Lab Answers

Daniel Aoulou, Kaushal Patel

# Task 2 – Create a Filter

This lab assumes you are using a you two real or virtual computers that can communicate over a real or virtual network. Virtual Box allows for a bridged networking to supply a second computer in addition to your host computer, or that you have access to WCU. This is the cleanest way to do this as the additional network traffic is minimal.

Set up Wireshark so that it has a filter looking for TCP packets from the IP address of the host and guest operating system so that you can capture the packets going back and forth between the two computers when using your communication application.

Type a copy of your filter below in a color other than black:

(ip.addr == 100.110.31.177 && ip.addr == 100.110.31.171) && tcp

(ip.addr == <host ipv4> && ip.addr == <guest ipv4>) && tcp

# Task 3 Analyze Communication

Using your new filter, capture packets while facilitating communication between the host and the guest operating system using your application:

1. Start your communication application on the host and guest operating system.
2. Start capturing packets in Wireshark.
3. First the host sends a message, “Hello guest operating system” that the guest receives.
4. Secondly, the guest sends a message, “Hello host operating system” that the host receives.
5. Stop the capturing and save this list of packets as a PCAPNG file called **capture1.pcapng** in the same folder as your Python program.

Remember too that the port number 65432 will be used, but you will need to carefully analyze the network traffic while using your application to see what other port numbers are used.

Type your responses to each question in a color such as green, blue or purple that stands out.

Q1: What port number(s) are used and in what context? For steps 3 and 4 describe for each communication in detail what TCP port numbers seem to be used to facilitate the communication and who is using them i.e. guest or host OS:

65432, the port we designated, is always used to receive the message being sent. The messages are being sent from ports 52219 and 37162

Q2: Are the port numbers the same every time the programs are run? If they are not, what changes?

The port numbers are not always the same; while the port we designated is always used to receive the actual message that is being sent. The sender may use other ports however to send the message, namely 52219 and 37162

Q3: How many packets are exchanged between the host and the guest operating system before the packet containing the payload is sent?

3 packets:

2062 (52219 -> 65432)

2063 (65432-> 52219)

2064 (52219 -> 65432)

Q4: How many packets are exchanged between the host and the guest operating system before the packet containing #<<END>># is sent?

5 packets:

2062-2064

2065 (Payload, 52219 -> 65432)

2066 (65432-> 52219)

Q5: In that instance, who sends the packet containing #<<END>># the guest or the host OS and why?

The packet containing #<<END>># is sent from the guest is sent to the host to signify that the message has been received. This is how it was designed in the python program we wrote.

Q6: In one round of communication between the host and guest, is the packet containing #<<END>># the last packet sent? Give your answer and a detailed reason for your answer.

No; in the first round of communication, the last package that is sent is 2071, with flags FIN and ACK. This is because TCP has a four way teardown process when a socket is closed, which involves the four packets that are sent after the #<<END>># packet is sent.

Q7: Some packets contain the flag SYN. Describe roughly what is going on at this point. This may require some additional research to answer.

SYN is part of the three-way handshake done by TCP when initializing a connection. It’s used for making sure everything is ‘synchronized’, which is important in making sure there are no duplicate packets, packets are in order, etc.

Q8: Assuming you knew how to generate packets using Wireshark, describe in detail how you might be able to disrupt or interfere with this communication?

You could orchestrate a man in the middle attack by intercepting packets and manipulating the flags to simulate acknowledgement from the destination machine. You could also alter the bytes in the packets before sending it to its original destination. You could even stop communication completely by sending an SYN packet with an invalid source or destination port number. This will cause the receiving host to send an RST packet, which will abort the TCP connection.

Q9: How long does the message being sent need to be before it uses more than one packet to send? Explain your answer. You may have to do some research to answer this.

The size limit for a message to be sent in a single packet depends on the Maximum Transmission Unit (MTU) of the underlying network. Our research found that the standard MTU ranges from 68 bytes (for Ipv4) to a maximum of 9000 bytes; if a message exceeds the MTU, it will be fragmented into multiple packets for transmission.

Q10: On your virtual machine, initiate a new capture session that only filters for TCP packets. Then take the built-in Firefox browser to **bbc.co.uk**. Once the page has fully loaded, stop the capture of packets in Wireshark. *Note, if you see the word Mozilla, that is referring to the Firebox browser.*

1. Pick a couple of fairly unique words from the home page (not including BBC). Can you find any packets with this content? Yes/No and explain your answer and say what words you looked for.

1. Do a filter **TCP** packets containing BBC and take a look at the first packet using the HTTP protocol. Besides your **IP** address, what information does that leak that a potential hacker might find useful?

Information such as the operating system that was used to make the request, the byte encoding by the requester, the DNS server used which could give location information.

1. What did you find most surprising about the captured packets?

The packets themselves seem to be encrypted; no plaintext is shown.