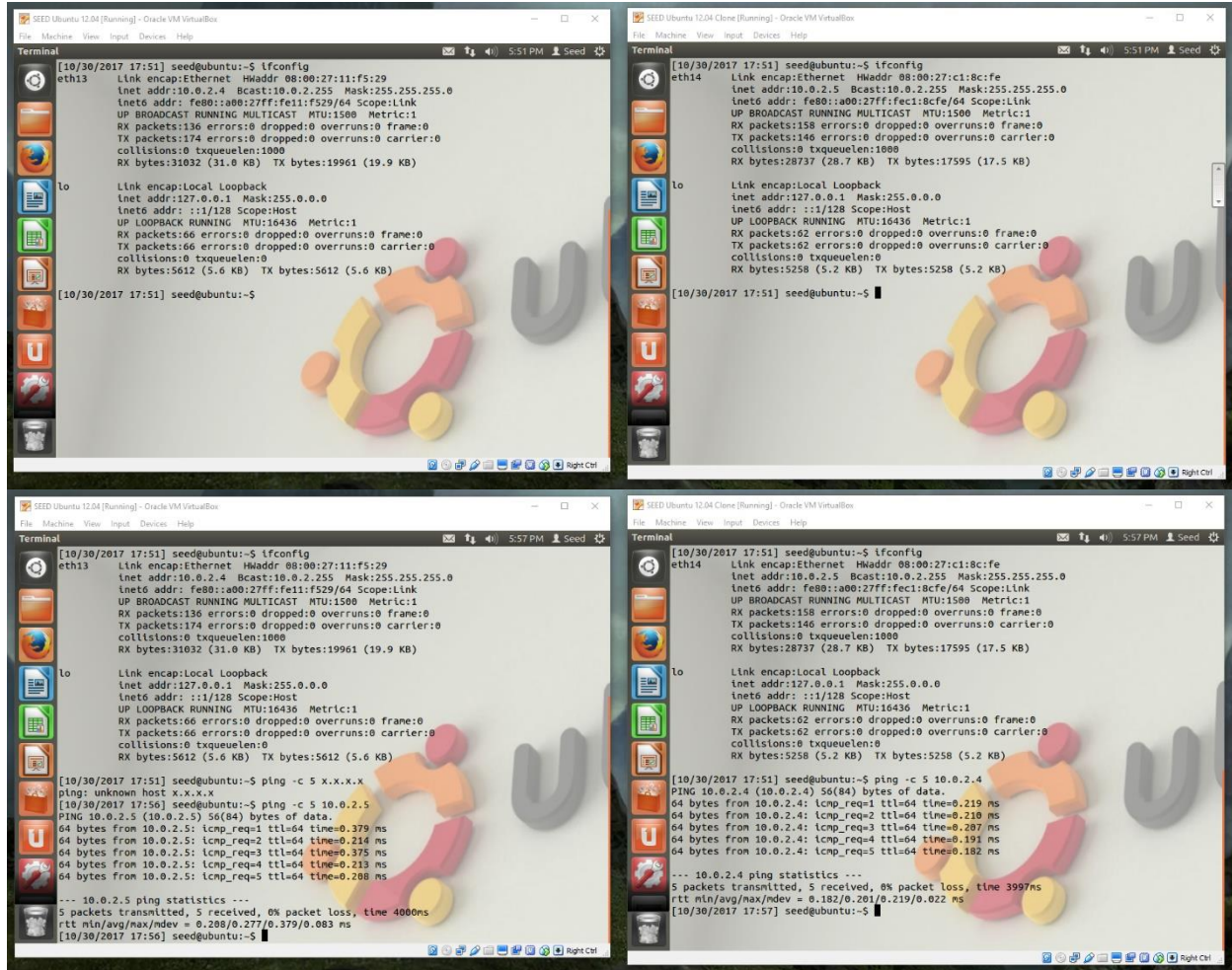


William Kenji Kiplinger

EX5

<https://github.com/Kenjum/CS380-EX5/>

Problem 1: Verifying the Network



The image displays four screenshots of a terminal window within a virtual machine environment, showing network configuration and testing steps.

Top Left Screenshot: The terminal shows the output of the `ifconfig` command for the `eth13` interface. The configuration includes the link encap (Ethernet), hardware address (08:00:27:11:f5:29), IP address (10.0.2.4), broadcast address (10.0.2.255), and mask (255.255.255.0). It also shows the loopback interface `lo` with IP address 127.0.0.1 and mask 255.0.0.0.

Top Right Screenshot: The terminal shows the output of the `ifconfig` command for the `eth14` interface. The configuration includes the link encap (Ethernet), hardware address (08:00:27:c1:8c:fe), IP address (10.0.2.5), broadcast address (10.0.2.255), and mask (255.255.255.0). It also shows the loopback interface `lo` with IP address 127.0.0.1 and mask 255.0.0.0.

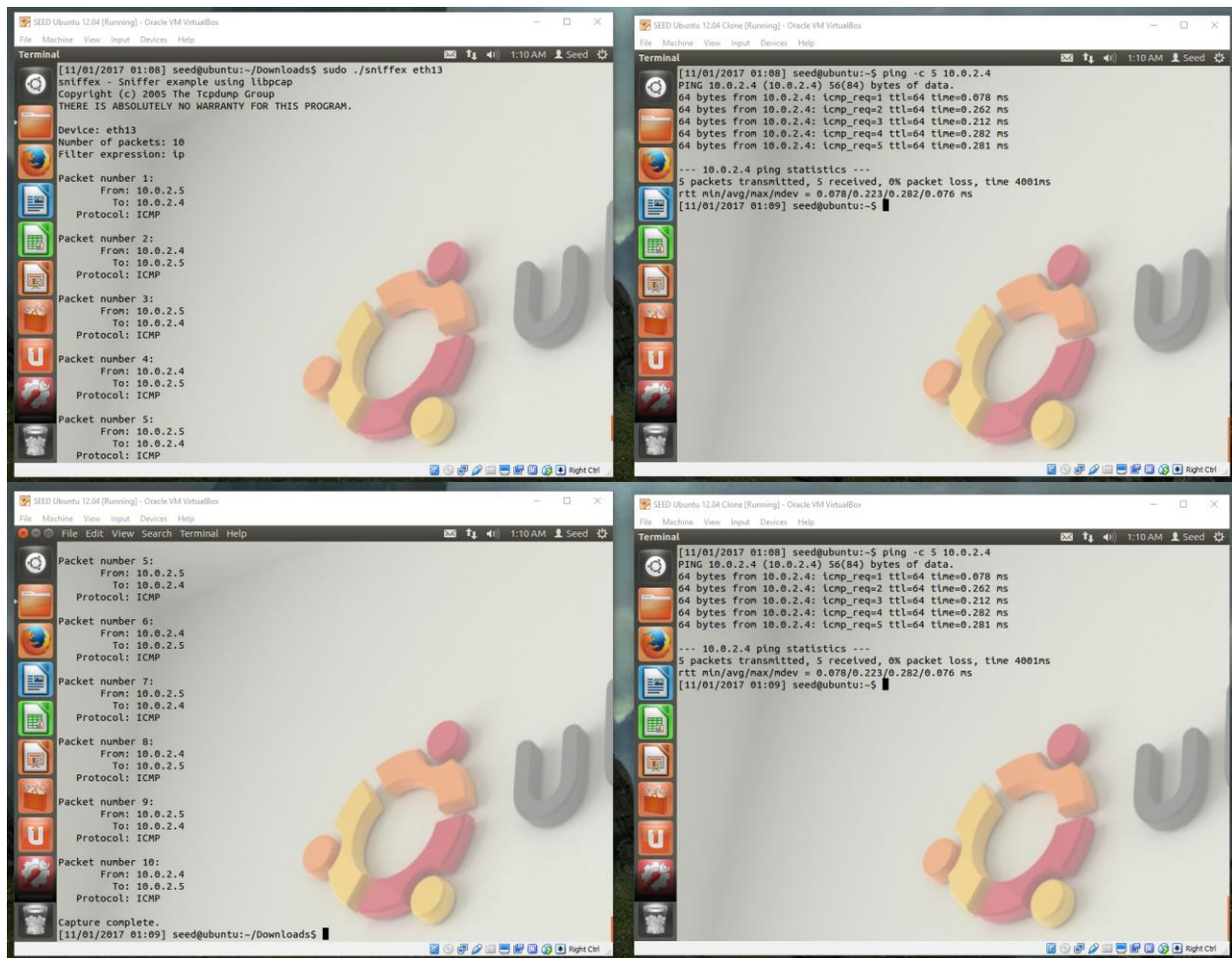
Bottom Left Screenshot: The terminal shows the output of the `ping -c 5 x.x.x.x` command, which results in a "ping: unknown host x.x.x.x" error. It then shows the output of the `ping -c 5 10.0.2.5` command, which successfully pings the host 10.0.2.5, showing 56(84) bytes of data and a round-trip time of approximately 0.379 ms.

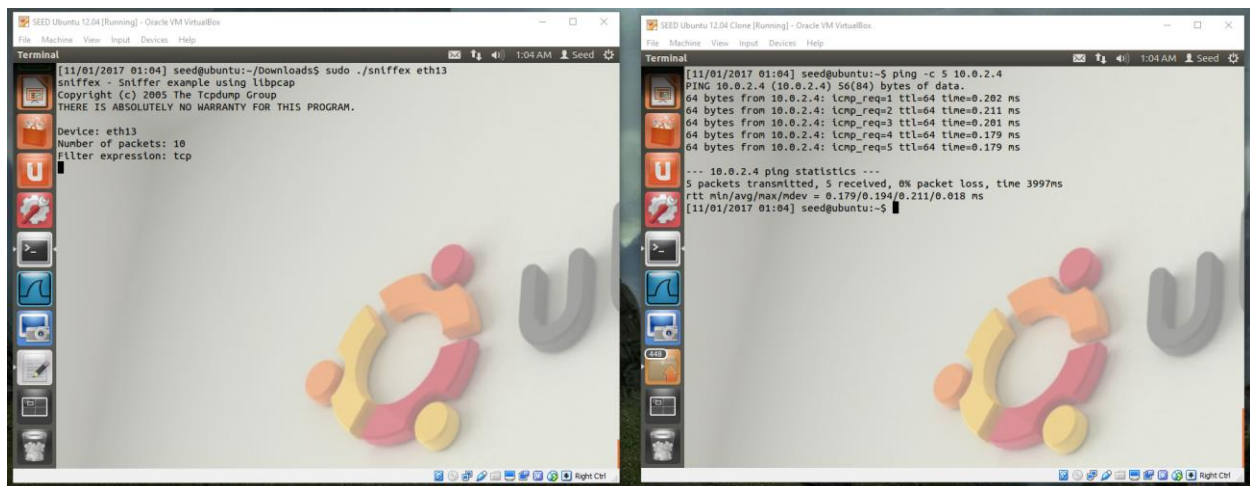
Bottom Right Screenshot: The terminal shows the output of the `ping -c 5 10.0.2.4` command, which successfully pings the host 10.0.2.4, showing 56(84) bytes of data and a round-trip time of approximately 0.219 ms. It then shows the output of the `ping -c 5 10.0.2.5` command, which successfully pings the host 10.0.2.5, showing 56(84) bytes of data and a round-trip time of approximately 0.219 ms.

Problem 2: Writing a Packet Sniffer

Summary of pcap library use:

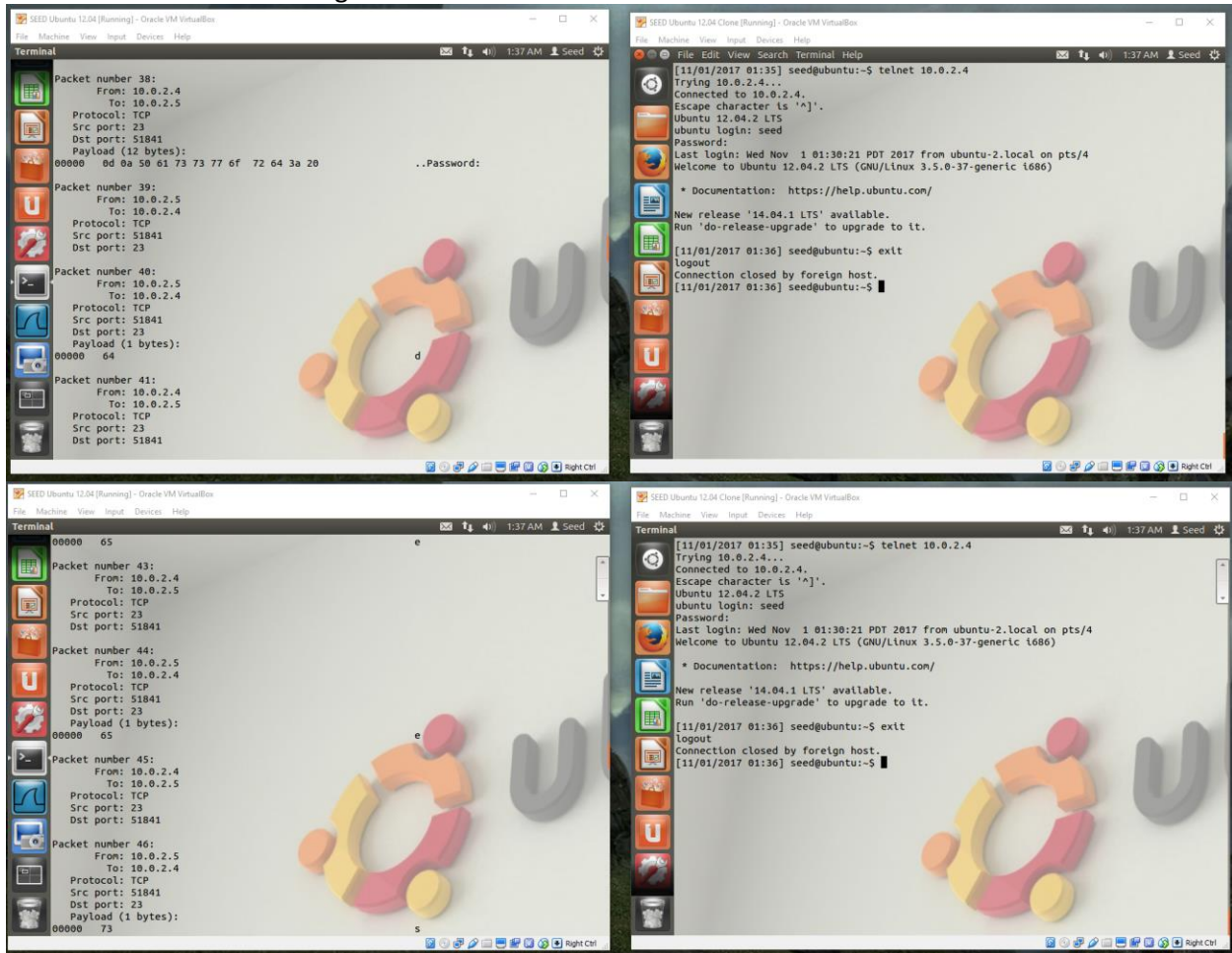
After the interface to be sniffed is chosen, either the device is defined in a string or we can ask pcap to provide the name of an interface that will do the job. Pcap is then initialized once it knows which device(s) it will be sniffing via file handles for sessions. If you want to get specific, you create a rule set, “compile” it, and apply it to whichever session that will be filtered. Lastly, pcap enters an execution loop where it waits until it has received however many packets desired. For every time, it gets a new packet in, it calls another function that we have already defined.





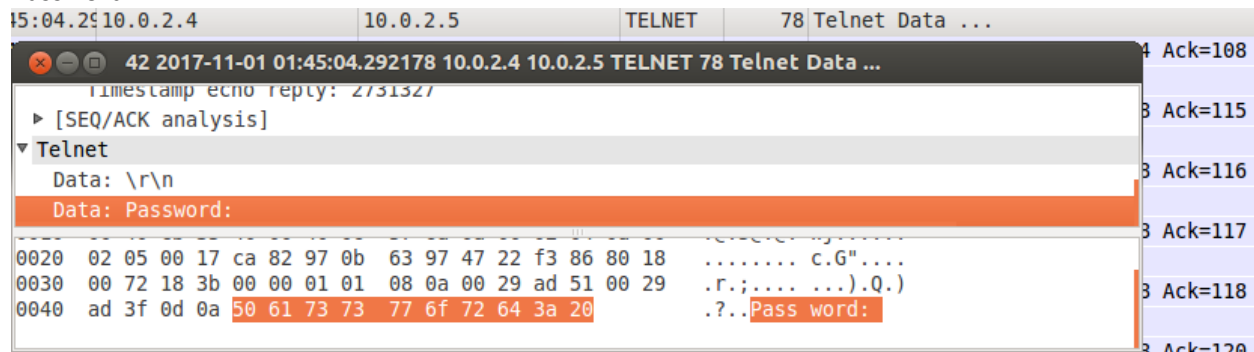
ICMP files are being sent over so with the tcp filter on, nothing appears.

Problem 3: Password Sniffing



it recorded the password: "dees"

Using Wireshark
Password

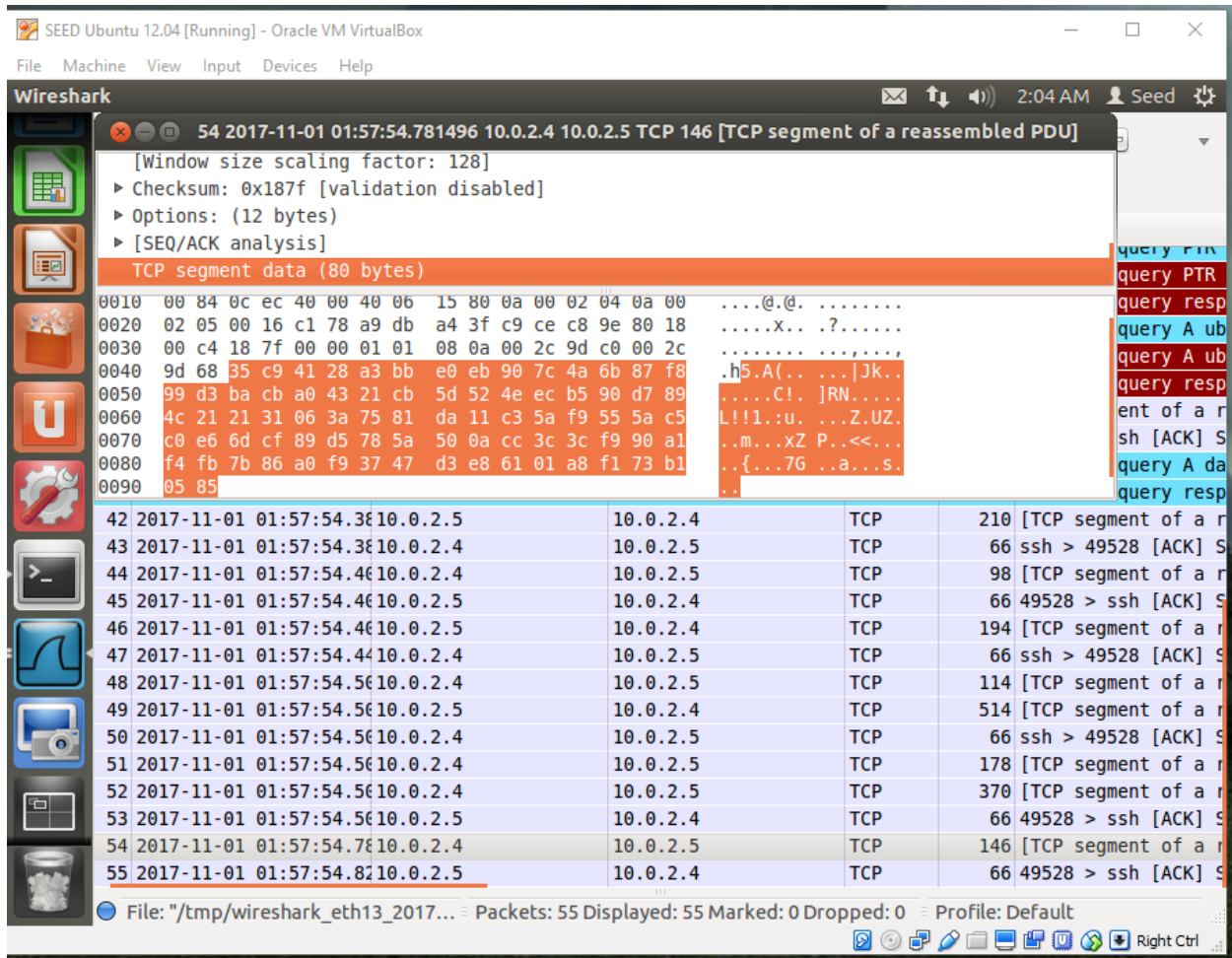


d

5:06:07	10.0.2.5	10.0.2.4	TELNET	67	Telnet Data ...	
44	2017-11-01 01:45:06.075707	10.0.2.5 10.0.2.4	TELNET	67	Telnet Data ...	8 Ack=115
Timestamp value: 2731772						
Timestamp echo reply: 2731345						
▶ [SEQ/ACK analysis]						
▼ Telnet						
Data: d						
0020	02 04 ca 82 00 17 47 22 f3 86 97 0b 63 a3 80 18G"C...				8 Ack=116
0030	00 73 9d a6 00 00 01 01 08 0a 00 29 ae fc 00 29	.S.....)....)				8 Ack=117
0040	ad 51 64	.0				8 Ack=118
e						
5:06:74	10.0.2.5	10.0.2.4	TELNET	67	Telnet Data ...	
46	2017-11-01 01:45:06.748012	10.0.2.5 10.0.2.4	TELNET	67	Telnet Data ...	8 Ack=116
Timestamp value: 2731940						
Timestamp echo reply: 2731801						
▶ [SEQ/ACK analysis]						
▼ Telnet						
Data: e						
0020	02 04 ca 82 00 17 47 22 f3 87 97 0b 63 a3 80 18G"C...				8 Ack=117
0030	00 73 9a 35 00 00 01 01 08 0a 00 29 af a4 00 29	.S.5....)....)				8 Ack=118
0040	af 19 65	..e				8 Ack=120
e						
5:07:33	10.0.2.5	10.0.2.4	TELNET	67	Telnet Data ...	
48	2017-11-01 01:45:07.337234	10.0.2.5 10.0.2.4	TELNET	67	Telnet Data ...	8 Ack=117
Timestamp value: 2732088						
Timestamp echo reply: 2731959						
▶ [SEQ/ACK analysis]						
▼ Telnet						
Data: e						
0020	02 04 ca 82 00 17 47 22 f3 88 97 0b 63 a3 80 18G"C...				8 Ack=118
0030	00 73 99 02 00 00 01 01 08 0a 00 29 b0 38 00 29	.S.....).8.)				8 Ack=120
0040	af b7 65	..e				8 Ack=110
e						
5:08:23	10.0.2.5	10.0.2.4	TELNET	67	Telnet Data ...	
50	2017-11-01 01:45:08.235210	10.0.2.5 10.0.2.4	TELNET	67	Telnet Data ...	8 Ack=118
Timestamp value: 2732312						
Timestamp echo reply: 2732107						
▶ [SEQ/ACK analysis]						
▼ Telnet						
Data: s						
0020	02 04 ca 82 00 17 47 22 f3 89 97 0b 63 a3 80 18G"C...				8 Ack=120
0030	00 73 89 8d 00 00 01 01 08 0a 00 29 b1 18 00 29	.S.....)....)				8 Ack=110
0040	b0 4b 73	.K				8 Ack=179
S						

After doing these tests, it's surprising how easy it is to access information that is passing through. At the same time, it makes a lot of sense how this works and is possible. You're just simply taking a deeper look at the packets being transmitted. This really shows the importance of encryption.

Problem 4: SSH



Instead of Wireshark being able to pick up the user input key by key, SSH had everything come in as a chunk. The data is also encrypted and I currently don't have any means of deciphering it.