INTRODUCTION TO NETWORK TRAFFICANALYSIS

Yunis Mohamed

Hack The Box module

Table of Contents

Introduction

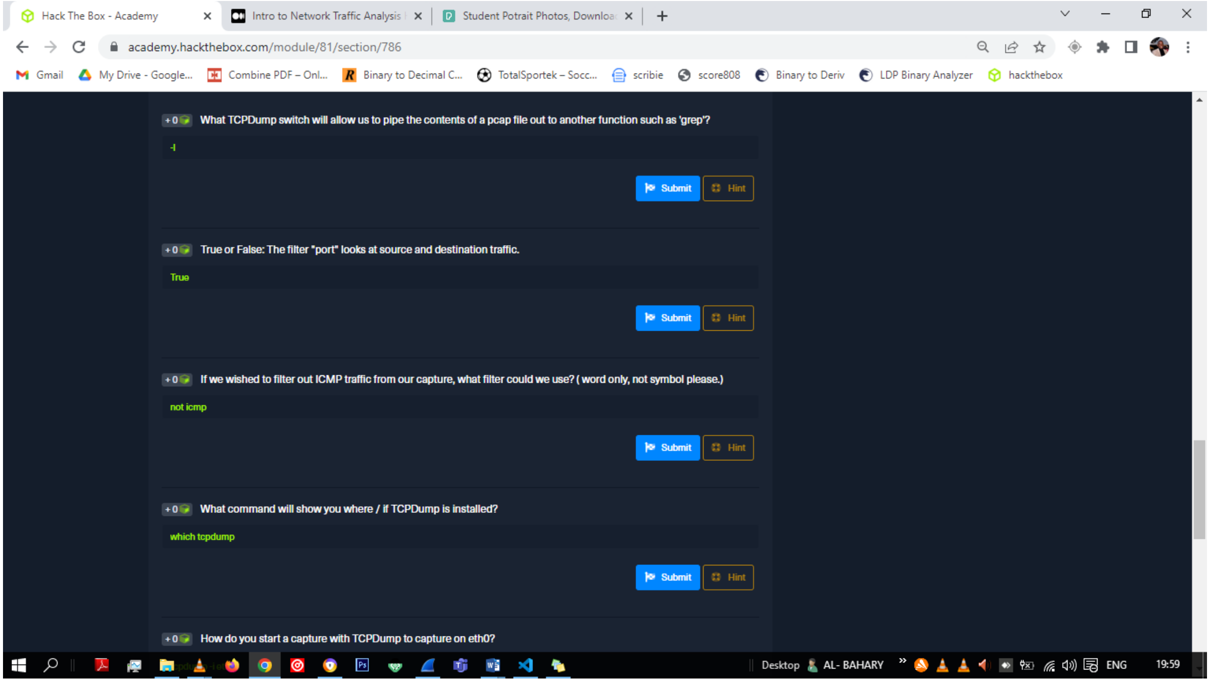
Network Traffic Analysis (NTA) can be described as the act of examining network traffic to characterize common ports and protocols utilized, establish a baseline for our environment, monitor and respond to threats, and ensure the greatest possible insight into our organization's network. This process helps security specialists determine anomalies, including security threats in the network, early and effectively pinpoint threats. Network Traffic Analysis can also facilitate the process of meeting security guidelines. Tools such as Wireshark and tcpdump usage to be able to sniff out sensitive data on a network.

Capturing with Tcpdump

The purpose of this lab is to expose us to tcpdump and give us time to familiarize ourselves with the terminal and utilizing tools within it. We will practice various tcpdump basics such as reading from and writing to files, utilizing basic switches, and locating files in the terminal. While completing these labs, we can explore and practice using different switches and functionality within tcpdump. When comfortable, take some time and try to determine if we can make out any traffic visible to us on the network.

1. What TCPDump switch will allow us to pipe the contents of a pcap file out to another function such as 'grep'?-I
2. True or False: The filter "port" looks at source and destination traffic True
3. If we wished to filter out ICMP traffic from our capture, what filter could we use? (Word only, not symbol please.)not icmp
4. What command will show you where / if TCPDump is installed? Which tcmpdump
5. How do you start a capture with TCPDump to capture on eth0? tcpdump -i eth0
6. What switch will provide more verbosity in your output?-V

7. What switch will write your capture output to a .pcap file?-W
8. What switch will read a capture from a .pcap file?-r
9. What switch will show the contents of a capture in Hex and ASCII?-x



+0 What TCPDump switch will allow us to pipe the contents of a pcap file out to another function such as 'grep'?

-l

+0 True or False: The filter "port" looks at source and destination traffic.

True

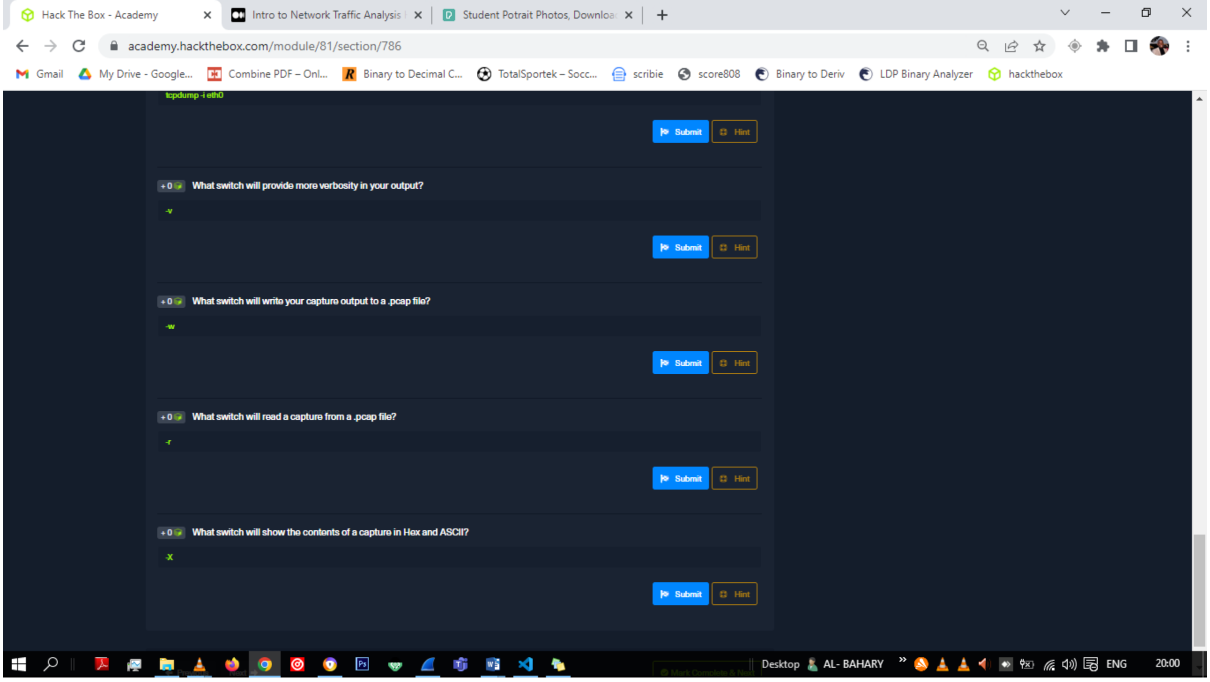
+0 If we wished to filter out ICMP traffic from our capture, what filter could we use? (word only, not symbol please.)

not icmp

+0 What command will show you where / if TCPDump is installed?

which tcpdump

+0 How do you start a capture with TCPDump to capture on eth0?



tcpdump -i eth0

+0 What switch will provide more verbosity in your output?

-v

+0 What switch will write your capture output to a .pcap file?

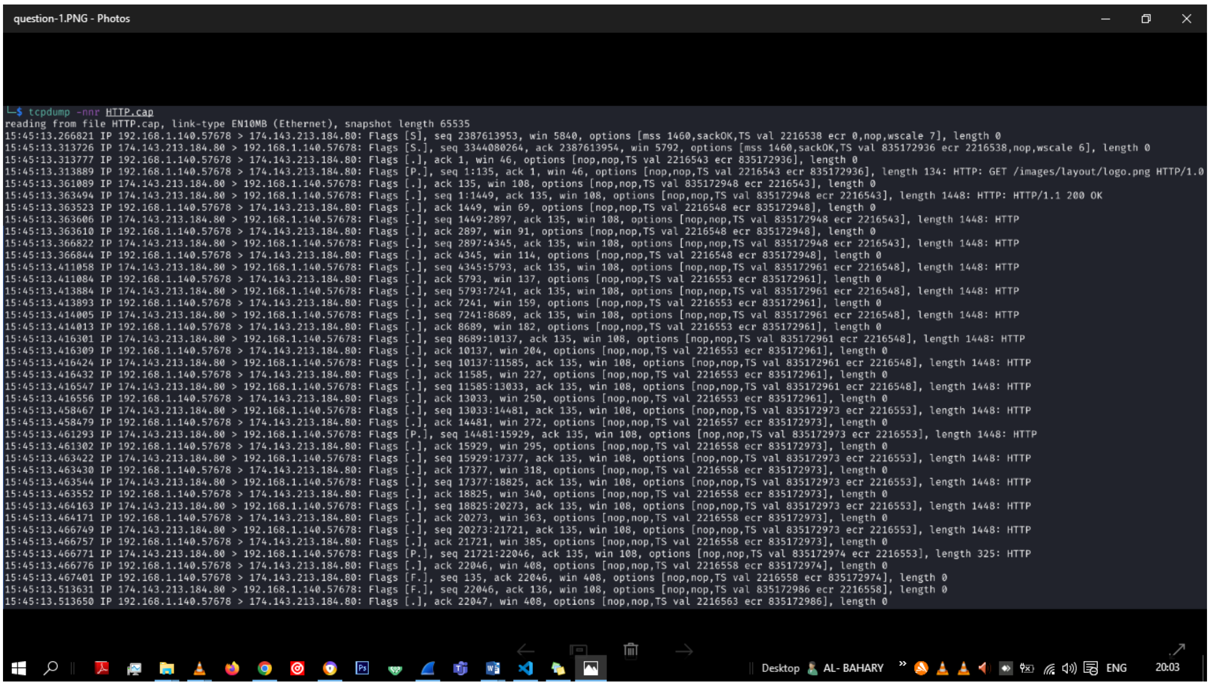
-w

+0 What switch will read a capture from a .pcap file?

-r

+0 What switch will show the contents of a capture in Hex and ASCII?

-X



```
└─$ tcpdump -nnr HTTP.cap
reading from file HTTP.cap, link-type EN10MB (Ethernet), snapshot length 65535
15:45:13.266821 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [S], seq 2387613953, win 5840, options [mss 1460,sackOK,TS val 2216538 ecr 0,nop,wscale 7], length 0
15:45:13.313726 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [S.], seq 3344080264, ack 2387613954, win 5792, options [mss 1460,sackOK,TS val 835172936 ecr 2216538,nop,wscale 6], length 0
15:45:13.313777 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 1, win 46, options [nop,nop,TS val 2216543 ecr 835172936], length 0
15:45:13.313889 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [P.], seq 1:135, ack 1, win 46, options [nop,nop,TS val 2216543 ecr 835172936], length 134: HTTP: GET /images/layout/logo.png HTTP/1.0
15:45:13.361089 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], ack 135, win 108, options [nop,nop,TS val 835172948 ecr 2216543], length 0
15:45:13.363494 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 1:1449, ack 135, win 108, options [nop,nop,TS val 835172948 ecr 2216543], length 1448: HTTP: HTTP/1.1 200 OK
15:45:13.363523 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 1449, win 69, options [nop,nop,TS val 2216548 ecr 835172948], length 0
15:45:13.363606 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 1449:2897, ack 135, win 108, options [nop,nop,TS val 835172948 ecr 2216543], length 1448: HTTP
15:45:13.363610 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 2897, win 91, options [nop,nop,TS val 2216548 ecr 835172948], length 0
15:45:13.366822 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 2897:4345, ack 135, win 108, options [nop,nop,TS val 835172948 ecr 2216543], length 1448: HTTP
15:45:13.366844 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 4345, win 114, options [nop,nop,TS val 2216548 ecr 835172948], length 0
15:45:13.411058 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 4345:5793, ack 135, win 108, options [nop,nop,TS val 835172961 ecr 2216548], length 1448: HTTP
15:45:13.411084 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 5793, win 137, options [nop,nop,TS val 2216553 ecr 835172961], length 0
15:45:13.413884 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 5793:7241, ack 135, win 108, options [nop,nop,TS val 835172961 ecr 2216548], length 1448: HTTP
15:45:13.413893 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 7241, win 159, options [nop,nop,TS val 2216553 ecr 835172961], length 0
15:45:13.414005 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 7241:8689, ack 135, win 108, options [nop,nop,TS val 835172961 ecr 2216548], length 1448: HTTP
15:45:13.414013 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 8689, win 182, options [nop,nop,TS val 2216553 ecr 835172961], length 0
15:45:13.416301 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 8689:10137, ack 135, win 108, options [nop,nop,TS val 835172961 ecr 2216548], length 1448: HTTP
15:45:13.416309 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 10137, win 204, options [nop,nop,TS val 2216553 ecr 835172961], length 0
15:45:13.416424 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 10137:11585, ack 135, win 108, options [nop,nop,TS val 835172961 ecr 2216548], length 1448: HTTP
15:45:13.416432 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 11585, win 227, options [nop,nop,TS val 2216553 ecr 835172961], length 0
15:45:13.416547 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 11585:13033, ack 135, win 108, options [nop,nop,TS val 835172961 ecr 2216548], length 1448: HTTP
15:45:13.416556 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 13033, win 250, options [nop,nop,TS val 2216553 ecr 835172961], length 0
15:45:13.458467 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 13033:14481, ack 135, win 108, options [nop,nop,TS val 835172973 ecr 2216553], length 1448: HTTP
15:45:13.458479 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 14481, win 272, options [nop,nop,TS val 2216557 ecr 835172973], length 0
15:45:13.461293 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [P.], seq 14481:15929, ack 135, win 108, options [nop,nop,TS val 835172973 ecr 2216553], length 1448: HTTP
15:45:13.461302 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 15929, win 295, options [nop,nop,TS val 2216558 ecr 835172973], length 0
15:45:13.463422 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 15929:17377, ack 135, win 108, options [nop,nop,TS val 835172973 ecr 2216553], length 1448: HTTP
15:45:13.463430 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 17377, win 318, options [nop,nop,TS val 2216558 ecr 835172973], length 0
15:45:13.463544 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 17377:18825, ack 135, win 108, options [nop,nop,TS val 835172973 ecr 2216553], length 1448: HTTP
15:45:13.463552 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 18825, win 340, options [nop,nop,TS val 2216558 ecr 835172973], length 0
15:45:13.464163 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 18825:20273, ack 135, win 108, options [nop,nop,TS val 835172973 ecr 2216553], length 1448: HTTP
15:45:13.464171 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 20273, win 363, options [nop,nop,TS val 2216558 ecr 835172973], length 0
15:45:13.466749 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [.], seq 20273:21721, ack 135, win 108, options [nop,nop,TS val 835172973 ecr 2216553], length 1448: HTTP
15:45:13.466757 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 21721, win 385, options [nop,nop,TS val 2216558 ecr 835172973], length 0
15:45:13.466771 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [P.], seq 21721:22046, ack 135, win 108, options [nop,nop,TS val 835172974 ecr 2216553], length 325: HTTP
15:45:13.466776 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 22046, win 408, options [nop,nop,TS val 2216558 ecr 835172974], length 0
15:45:13.467401 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [F.], seq 135, ack 22046, win 408, options [nop,nop,TS val 2216558 ecr 835172974], length 0
15:45:13.513631 IP 174.143.213.184.80 > 192.168.1.140.57678: Flags [F.], seq 22046, ack 136, win 108, options [nop,nop,TS val 835172986 ecr 2216558], length 0
15:45:13.513650 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [.], ack 22047, win 408, options [nop,nop,TS val 2216563 ecr 835172986], length 0
```

Tcpdump Packet Filtering

Tcpdump provides a robust and efficient way to parse the data included in our captures via packet filters. This section will examine those filters and get a glimpse at how it modifies the output from our capture.

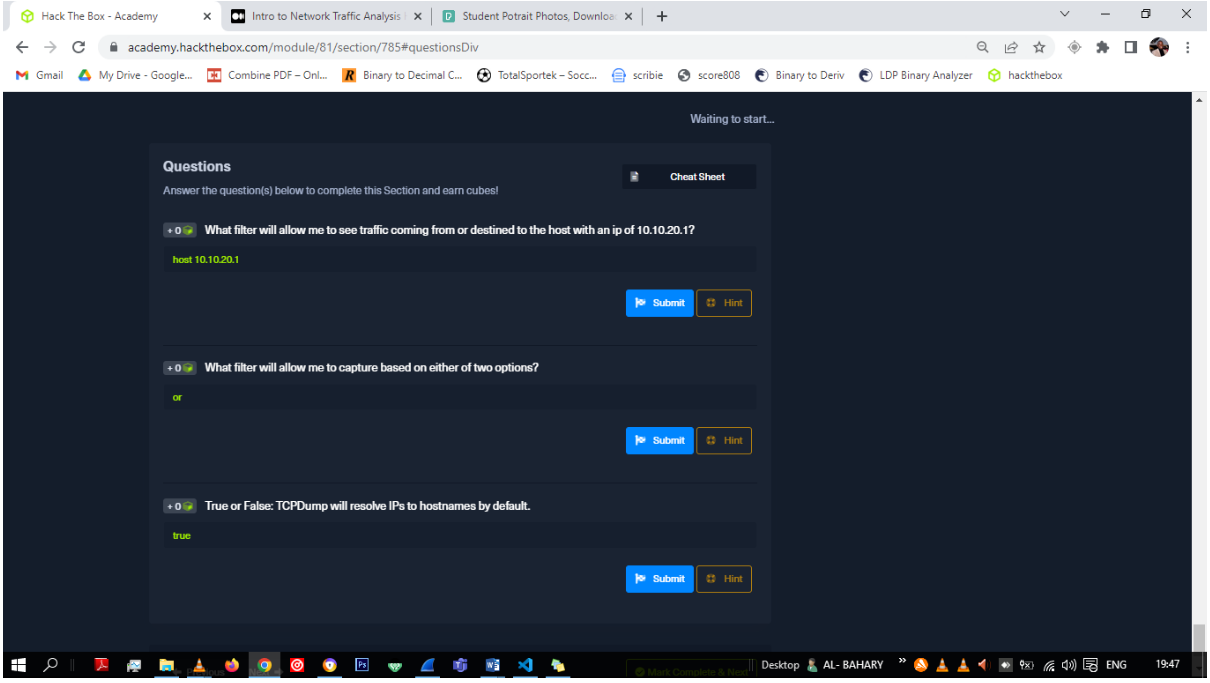
1. What filter will allow me to see traffic coming from or destined to the host with an ip of 10.10.20.1? host 10.10.20.1

Host will filter visible traffic to show anything involving the designated host. Bi-directional.

2. What filter will allow me to capture based on either of two options? Or

Or allows for a match on either of two conditions. It does not have to meet both. It can be tricky.

3. True or False: TCPDump will resolve IPs to hostnames by default. True


Hack The Box - Academy
Intro to Network Traffic Analysis
Student Potrait Photos, Downloa
academy.hackthebox.com/module/81/section/785#questionsDiv
Waiting to start...
Cheat Sheet
Submit
Hint
19:47

Questions

Answer the question(s) below to complete this Section and earn cubes!

+0 What filter will allow me to see traffic coming from or destined to the host with an ip of 10.10.20.1?

host 10.10.20.1

+0 What filter will allow me to capture based on either of two options?

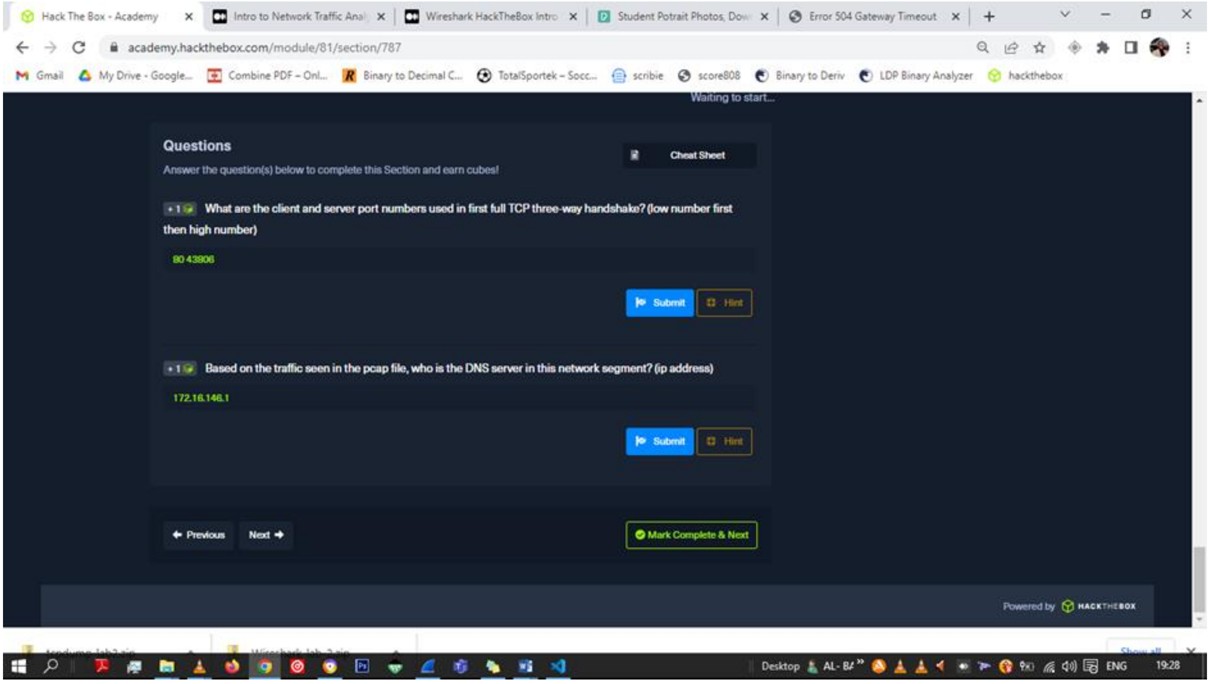
or

+0 True or False: TCPDump will resolve IPs to hostnames by default.

true

Interrogating Network Traffic with Capture and Display Filters

This lab aims to provide some exposure to interrogating network traffic and give everyone some valuable practice implementing packet filters. We will be utilizing filters like host, port, protocol, and more to change our view while digging through a .PCAP file.


Hack The Box - Academy
Intro to Network Traffic Anal
Wireshark HackTheBox Intro
Student Potrait Photos, Dow
Error 504 Gateway Timeout
academy.hackthebox.com/module/81/section/787
Waiting to start...
Cheat Sheet
Submit
Hint
Previous
Next
Mark Complete & Next
Powered by HACKTHEBOX
19:28

Questions

Answer the question(s) below to complete this Section and earn cubes!

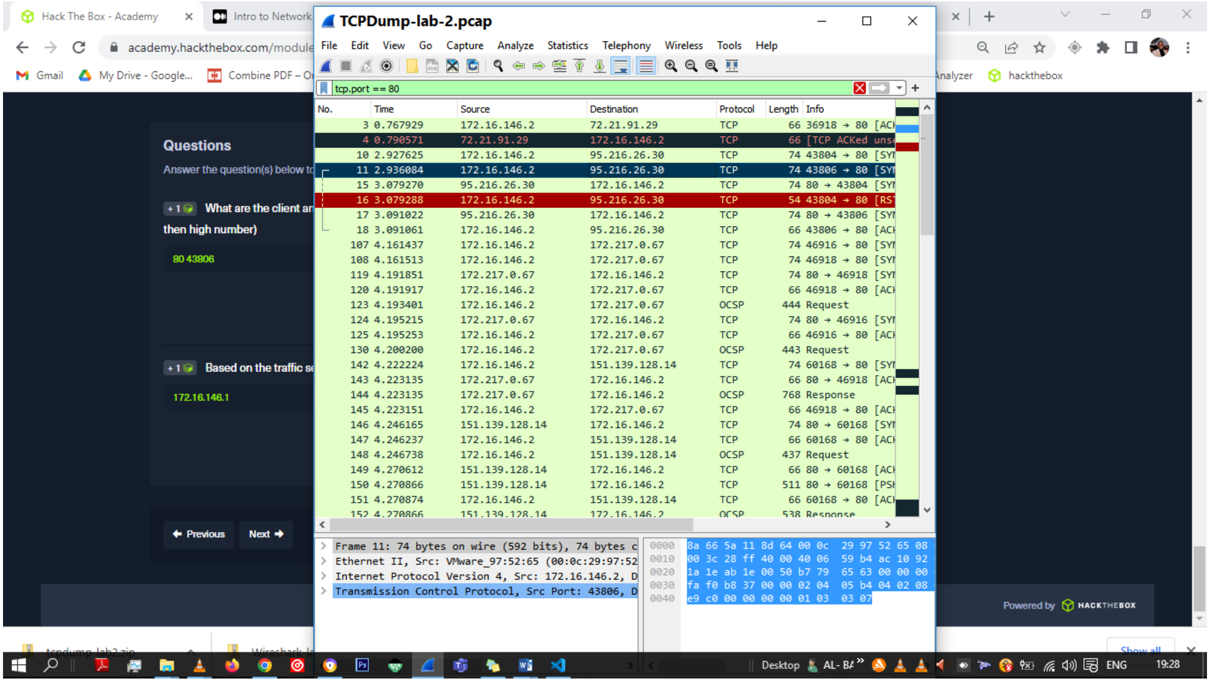
+1 What are the client and server port numbers used in first full TCP three-way handshake? (low number first then high number)

80 43806

+1 Based on the traffic seen in the pcap file, who is the DNS server in this network segment? (ip address)

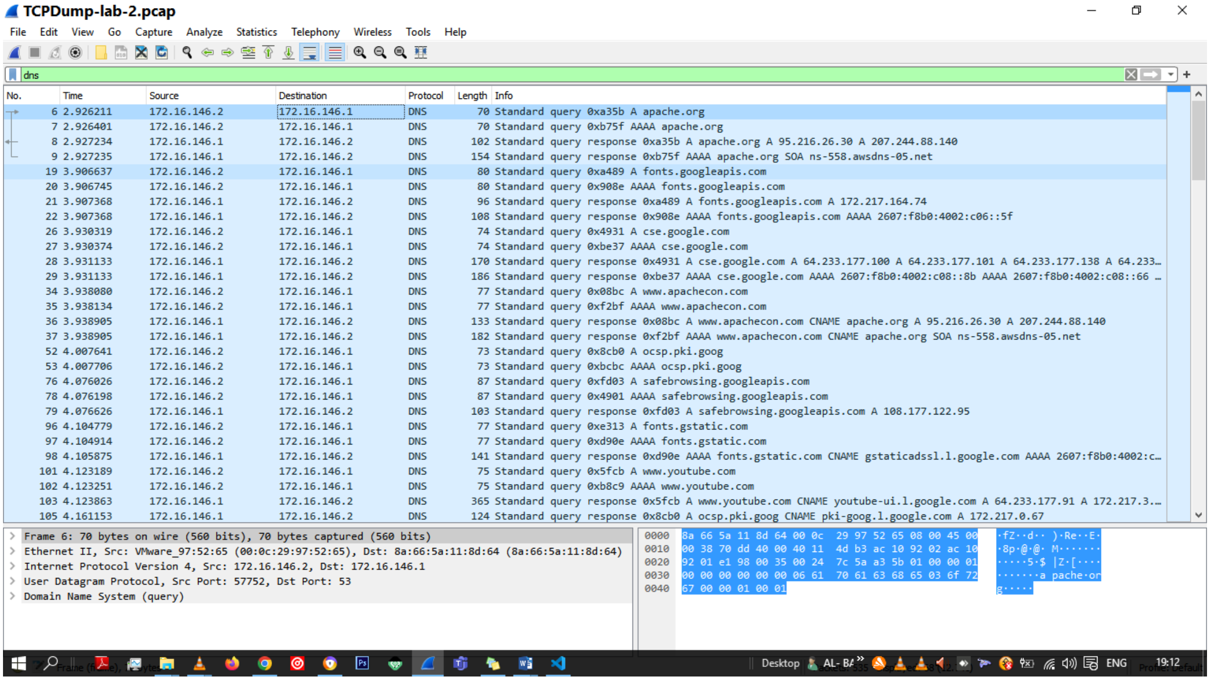
172.16.146.1

1. **What are the client and server port numbers used in first full TCP three-way handshake? (low number first then high number)**80 43806



```
TCPDump-lab-2.pcap
File  Edit  View  Go  Capture  Analyze  Statistics  Telephony  Wireless  Tools  Help
tcp.port == 80

No.  Time       Source          Destination     Protocol  Length  Info
  3 0.767929    172.16.146.2    72.21.91.29     TCP        66 36918 → 80 [ACK
  4 0.790571    72.21.91.29     172.16.146.2    TCP        66 [TCP ACKed unse
 10 2.927625    172.16.146.2    95.216.26.30    TCP        74 43804 → 80 [SYN
 11 2.936084    172.16.146.2    95.216.26.30    TCP        74 43806 → 80 [SYN
 15 3.079270    95.216.26.30    172.16.146.2    TCP        74 80 → 43804 [SYN
 16 3.079288    172.16.146.2    95.216.26.30    TCP        54 43804 → 80 [RST
 17 3.091022    95.216.26.30    172.16.146.2    TCP        74 80 → 43806 [SYN
 18 3.091061    172.16.146.2    95.216.26.30    TCP        66 43806 → 80 [ACK
107 4.161437    172.16.146.2    172.217.0.67    TCP        74 46916 → 80 [SYN
108 4.161513    172.16.146.2    172.217.0.67    TCP        74 46918 → 80 [SYN
119 4.191851    172.217.0.67    172.16.146.2    TCP        74 80 → 46918 [SYN
120 4.191917    172.16.146.2    172.217.0.67    TCP        66 46918 → 80 [ACK
123 4.193401    172.16.146.2    172.217.0.67    OCSP      444 Request
124 4.195215    172.217.0.67    172.16.146.2    TCP        74 80 → 46916 [SYN
125 4.195253    172.16.146.2    172.217.0.67    TCP        66 46916 → 80 [ACK
130 4.200200    172.16.146.2    172.217.0.67    OCSP      443 Request
142 4.222224    172.16.146.2    151.139.128.14  TCP        74 60168 → 80 [SYN
143 4.223135    172.217.0.67    172.16.146.2    TCP        66 80 → 46918 [ACK
144 4.223135    172.217.0.67    172.16.146.2    OCSP      768 Response
145 4.223151    172.16.146.2    172.217.0.67    TCP        66 46918 → 80 [ACK
146 4.246165    151.139.128.14  172.16.146.2    TCP        74 80 → 60168 [SYN
147 4.246237    172.16.146.2    151.139.128.14  TCP        66 60168 → 80 [ACK
148 4.246738    172.16.146.2    151.139.128.14  OCSP      437 Request
149 4.270612    151.139.128.14  172.16.146.2    TCP        66 80 → 60168 [ACK
150 4.270866    151.139.128.14  172.16.146.2    TCP       511 80 → 60168 [PSH
151 4.270874    172.16.146.2    151.139.128.14  TCP        66 60168 → 80 [ACK
152 4.270866    151.139.128.14  172.16.146.2    OCSP      538 Response

> Frame 11: 74 bytes on wire (592 bits), 74 bytes c
> Ethernet II, Src: VMware_97:52:65 (00:0c:29:97:52
> Internet Protocol Version 4, Src: 172.16.146.2, D
> Transmission Control Protocol, Src Port: 43806, D

0000  8a 66 5a 11 8d 64 00 0c  29 97 52 65 08
0010  00 3c 28 ff 40 00 40 06  59 b4 ac 10 92
0020  1a 1e ab 1e 00 50 b7 79  65 63 00 00 00
0030  fa f0 b8 37 00 00 02 04  05 b4 04 02 08
0040  e9 c0 00 00 00 00 01 03  03 07
```

2. **Based on the traffic seen in the pcap file, who is the DNS server in this network segment? (ip address)172.16.146.1**

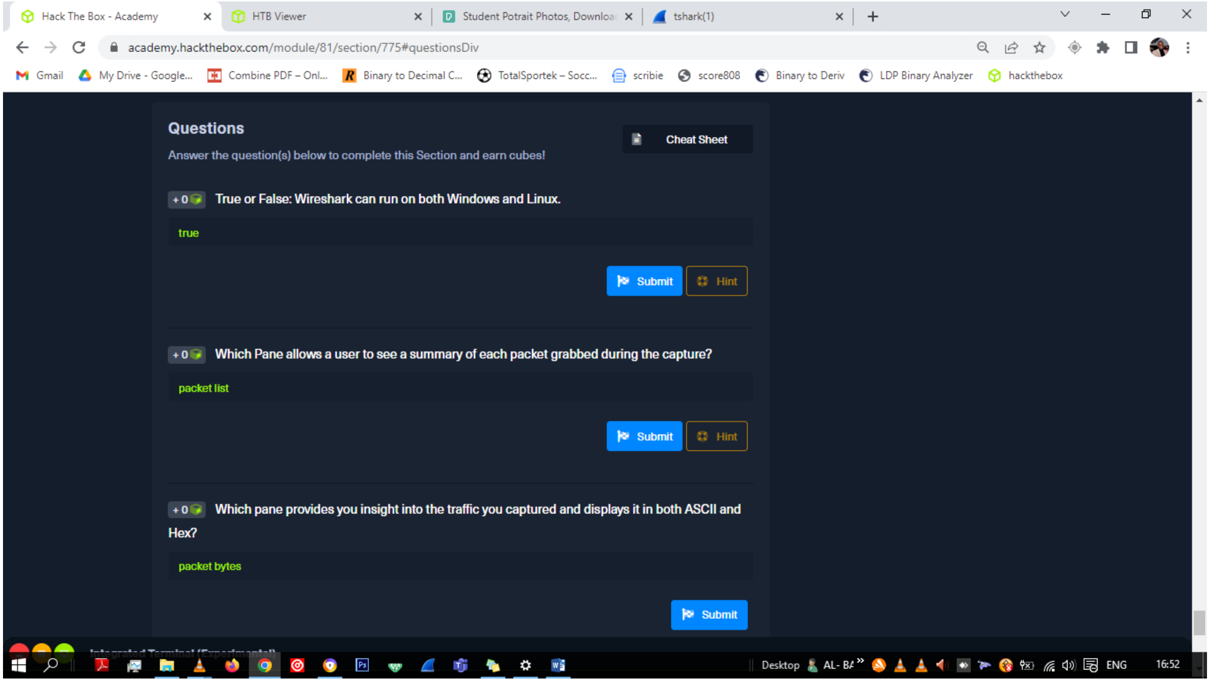


```
TCPDump-lab-2.pcap
File  Edit  View  Go  Capture  Analyze  Statistics  Telephony  Wireless  Tools  Help
dns

No.  Time       Source        Destination   Protocol  Length  Info
  6 2.926211    172.16.146.2  172.16.146.1  DNS        70 Standard query 0xa35b A apache.org
  7 2.926401    172.16.146.2  172.16.146.1  DNS        70 Standard query 0xb75f AAAA apache.org
  8 2.927234    172.16.146.1  172.16.146.2  DNS       102 Standard query response 0xa35b A apache.org A 95.216.26.30 A 207.244.88.140
  9 2.927235    172.16.146.1  172.16.146.2  DNS       154 Standard query response 0xb75f AAAA apache.org SOA ns-558.awsdns-05.net
 19 3.906637    172.16.146.2  172.16.146.1  DNS        80 Standard query 0xa489 A fonts.googleapis.com
 20 3.906745    172.16.146.2  172.16.146.1  DNS        80 Standard query 0x908e AAAA fonts.googleapis.com
 21 3.907368    172.16.146.1  172.16.146.2  DNS        96 Standard query response 0xa489 A fonts.googleapis.com A 172.217.164.74
 22 3.907368    172.16.146.1  172.16.146.2  DNS       108 Standard query response 0x908e AAAA fonts.googleapis.com AAAA 2607:f8b0:4002:c06::5f
 26 3.930319    172.16.146.2  172.16.146.1  DNS        74 Standard query 0x4931 A cse.google.com
 27 3.930374    172.16.146.2  172.16.146.1  DNS        74 Standard query 0xbe37 AAAA cse.google.com
 28 3.931133    172.16.146.1  172.16.146.2  DNS       170 Standard query response 0x4931 A cse.google.com A 64.233.177.100 A 64.233.177.101 A 64.233.177.138 A 64.233…
 29 3.931133    172.16.146.1  172.16.146.2  DNS       186 Standard query response 0xbe37 AAAA cse.google.com AAAA 2607:f8b0:4002:c08::8b AAAA 2607:f8b0:4002:c08::66 …
 34 3.938080    172.16.146.2  172.16.146.1  DNS        77 Standard query 0x08bc A www.apachecon.com
 35 3.938134    172.16.146.2  172.16.146.1  DNS        77 Standard query 0xf2bf AAAA www.apachecon.com
 36 3.938905    172.16.146.1  172.16.146.2  DNS       133 Standard query response 0x08bc A www.apachecon.com CNAME apache.org A 95.216.26.30 A 207.244.88.140
 37 3.938905    172.16.146.1  172.16.146.2  DNS       182 Standard query response 0xf2bf AAAA www.apachecon.com CNAME apache.org SOA ns-558.awsdns-05.net
 52 4.007641    172.16.146.2  172.16.146.1  DNS        73 Standard query 0x8cb0 A ocsp.pki.goog
 53 4.007706    172.16.146.2  172.16.146.1  DNS        73 Standard query 0xbcbc AAAA ocsp.pki.goog
 76 4.076026    172.16.146.2  172.16.146.1  DNS        87 Standard query 0xfd03 A safebrowsing.googleapis.com
 78 4.076198    172.16.146.2  172.16.146.1  DNS        87 Standard query 0x4901 AAAA safebrowsing.googleapis.com
 79 4.076626    172.16.146.1  172.16.146.2  DNS       103 Standard query response 0xfd03 A safebrowsing.googleapis.com A 108.177.122.95
 96 4.104779    172.16.146.2  172.16.146.1  DNS        77 Standard query 0xe313 A fonts.gstatic.com
 97 4.104914    172.16.146.2  172.16.146.1  DNS        77 Standard query 0xd90e AAAA fonts.gstatic.com
 98 4.105875    172.16.146.1  172.16.146.2  DNS       141 Standard query response 0xd90e AAAA fonts.gstatic.com CNAME gstaticadssl.l.google.com AAAA 2607:f8b0:4002:c…
101 4.123189    172.16.146.2  172.16.146.1  DNS        75 Standard query 0x5fcb A www.youtube.com
102 4.123251    172.16.146.2  172.16.146.1  DNS        75 Standard query 0xb8c9 AAAA www.youtube.com
103 4.123863    172.16.146.1  172.16.146.2  DNS       365 Standard query response 0x5fcb A www.youtube.com CNAME youtube-ui.l.google.com A 64.233.177.91 A 172.217.3.…
105 4.161153    172.16.146.1  172.16.146.2  DNS       124 Standard query response 0x8cb0 A ocsp.pki.goog CNAME pki-goog.l.google.com A 172.217.0.67

> Frame 6: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
> Ethernet II, Src: VMware_97:52:65 (00:0c:29:97:52:65), Dst: 8a:66:5a:11:8d:64 (8a:66:5a:11:8d:64)
> Internet Protocol Version 4, Src: 172.16.146.2, Dst: 172.16.146.1
> User Datagram Protocol, Src Port: 57752, Dst Port: 53
> Domain Name System (query)

0000  8a 66 5a 11 8d 64 00 0c  29 97 52 65 08 00 45 00   ·fZ··d·· )·Re··E·
0010  00 38 70 dd 40 00 40 11  4d b3 ac 10 92 02 ac 10   ·8p·@·@· M·······
0020  92 01 e1 98 00 35 00 24  7c 5a a3 5b 01 00 00 01   ·····5·$ |Z·[····
0030  00 00 00 00 00 00 06 61  70 61 63 68 65 03 6f 72   ·······a pache·or
0040  67 00 00 01 00 01                                  g·····
```

Analysis with Wireshark

1. **True or False: Wireshark can run on both Windows and Linux.**

Wireshark is a free and open-source network traffic analyser much like tcpdump but with a graphical interface. Wireshark is multi-platform and capable of capturing live data off many different interface types (to include Wi-Fi, USB, and Bluetooth) and saving the traffic to several different formats.



Questions

Answer the question(s) below to complete this Section and earn cubes!

+0 True or False: Wireshark can run on both Windows and Linux.

true

+0 Which Pane allows a user to see a summary of each packet grabbed during the capture?

packet list

+0 Which pane provides you insight into the traffic you captured and displays it in both ASCII and Hex?

packet bytes

2. **Which Pane allows a user to see a summary of each packet grabbed during the capture?** Packet list :orange
3. **Which pane provides you insight into the traffic you captured and displays it in both ASCII and Hex?** Packet byte : green

Packet list: orange in this window, we see a summary line of each packet that includes the fields listed below by default. We can add or remove columns to change what information is presented.

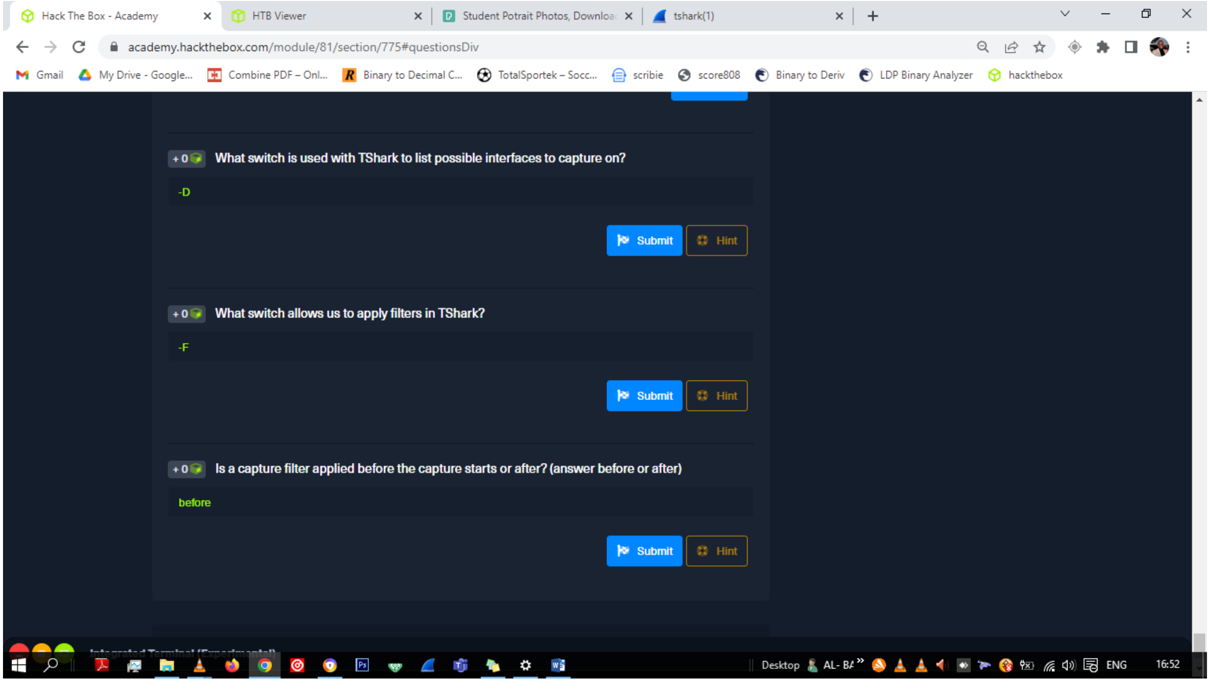
The Packet Details: **blue** window allows us to drill down into the packet to inspect the protocols with greater detail. It will break it down into chunks that we would expect following the typical OSI Model reference

The Packet Bytes: green window allows us to look at the packet contents in ASCII or hex output. As we select a field from the windows above, it will be highlighted in the Packet Bytes window and show us where that bit or byte falls within the overall packet**.**

4. **What switch is used with TShark to list possible interfaces to capture on?-D**
5. **What switch allows us to apply filters in TShark? –F**

D Will display any interfaces available to capture from and then exit ou**t.**
L Will list the Link-layer mediums you can capture from and then exit out. (Ethernet as an example)
i choose an interface to capture from. (-i eth0)
f packet filter in libpcap syntax. Used during capture.

6. **Is a capture filter applied before the capture starts or after? (Answer before or after)** Capture Filters- are entered before the capture is started.



+0 What switch is used with TShark to list possible interfaces to capture on?

-D

+0 What switch allows us to apply filters in TShark?

-F

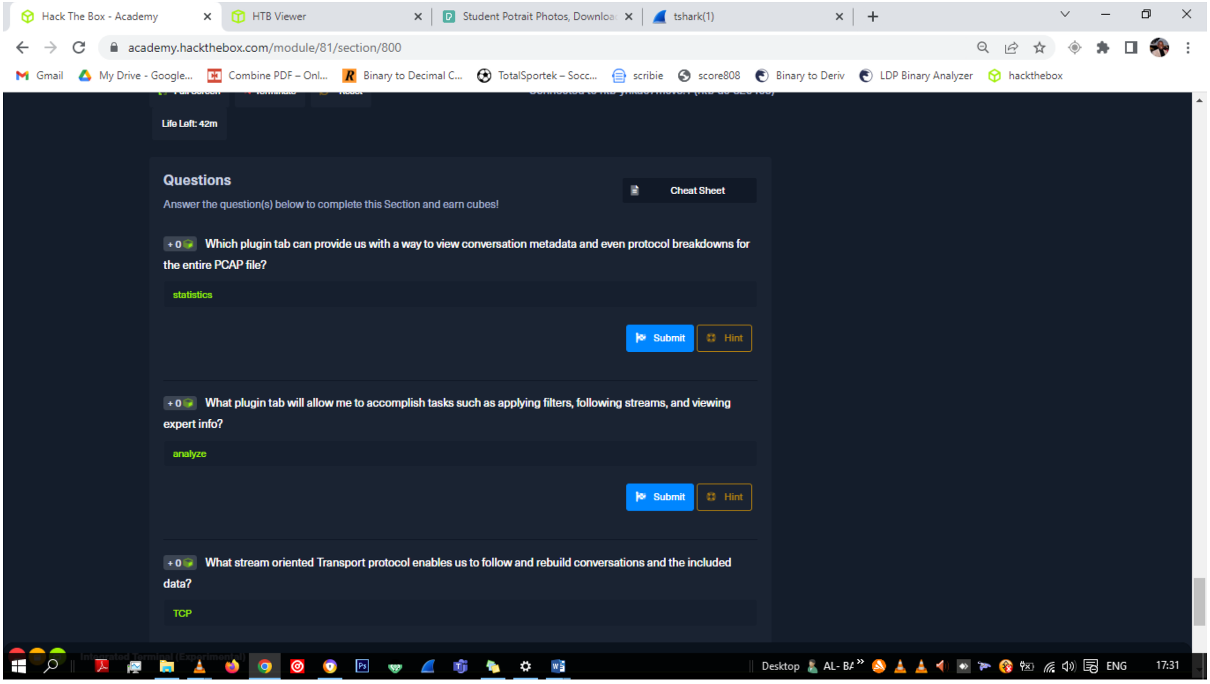
+0 Is a capture filter applied before the capture starts or after? (answer before or after)

before

Wireshark Advanced Usage

1. **Which plugin tab can provide us with a way to view conversation metadata and even protocol breakdowns for the entire PCAP file?** Statistics tab

2. **What plugin tab will allow me to accomplish tasks such as applying filters, following streams, and viewing expert info?** Analyse tab



Questions

Answer the question(s) below to complete this Section and earn cubes!

+0 Which plugin tab can provide us with a way to view conversation metadata and even protocol breakdowns for the entire PCAP file?

statistics

+0 What plugin tab will allow me to accomplish tasks such as applying filters, following streams, and viewing expert info?

analyze

+0 What stream oriented Transport protocol enables us to follow and rebuild conversations and the included data?

TCP

Analyze tab

We can utilize plugins that allow us to do things such as following TCP streams, filter on conversation types, prepare new packet filters and examine the expert info Wireshark generates about the traffic.

Statistics tab

The Statistics and Analyze tabs can provide us with great insight into the data we are examining.

1. **What stream oriented Transport protocol enables us to follow and rebuild conversations and the included data?** TCP

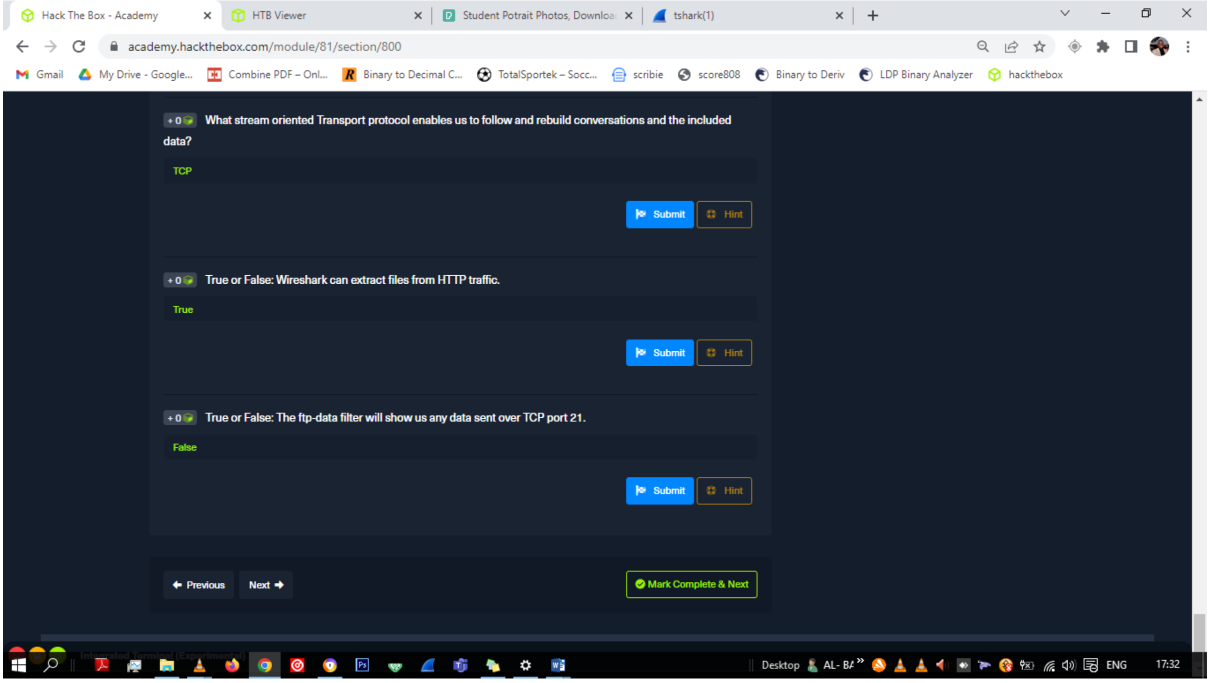
Wireshark can stitch TCP packets back together to recreate the entire stream in a readable format. This ability also allows us to pull data (images, files, etc.) out of the capture. This works for almost any protocol that utilizes TCP as a transport mechanism.

2. **True or False: Wireshark can extract files from HTTP traffic.** True

Wireshark can recover many different types of data from streams. It requires you to have captured the entire conversation.

1. **True or False: The ftp-data filter will show us any data sent over TCP port 21.**

   False



+0 What stream oriented Transport protocol enables us to follow and rebuild conversations and the included data?

TCP

Submit Hint

+0 True or False: Wireshark can extract files from HTTP traffic.

True

Submit Hint

+0 True or False: The ftp-data filter will show us any data sent over TCP port 21.

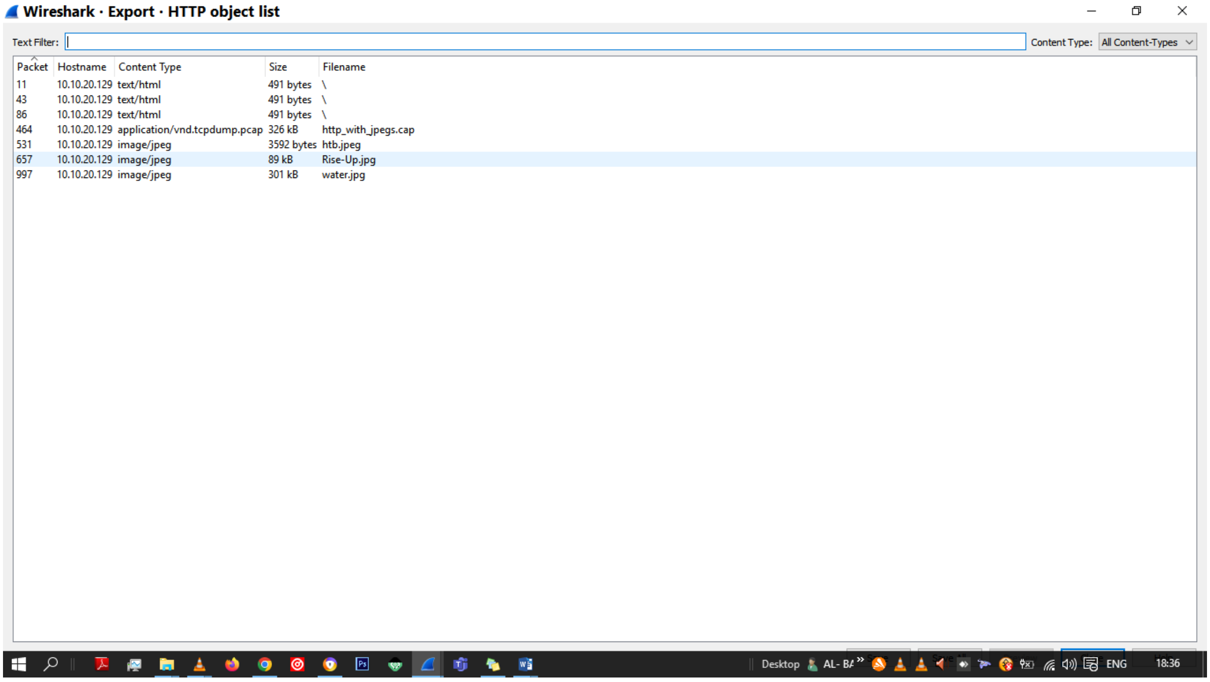
False

Submit Hint

Packet Inception, Dissecting Network Traffic with Wireshark

The purpose of this lab is to provide experience with dissecting traffic in Wireshark. We will have the chance to pull objects out of previously captured network traffic along with pulling data from live traffic.

1. **What was the filename of the image that contained a certain Transformer Leader? (name.filetype)** rise-up.jpg
2. **Which employee is suspected of performing potentially malicious actions in the live environment?** Bob

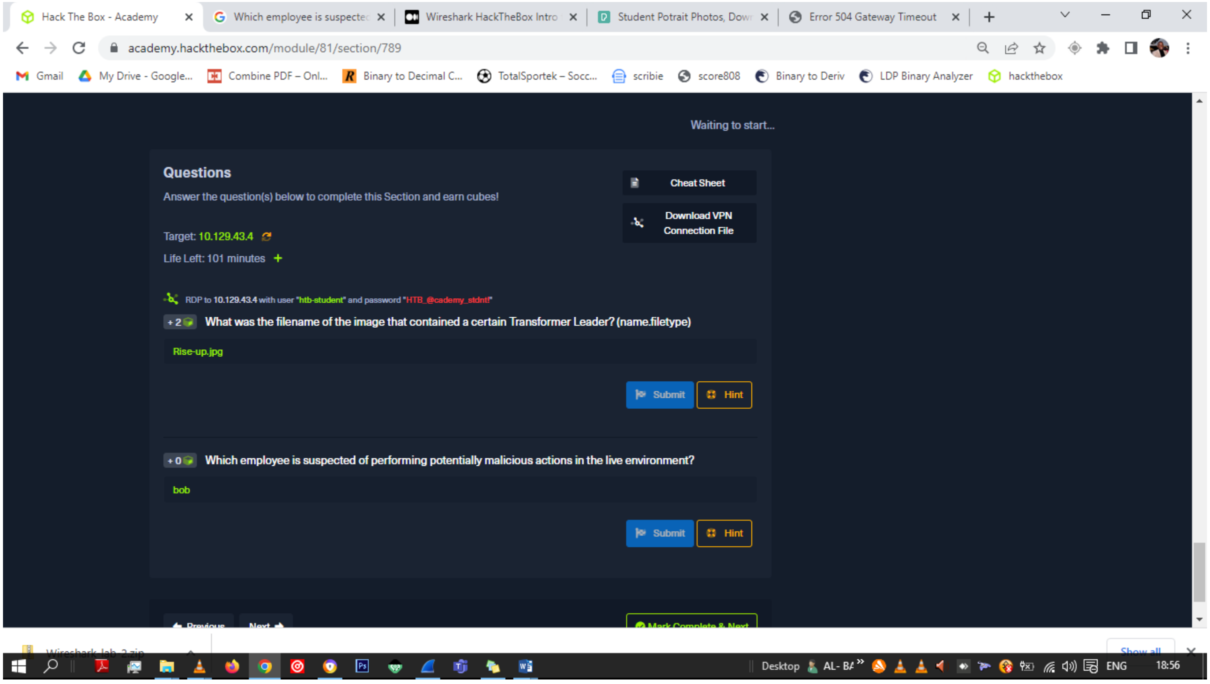


Wireshark · Export · HTTP object list

Text Filter:

Content Type: All Content-Types

Packet	Hostname	Content Type	Size	Filename
11	10.10.20.129	text/html	491 bytes	\
43	10.10.20.129	text/html	491 bytes	\
86	10.10.20.129	text/html	491 bytes	\
464	10.10.20.129	application/vnd.tcpdump.pcap	326 kB	http_with_jpegs.cap
531	10.10.20.129	image/jpeg	3592 bytes	htb.jpeg
657	10.10.20.129	image/jpeg	89 kB	Rise-Up.jpg
997	10.10.20.129	image/jpeg	301 kB	water.jpg



academy.hackthebox.com/module/81/section/789

Waiting to start...

Questions

Answer the question(s) below to complete this Section and earn cubes!

Target: 10.129.43.4

Life Left: 101 minutes

RDP to 10.129.43.4 with user "htb-student" and password "HTB_@cademy_stdnt!"

+2 What was the filename of the image that contained a certain Transformer Leader? (name.filetype)

Rise-up.jpg

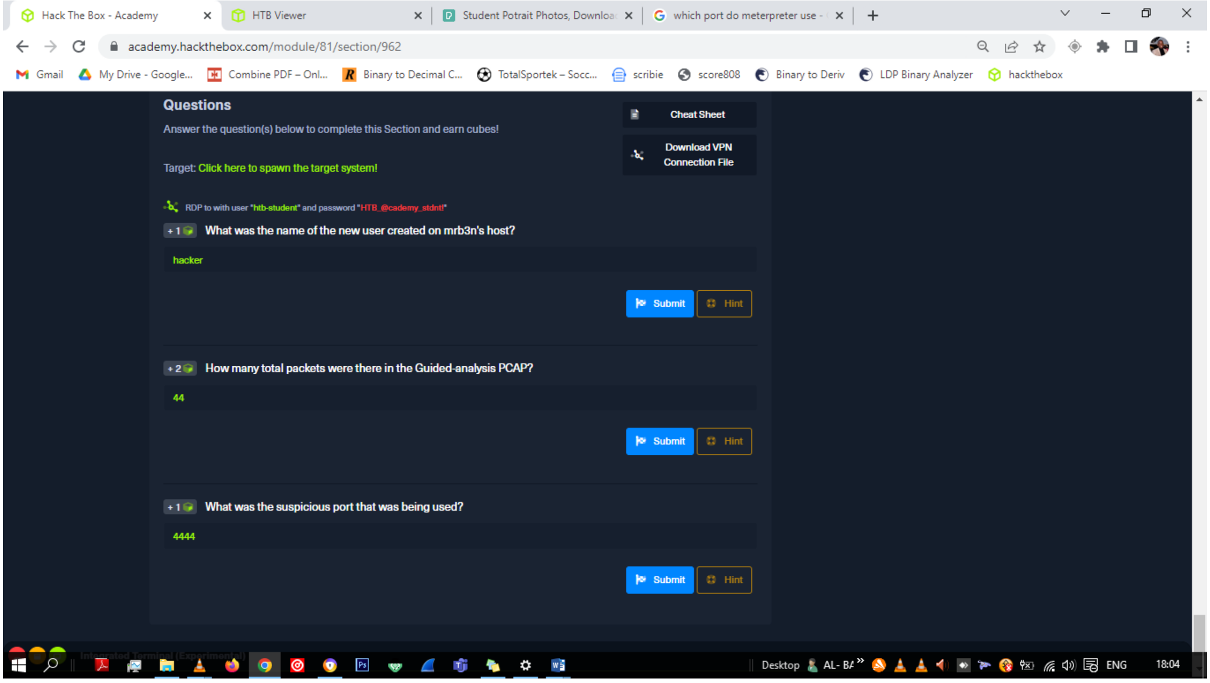
+0 Which employee is suspected of performing potentially malicious actions in the live environment?

bob

Guided Lab: Traffic Analysis Workflow

1. **What was the name of the new user created on mrb3n's host? hacker**
2. **How many total packets were there in the Guided-analysis PCAP? 44**
3. **What was the suspicious port that was being used?**

   Port 4444



Questions

Answer the question(s) below to complete this Section and earn cubes!

Target: Click here to spawn the target system!

RDP to with user "htb-student" and password "HTB_@cademy_stdnt!"

+1 What was the name of the new user created on mrb3n's host?

hacker

Submit Hint

+2 How many total packets were there in the Guided-analysis PCAP?

44

Submit Hint

+1 What was the suspicious port that was being used?

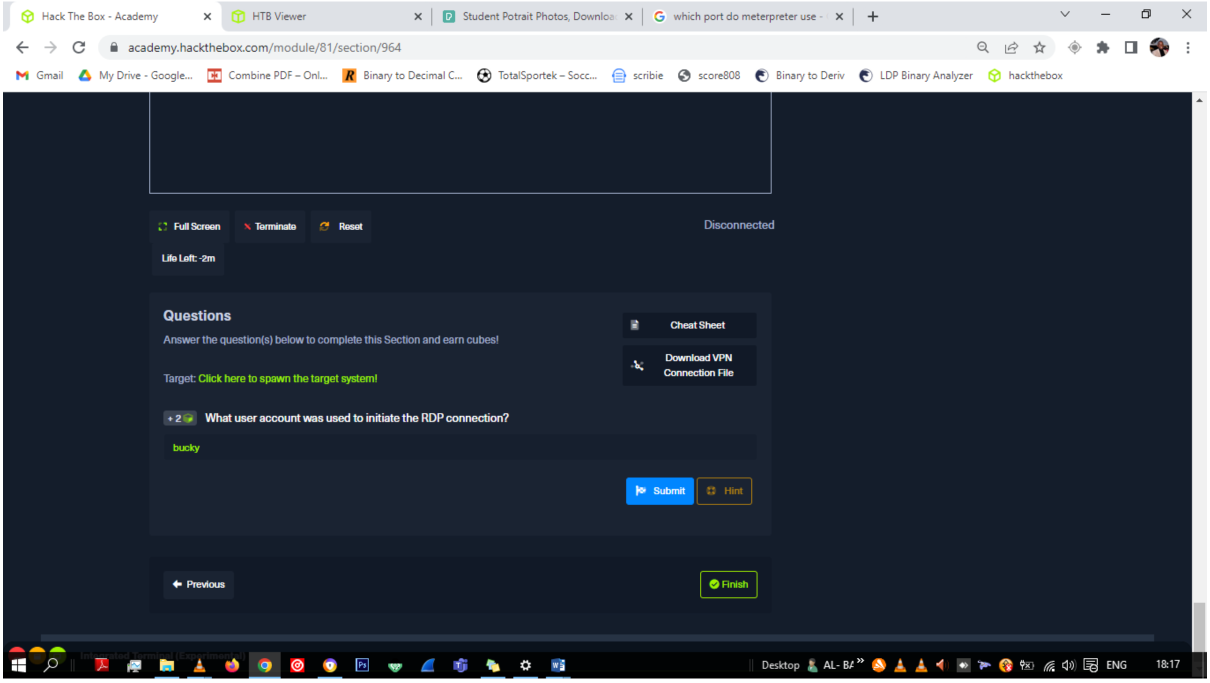
4444

Submit Hint

Decrypting RDP connections

1. **What user account was used to initiate the RDP connection?**

   Bucky



Hack The Box - Academy
HTB Viewer
Student Potrait Photos, Downloa
which port do meterpreter use
academy.hackthebox.com/module/81/section/964
Full Screen
Terminate
Reset
Disconnected
Life Left: -2m
Cheat Sheet
Download VPN Connection File
Submit
Hint
Previous
Finish

Questions

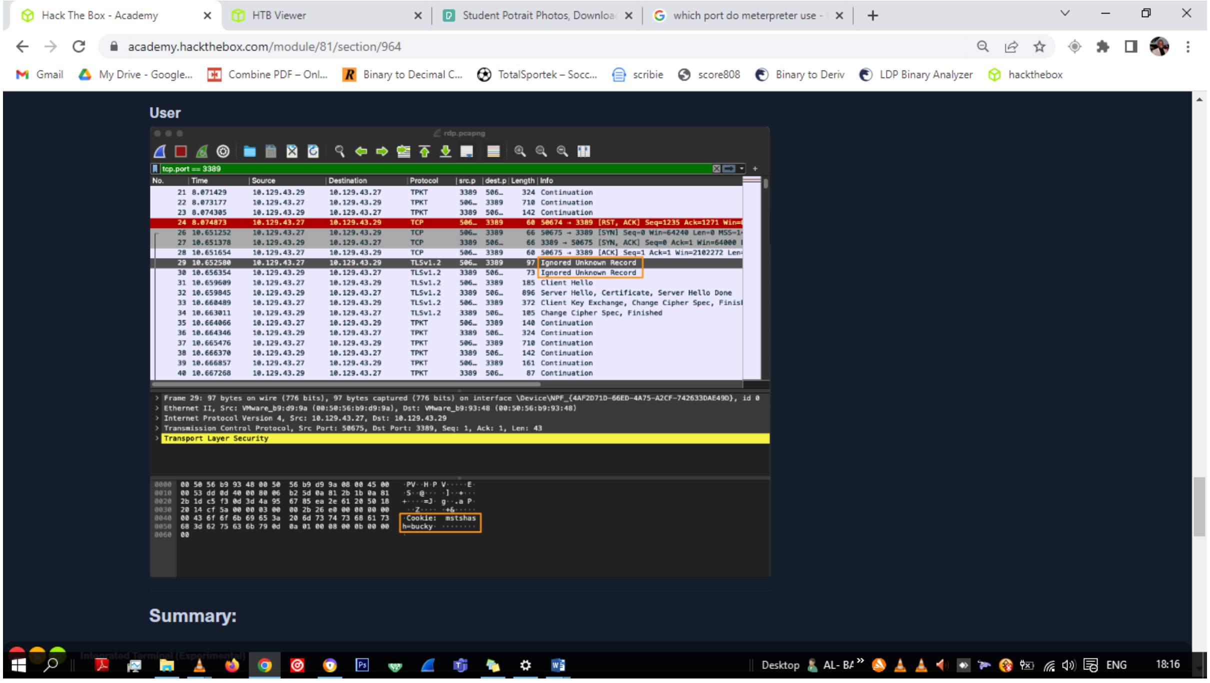
Answer the question(s) below to complete this Section and earn cubes!

Target: Click here to spawn the target system!

+2 What user account was used to initiate the RDP connection?

bucky

When filter on tcp.port == 3389, we can see a record labeled Ignored Unknown Record. If we examine the ASCII, it will show us a username.



Hack The Box - Academy
HTB Viewer
Student Potrait Photos, Downloa
which port do meterpreter use
academy.hackthebox.com/module/81/section/964
User
rdp.pcapng
Summary:

```
tcp.port == 3389
No.   Time       Source        Destination   Protocol  src.p dest.p Length Info
   21 8.071429   10.129.43.29  10.129.43.27  TPKT      3389  506…    324 Continuation
   22 8.073177   10.129.43.27  10.129.43.29  TPKT      506…  3389    710 Continuation
   23 8.074305   10.129.43.29  10.129.43.27  TPKT      3389  506…    142 Continuation
   24 8.074873   10.129.43.27  10.129.43.29  TCP       506…  3389     60 50674 → 3389 [RST, ACK] Seq=1235 Ack=1271 Win=
   26 10.651252  10.129.43.27  10.129.43.29  TCP       506…  3389     66 50675 → 3389 [SYN] Seq=0 Win=64240 Len=0 MSS=1
   27 10.651378  10.129.43.29  10.129.43.27  TCP       3389  506…     66 3389 → 50675 [SYN, ACK] Seq=0 Ack=1 Win=64000
   28 10.651654  10.129.43.27  10.129.43.29  TCP       506…  3389     60 50675 → 3389 [ACK] Seq=1 Ack=1 Win=2102272 Len
   29 10.652580  10.129.43.27  10.129.43.29  TLSv1.2   506…  3389     97 Ignored Unknown Record
   30 10.656354  10.129.43.29  10.129.43.27  TLSv1.2   3389  506…     73 Ignored Unknown Record
   31 10.659609  10.129.43.27  10.129.43.29  TLSv1.2   506…  3389    185 Client Hello
   32 10.659845  10.129.43.29  10.129.43.27  TLSv1.2   3389  506…    896 Server Hello, Certificate, Server Hello Done
   33 10.660489  10.129.43.27  10.129.43.29  TLSv1.2   506…  3389    372 Client Key Exchange, Change Cipher Spec, Finis
   34 10.663011  10.129.43.29  10.129.43.27  TLSv1.2   3389  506…    105 Change Cipher Spec, Finished
   35 10.664066  10.129.43.27  10.129.43.29  TPKT      506…  3389    140 Continuation
   36 10.664346  10.129.43.29  10.129.43.27  TPKT      3389  506…    324 Continuation
   37 10.665476  10.129.43.27  10.129.43.29  TPKT      506…  3389    710 Continuation
   38 10.666370  10.129.43.29  10.129.43.27  TPKT      3389  506…    142 Continuation
   39 10.666857  10.129.43.27  10.129.43.29  TPKT      506…  3389    161 Continuation
   40 10.667268  10.129.43.29  10.129.43.27  TPKT      3389  506…     87 Continuation

> Frame 29: 97 bytes on wire (776 bits), 97 bytes captured (776 bits) on interface \Device\NPF_{4AF2D71D-66ED-4A75-A2CF-742633DAE49D}, id 0
> Ethernet II, Src: VMware_b9:d9:9a (00:50:56:b9:d9:9a), Dst: VMware_b9:93:48 (00:50:56:b9:93:48)
> Internet Protocol Version 4, Src: 10.129.43.27, Dst: 10.129.43.29
> Transmission Control Protocol, Src Port: 50675, Dst Port: 3389, Seq: 1, Ack: 1, Len: 43
> Transport Layer Security

0000  00 50 56 b9 93 48 00 50  56 b9 d9 9a 08 00 45 00   ·PV··H·P V·····E·
0010  00 53 dd 0d 40 00 80 06  b2 5d 0a 81 2b 1b 0a 81   ·S··@··· ·]··+···
0020  2b 1d c5 f3 0d 3d 4a 95  67 85 ea 2e 61 20 50 18   +····=J· g··.a P·
0030  20 14 cf 5a 00 00 03 00  00 2b 26 e0 00 00 00 00    ··Z···· ·+&·····
0040  00 43 6f 6f 6b 69 65 3a  20 6d 73 74 73 68 61 73   ·Cookie:  mstshas
0050  68 3d 62 75 63 6b 79 0d  0a 01 00 08 00 0b 00 00   h=bucky· ········
0060  00                                                 ·
```

Link to module completion

https://academy.hackthebox.com/achievement/820405/81

Conclusion

The introduction to network analysis module has introduced me in deeper analysis of network traffic using tools such as Wireshark and tcpdumps. These tools assist in sniffing through the network to view activities and trace packets paths. Is also help in monitoring the network for malicious activities. The module has also introduced me to the Linux command line and switches used in order to perform compact network analysis.