, 3 and 8. Each ICMP message starts with an 8-bit **Type** field, an 8-bit **Code** field, and then followed by a computed 16-bit **Checksum** field.

Figure 4 lists three common types of ICMP messages with some associated code. The **Code** field provides further information to the **Type** field about nature of problem. For example, when an ICMP error message is returned, the Type field contains a value of 3 and the Code Value field contains a value of 3 as well. This error message means that the error message indicates the destination port was not available by the destination host.

Type Value	Code Value	Meaning
0	0	Echo Reply
3	0	Network unreachable
	1	Host unreachable
	2	Protocol unreachable
	3	Port unreachable
	4	Fragment needed and DF is set
8	0	Echo Request

Figure 4: ICMP Message Types and Codes.

Task 2: Use Wireshark to Capture and Examine ICMP Messages

Step 1: Capture ICMP messages

1. Start Wireshark and then choose the correct interface to capture network traffic.
2. From the command prompt, type **ping 43.245.43.93** and then press Enter.
3. After the ping replies are returned, stop the Wireshark packet capture.

Step 2: Analyse the ICMP header fields

1. From Wireshark, view the packets listed on the Packet List Pane. Focus on the packets that are labelled as "**Echo (ping) request**" and "**Echo (ping) reply**" in the Info column.
2. Select the first ICMP echo request message from the Packet List Pane.

   Expand the Internet Control Message Protocol layer from the Packet Detail Pane, and then use the following table to record this request message.

Field	Value
Type	8 (Echo (ping) request)
Code	0
Checksum	0x4d60 (answer may vary)
Identifier(BE)	0x0001 (answer may vary)
Sequence number(BE)	1

3. Select the first ICMP echo reply message form the Packet List Pane.

   Use the following table to record this reply message.

Field	Value
Type	0 (Echo (ping) reply)
Code	0
Checksum	0x5560 (answer may vary)
Identifier(BE)	0x0001 (answer may vary)
Sequence number(BE)	1

Q1: Compare the field values of these two messages to determine if the field values are different?
___Type and Checksum fields_____.

Lab Activity 3

The two protocols in the TCP/IP Transport layer are the user datagram protocol (UDP) defined in RFC 768, August 1980, and transmission control protocol (TCP) defined in RFC 761, January 1980. Both protocols support upper-layer protocol communication. UDP is used to provide unreliable and connectionless datagram services to DNS and DHCP without the need of acknowledgments and retransmissions. But TCP provides reliable and connection oriented datagram service for higher level protocols, such as HTTP, SMTP and FTP.

Task 1: Understand the UDP Header Fields

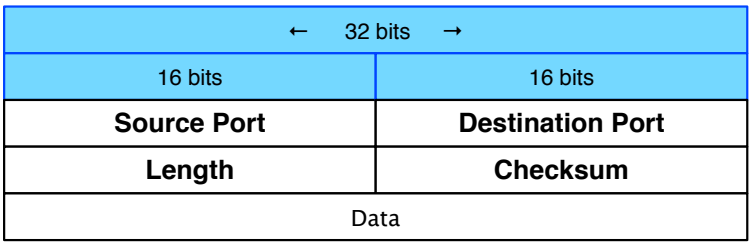


Figure 5: UDP Header.

UDP has little overhead compared to TCP. Some applications use UDP rather than TCP for fast, lightweight, unreliable data delivery.

Task 2: Use Wireshark to Capture DNS traffic and Examine UDP Header Fields

Step 1: Capture DNS traffic

1. Start Wireshark and then choose the correct interface to capture network traffic.
2. Clear DNS cache on your computer.

   From the command prompt, type **ipconfig /flushdns** and then press Enter.

3. From the command prompt, type **ping www.qut.edu.au** and then press Enter.
4. After the ping replies are returned, stop the Wireshark packet capture.

Step 2: Analyse the UDP header fields

1. From Wireshark, in the **Filter** field, enter **udp** and then click on the **Apply** button.

2. Select the first DNS packet that is labelled as "**Standard query A www.qut.edu.au**".

   Expand the Internet Protocol layer and User Datagram Protocol layer from the Packet Detail Pane. Use the following table to record the information of this packet:

Field	Value
Source IP address	131.181.110.50 (answer may vary)
Destination IP address	131.181.59.48 (answer may vary)
Source port number	54676 (answer may vary)
Destination port number	domain (53)

3. Select the returned packet from the DNS query, which is labelled as "**Standard query response A 43.245.43.93**".

   Use the following table to record the information of this packet.

Field	Value
Source IP address	131.181.59.48 (answer may vary)
Destination IP address	131.181.110.50 (answer may vary)
Source port number	domain (53)
Destination port number	51645 (answer may vary)

Task 3: Understand the TCP Header Fields

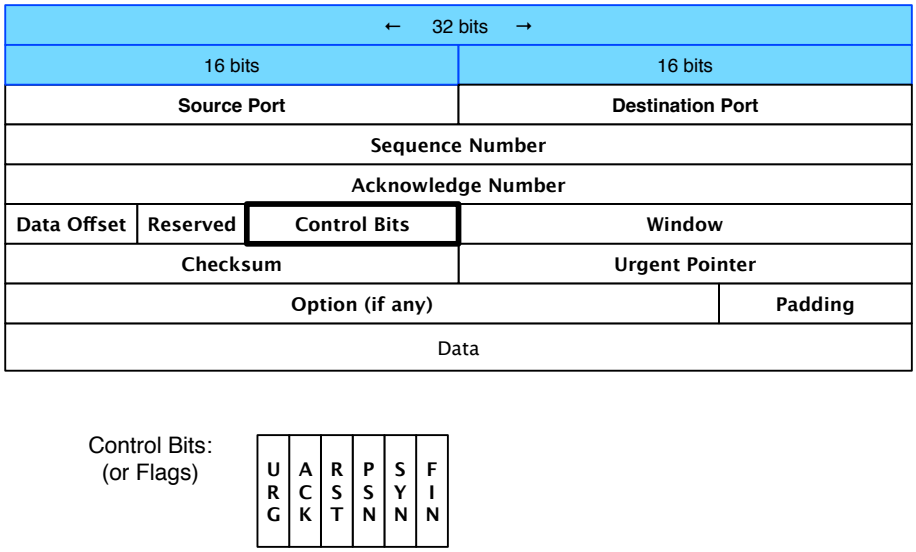


Figure 6: TCP Header.

Referring to **Figures 5 and 6**, TCP manages communication much differently from UDP, but reliability and guaranteed delivery requires additional control over the communication channel.

- **Control Bits (or Flags)** have a special meaning in session management and in the treatment of segments. Among interesting values are:
  - ACK – Acknowledgement of a received segment
  - SYN – Synchronise; only set when a new TCP session is negotiated during the TCP 3-way handshake.
  - FIN – Finish; request to close the TCP session.

Task 4: Use Wireshark to Capture TCP 3-way Handshake Traffic and Examine TCP Header Fields

Step 1: Capture TCP 3-way handshake traffic

1. Disable “Relative sequence numbers” in Wireshark.

   Start **Wireshark** > Click on **Edit** menu > **Preferences** > **Protocols** > **TCP** > Unselect **Relative sequence numbers** > **OK**.

   > **Note:** Wireshark converse all Sequence Number (SEQ) and Acknowledge Numbers (ACK) into relative numbers. Namely, all SEQ and ACK numbers always start at zero (0) for the first packet seen in each conversation. This makes the numbers much smaller and easier to read and compare to the real numbers, which are normally initialised to randomly selected numbers during the SYN phase. To gain a better understanding of TCP process, we need to disable this feature in Wireshark.

2. After you start the Wireshark Capture, then open up a Web browser to visit http://www.bom.gov.au to capture a 3-way handshake dialogue.

3. Stop the Wireshark packet capturing.

Step 2: Analyse the TCP header fields

You are required to observe a 3-way handshake. Web browsers will potentially open many connections from a single request to load the various elements of the page. You must use your knowledge gained from the lecture to match the sequence and acknowledgement numbers to determine which packets are related to the connection you are looking at. Packets may arrive in different orders, or other unrelated packets may arrive first. Rarely will the 3-way handshake display consecutively and neatly in Wireshark.

1. From Wireshark, in the **Filter** field, enter **tcp.port == 80** and then click on the **Apply** button.

2. Select the first TCP packet that is labelled as "**[SYN]**".

   Expand the Internet Protocol Version 4 layer and Transmission Control Protocol layer from the Packet Detail Pane. Use the following table to record the information of this packet.

Field	Value
Source IP address	131.181.110.50 (answer may vary)
Destination IP address	131.181.196.36
Source port number	51207 (answer may vary)
Destination port number	http (80)
Sequence number	1856199414 (answer may vary)
Acknowledgment number	N/A
Flags – Acknowledgement	0 (Not set)
Flags – Syn	1 (Set)
Flags – Fin	0 (Not set)

3. Locate the first packets corresponding reply labelled as "**[SYN, ACK]**".

   Use the following table to record the information of this packet.

Field	Value
Source IP address	131.181.196.36
Destination IP address	131.181.110.50 (answer may vary)
Source port number	http (80)
Destination port number	51207 (answer may vary)
Sequence number	788478365 (answer may vary)
Acknowledgment number	1856199414 (answer may vary)
Flags – Acknowledgement	1 (Set)
Flags – Syn	1 (Set)
Flags – Fin	0 (Not set)

4. Locate the final TCP packet of the 3-way handshake labelled as "**[ACK]**".

   Use the following table to record the information of this packet.

Field	Value
Source IP address	131.181.110.50 (answer may vary)
Destination IP address	131.181.196.36
Source port number	51207 (answer may vary)
Destination port number	http (80)
Sequence number	1856199415 (answer may vary)
Acknowledgment number	788478366 (answer may vary
Flags – Acknowledgement	1 (Set)
Flags – Syn	0 (Not Set)
Flags – Fin	0 (Not set)

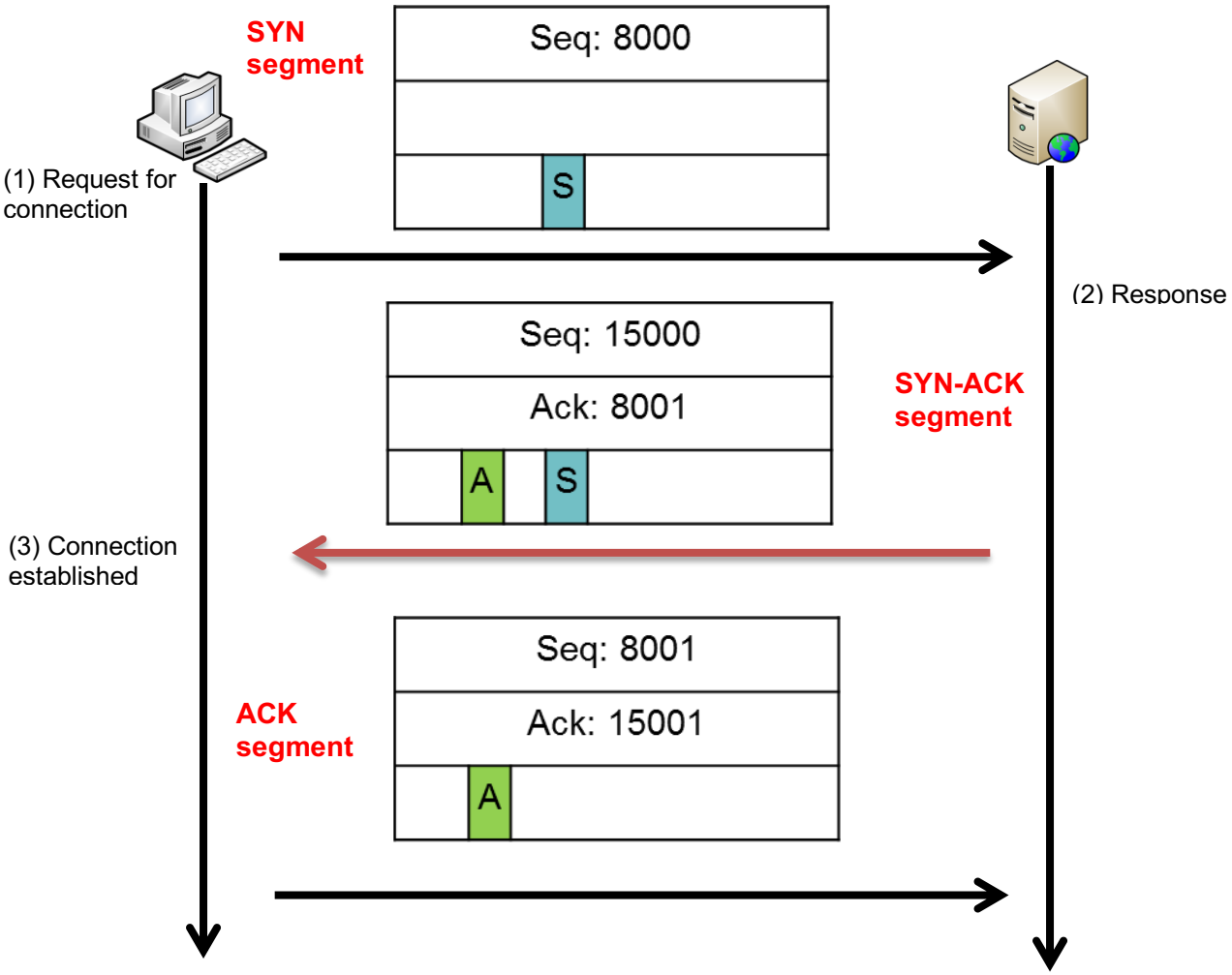
Step 3: Describe a TCP 3-way handshake dialogue

Describe the dialogue between a client and server when establishing a TCP connection. Include an appropriate diagram with your explanation.

1. The client sends the first segment, which is a SYN segment with the SYN bit set to the server. The value of the sequence number field is called the initial sequence number. The SYN segment is a control segment and carries no data, however it consumes one sequence number, i.e. data transfers start with ISN +1.

2. The server responds with a SYN-ACK segment with the SYN and ACK bits set. The segment carries the following functions:
   - The segment provides the initial sequence number (ISN) for communications from server to client. The ISN is incremented for the first data transfer
   - The segment provides acknowledgement of the receipt of the SYN segment sent by the client.

3. The client replies with an ACK segment with ACK bit set. The segment provides an acknowledgement to the server's SYN-ACK segment. The sequence number used in this segment is the same as the acknowledgment number used in the server's SYN-ACK segment.



Lab Activity 4

Address Resolution Protocol (ARP) is used to map a Network layer IP address to a Data-link layer MAC address. When a frame is placed on the network, it must have a destination MAC address to communicate on the network. To dynamically discover the MAC address of the destination host, an ARP request is broadcast on the LAN. The destination host responds with its MAC address via unicast. Then this information is recorded in ARP cache. Every device on the LAN keeps its own ARP cache to avoid the need to ARP for every packet being sent to an IP address. ARP cache entries remain until not being used or ARP cache times out.

Task 1: Manage the Local ARP Cache

This activity requires two computers in the same LAN. You can use the two virtual machines from Practical 1 operating in Bridged Mode (at home) or NAT mode (at QUT) if you don't have two PCs.

Step 1: Examine the IP address and MAC (physical) address

1. From both PC1 and PC2, click **Start** > type **cmd** in the Start menu search box > Right-click on **Command Prompt** > Select **Run as administrator**.

2. From the command prompt, type **ipconfig /all** and then press Enter.

   Use the following table to record the IPv4 addresses and corresponding MAC (physical) addresses of PC1 and PC2. (answers may vary)

Device	IPv4 Address	MAC (Physical) Address
PC1	131.181.110.50	B8-CA-3A-7E-F8-9F
PC2	131.181.110.59	B8-CA-3A-7F-02-48

3. From the command prompt, type **arp –a** to view the current ARP cache.

4. From PC1:

   a) Type **arp –d** to flush the existing ARP cache.
   b) Use the **ping** command to send ICMP messages to PC2.
   c) Type **arp –a** to view and record the output of type dynamic: (answers may vary)

Internet Address	Physical Address	Type
131.181.110.1	10-27-E9-48-49-E1	dynamic
131.181.110.59	B8-CA-3A-7F-02-48	dynamic

5. From PC2:

   a) Type **arp –d** to flush the existing ARP cache.
   b) Use the **ping** command to send ICMP messages to PC1.
   c) Type **arp –a** to view and record the output of type dynamic: (answers may vary)

Internet Address	Physical Address	Type
131.181.110.1	10-27-E9-48-49-E1	dynamic
131.181.110.50	B8-CA-3A-7E-F8-9F	dynamic

Task 2: Use Wireshark to Observe the ARP Process

Step 1: Capture ARP messages on PC1

1. From PC1, choose the right interface to capture network traffic.
2. Click **Start** > type **cmd** in the Start menu search box > Right-click on **cmd.exe** > Select **Run as administrator**.
3. From the command prompt, type **arp –d** again to flush the existing ARP cache.
4. Use the **ping** command to send ICMP messages to PC2.
5. After the ping replies are returned, stop the Wireshark packet capture.

Step 2: Examine ARP messages on PC1

1. From Wireshark, in the **Filter** field, enter **icmp || arp && eth.addr ==** *PC1-physical-address* and then click on the **Apply** button. Through this filter string, the Packet List Pane will only display the ICMP or ARP messages associated with PC1's physical address.

   Note that you must enter PC1's physical address with the format as shown in **Figure 6.6.**

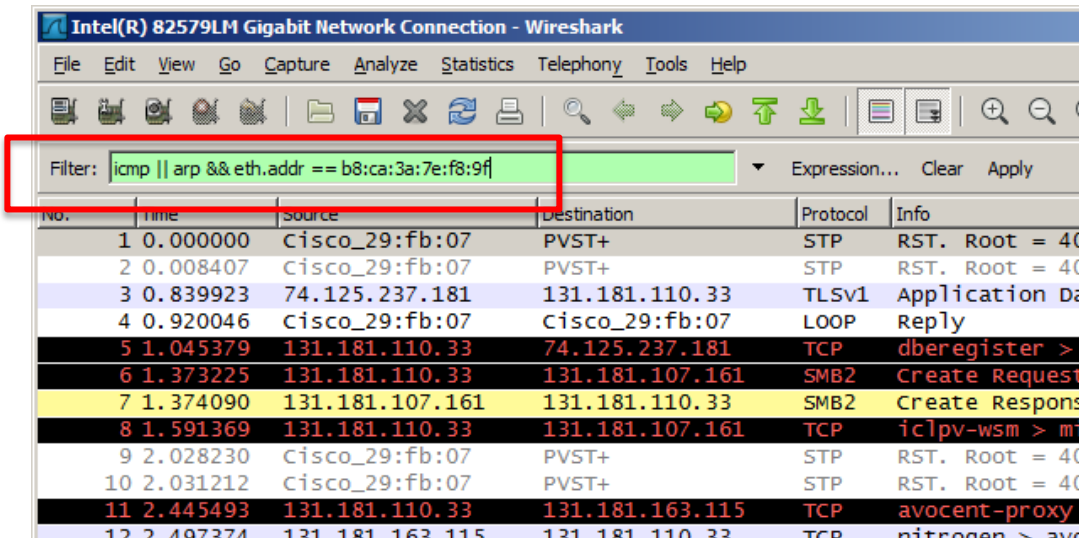


Figure 6.6: Wireshark Sample Display Filter String.

2. Use the following table to record first ARP message. (answer may vary)

Field	Value
Sender MAC Address	B8-CA-3A-7E-F8-9F
Sender IP Address	131.181.110.50
Target MAC Address	00:00:00:00:00:00
Target IP Address	131.181.110.59

3. Use the following table to record the second ARP message. (answer may vary)

Field	Value
Sender MAC Address	B8-CA-3A-7F-02-48
Sender IP Address	131.181.110.59
Target MAC Address	B8-CA-3A-7E-F8-9F
Target IP Address	131.181.110.50

Q2: Between the ARP request and reply packets, how to identify one is an ARP request and the other is an ARP reply?

You can look at the information at layer 2 and layer 3.

- Layer 2: An ARP request is broadcast on the network with the destination address FF-FF-FF-FF-FF-FF.
- Layer 3: the value of the Target MAC address is 00:00:00:00:00:00

Note: Frames are addressed to reach every computer on a given LAN, if they are addressed to MAC address FF:FF:FF:FF:FF:FF

Information for the instructors: Students may ask that the Ethernet II frame for an ARP request is a broadcast address (FF:FF:FF:FF:FF:FF), why does the Target MAC address contain all 0s (00:00:00:00:00:00)? This is because ARP request is sent to ask the MAC address of the destination, as it is unknown it contains zeros as placeholders.

End of the lab activities

End of Document