COURSE:Cloud and Network security.

STUDENTS NAME:Alex mwangi.

STUDENTS NUMBER:CS-CNS-25113.

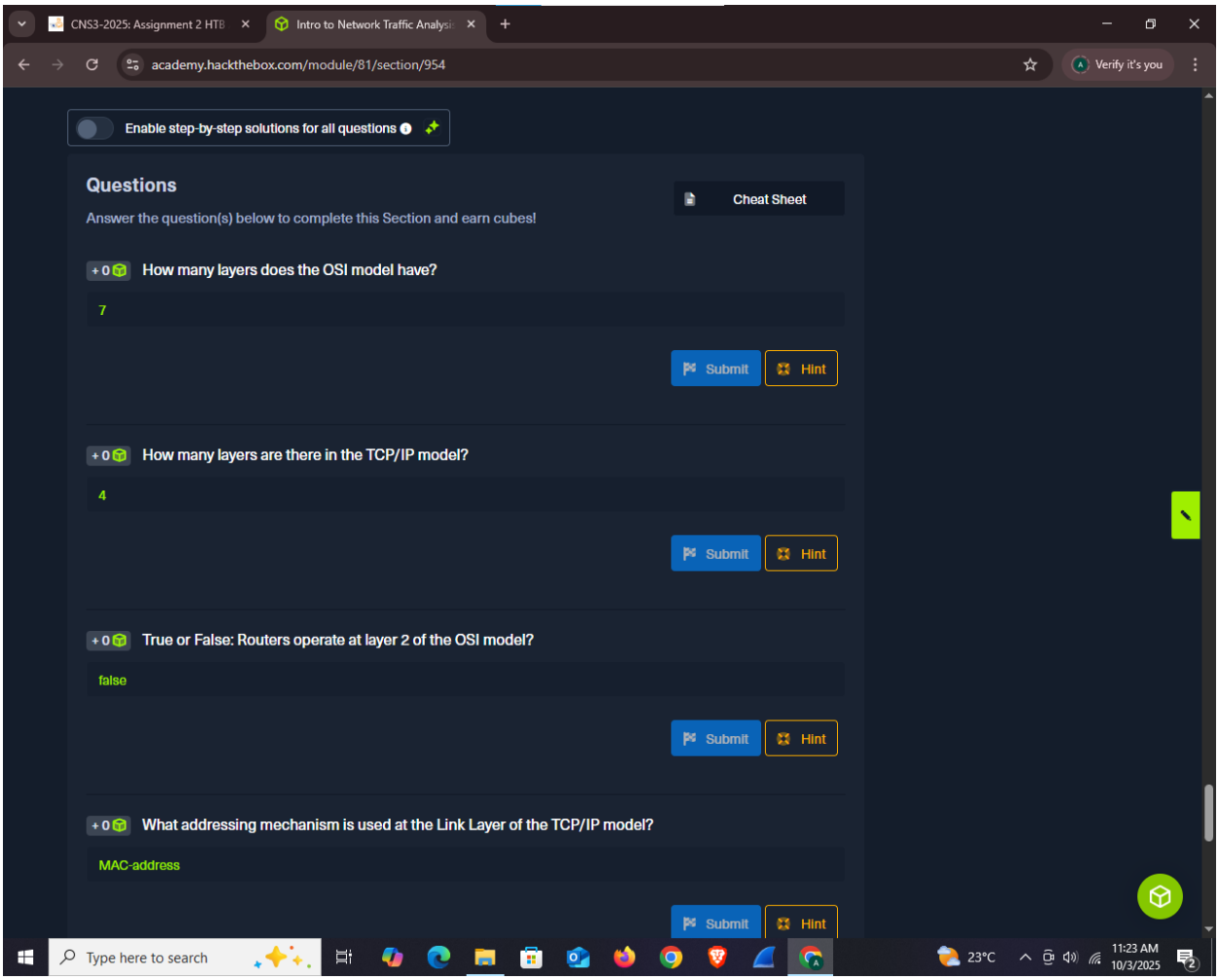
SUBMISSION DATE:

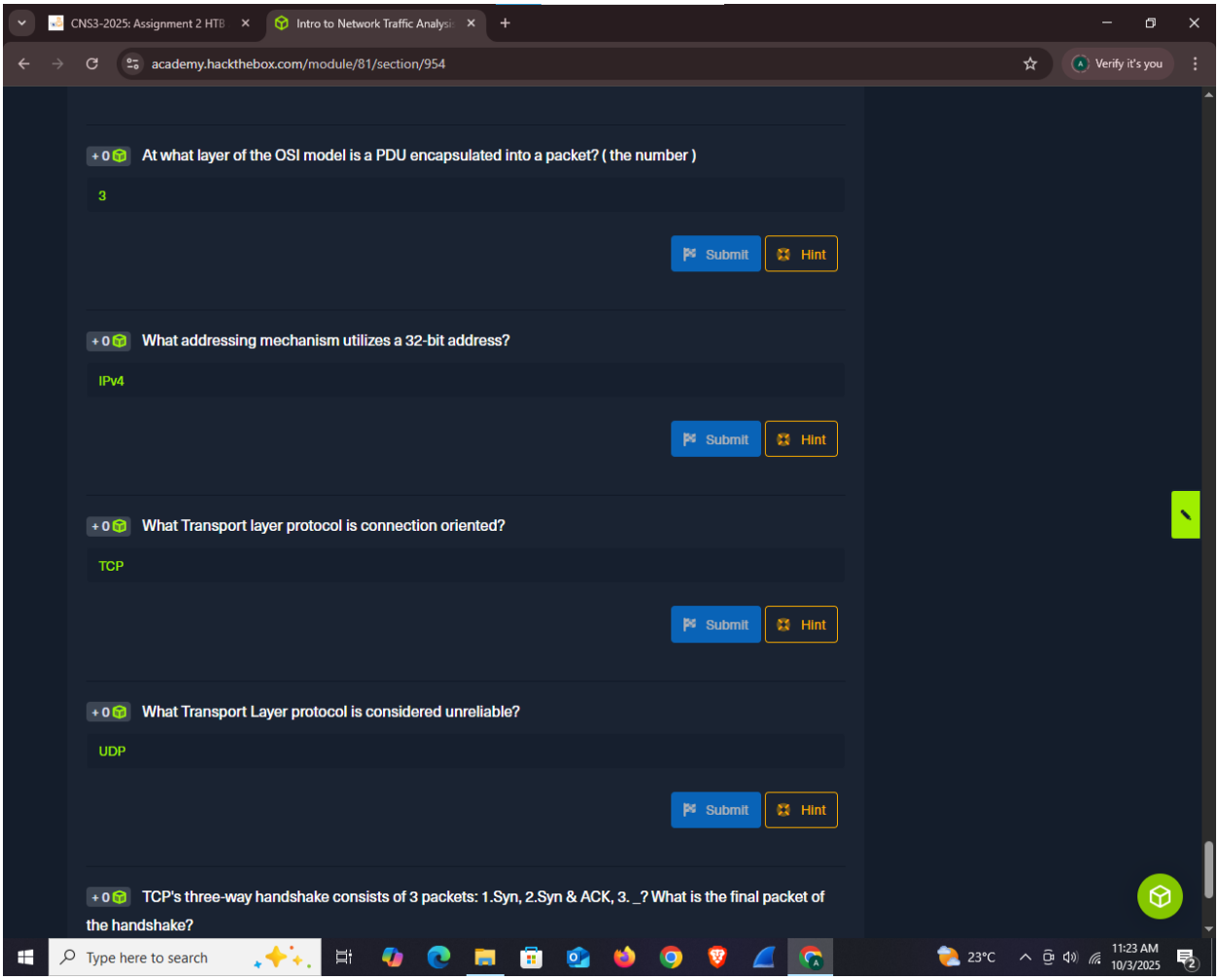
WEEK 2 ASSIGNMENT 2 , HTB ACADEMY:INTRODUCTION TO NETWORK TRAFFIC ANALYSIS.

INTRODUCTION.

In this activity we were venturing the HTB academy where it was network traffic analysis course .In this course we were to go through network primerlayer 1-7 ,analysis process and how it goes by tcpdump fundamentals , capturing with tcp dump ,packet filtering ,interrogating network traffic , analysis with wireshark,advanced usage of wireshark and decrypting RDP connections.

Quiz 1. Networking primer layer layer 1-4.

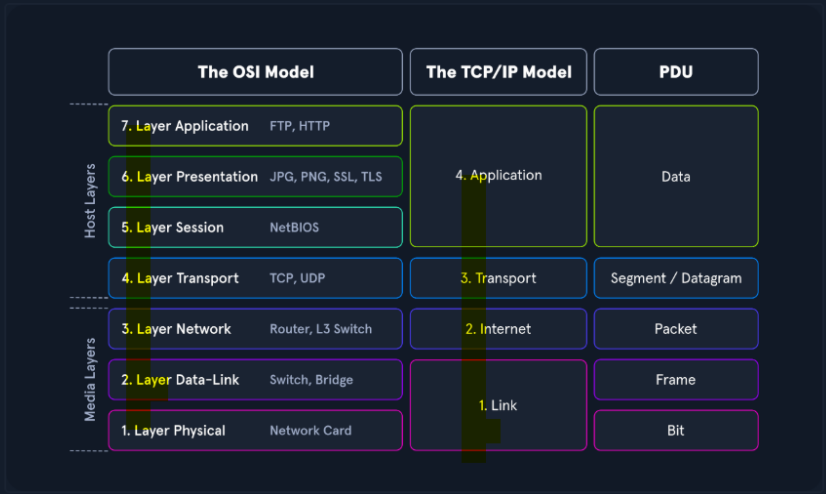



CNS3-2025: Assignment 2 HTB
Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/954
Verify it's you
+0 At what layer of the OSI model is a PDU encapsulated into a packet? (the number)
3
Submit
Hint
+0 What addressing mechanism utilizes a 32-bit address?
IPv4
Submit
Hint
+0 What Transport layer protocol is connection oriented?
TCP
Submit
Hint
+0 What Transport Layer protocol is considered unreliable?
UDP
Submit
Hint
+0 TCP's three-way handshake consists of 3 packets: 1.Syn, 2.Syn & ACK, 3. _? What is the final packet of the handshake?
Type here to search
23°C
11:23 AM
10/3/2025

Proof.

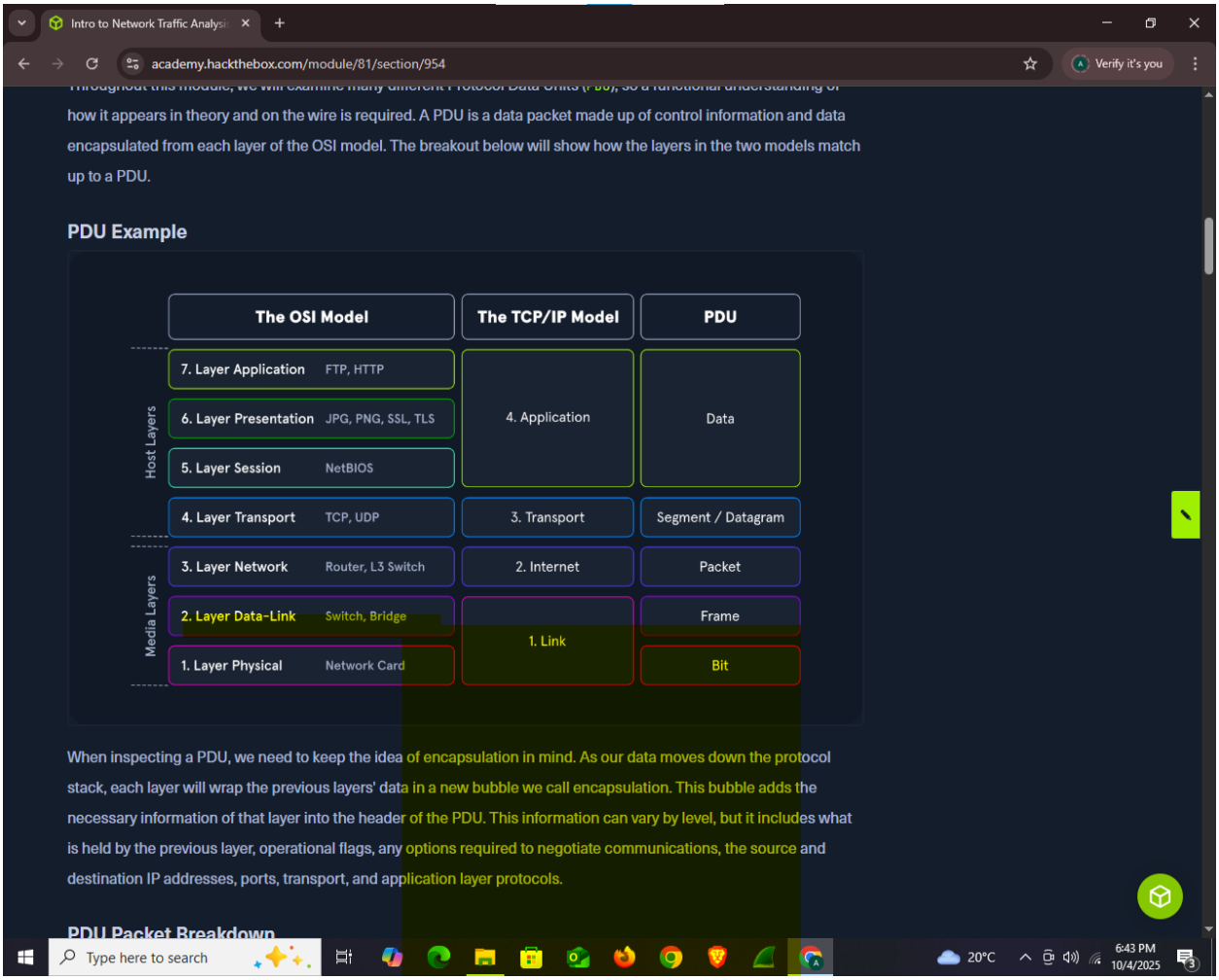
how it appears in theory and on the wire is required. A PDU is a data packet made up of control information and data encapsulated from each layer of the OSI model. The breakout below will show how the layers in the two models match up to a PDU.

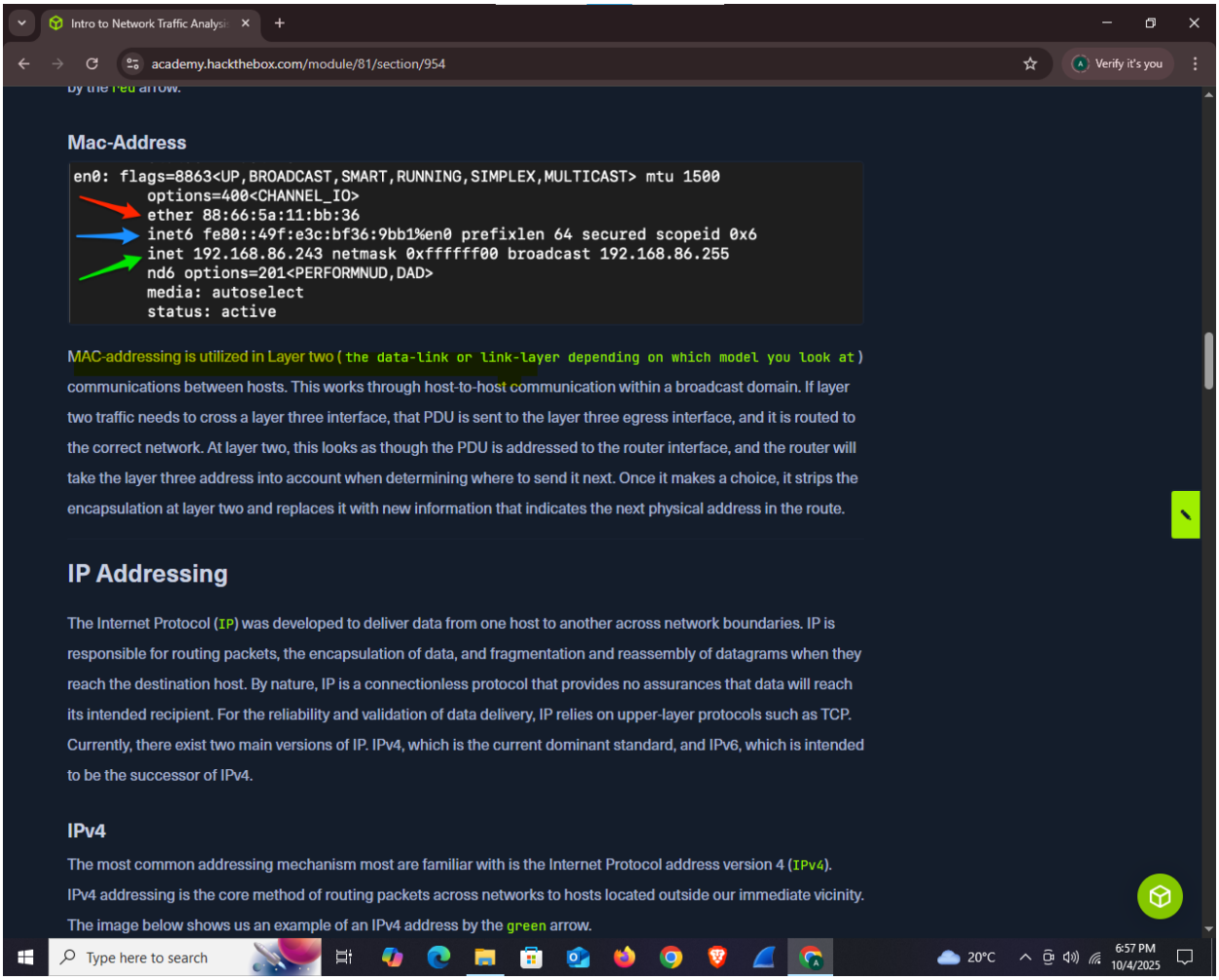
PDU Example

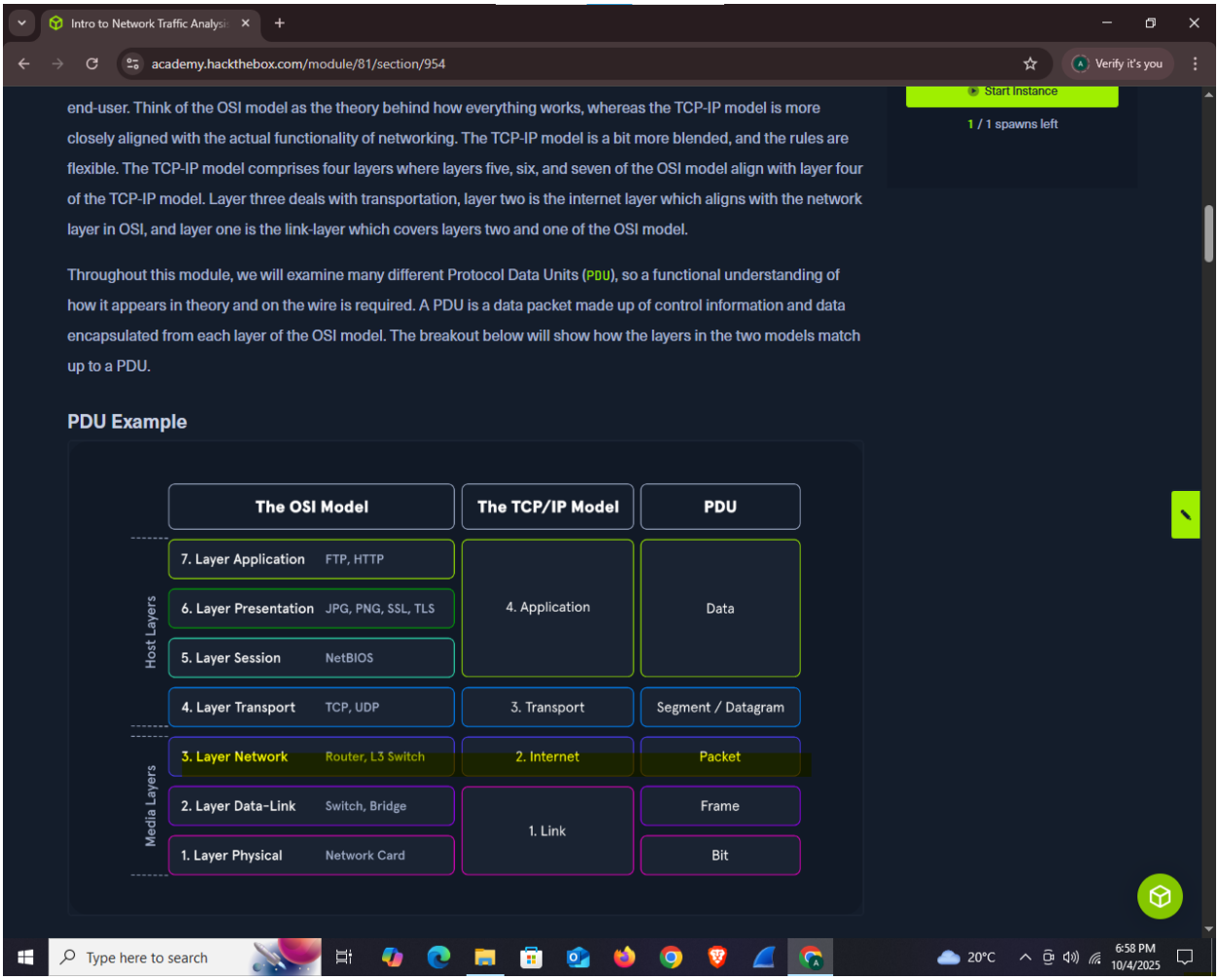


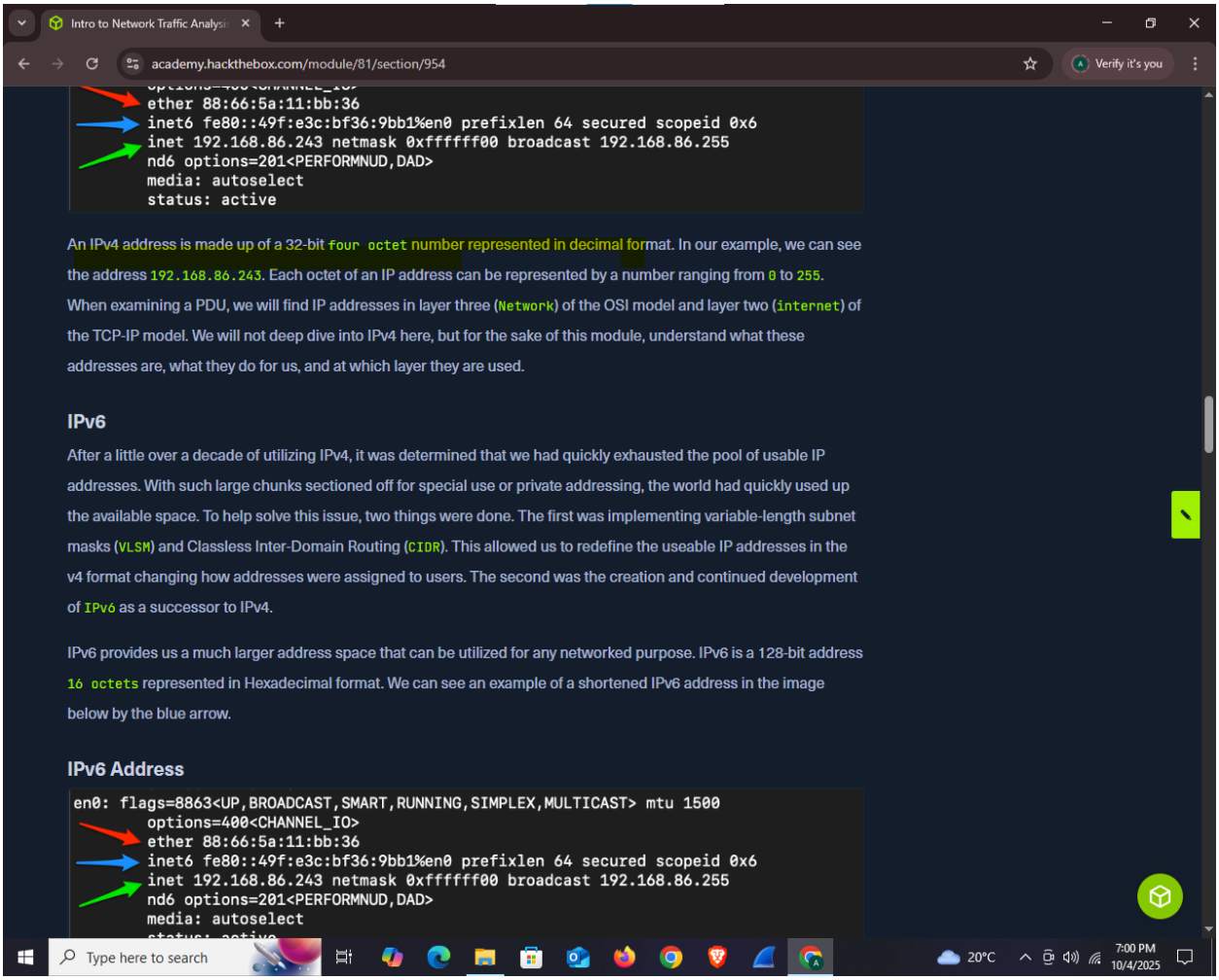
When inspecting a PDU, we need to keep the idea of encapsulation in mind. As our data moves down the protocol stack, each layer will wrap the previous layers' data in a new bubble we call encapsulation. This bubble adds the necessary information of that layer into the header of the PDU. This information can vary by level, but it includes what is held by the previous layer, operational flags, any options required to negotiate communications, the source and destination IP addresses, ports, transport, and application layer protocols.

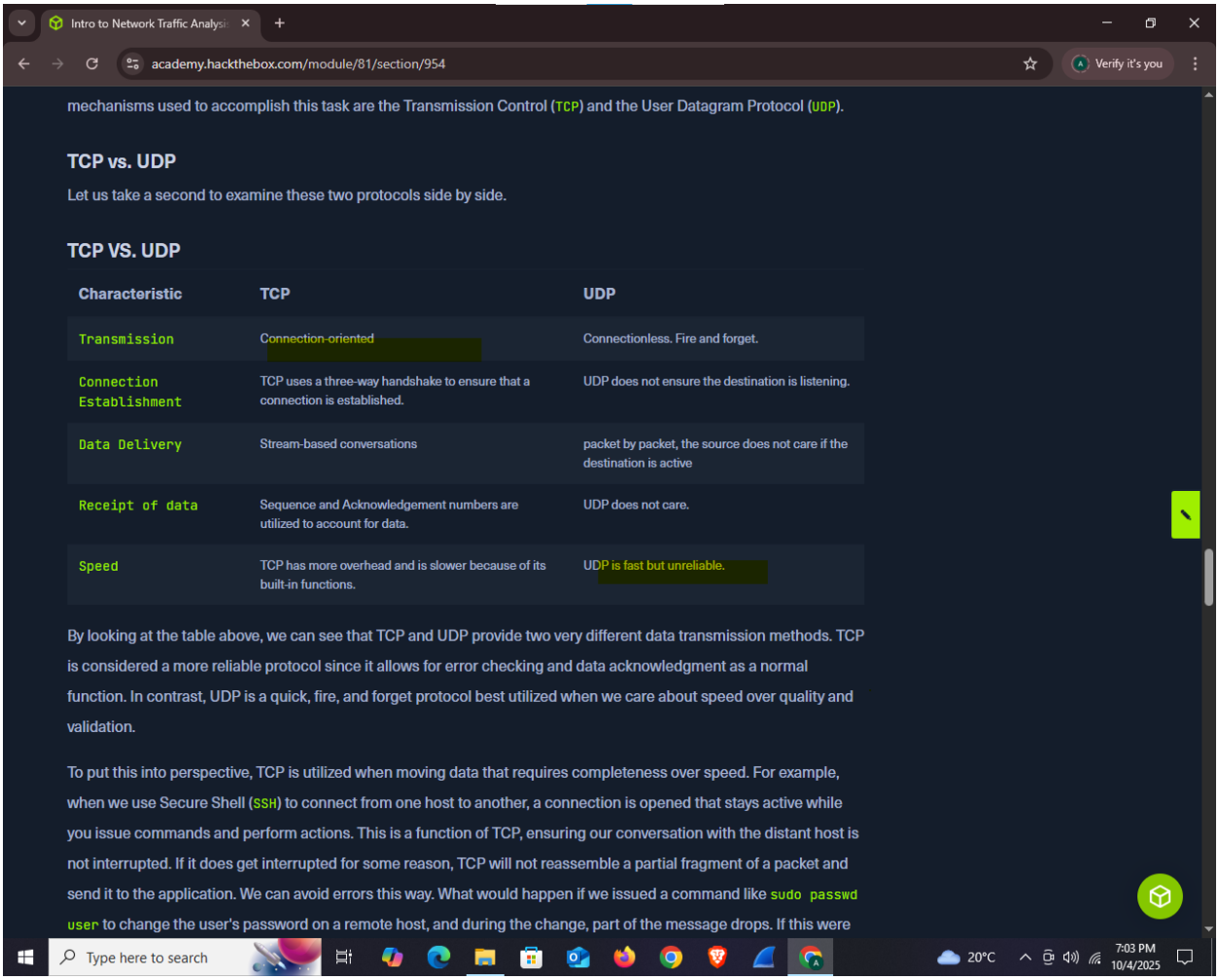
PDU Packet Breakdown

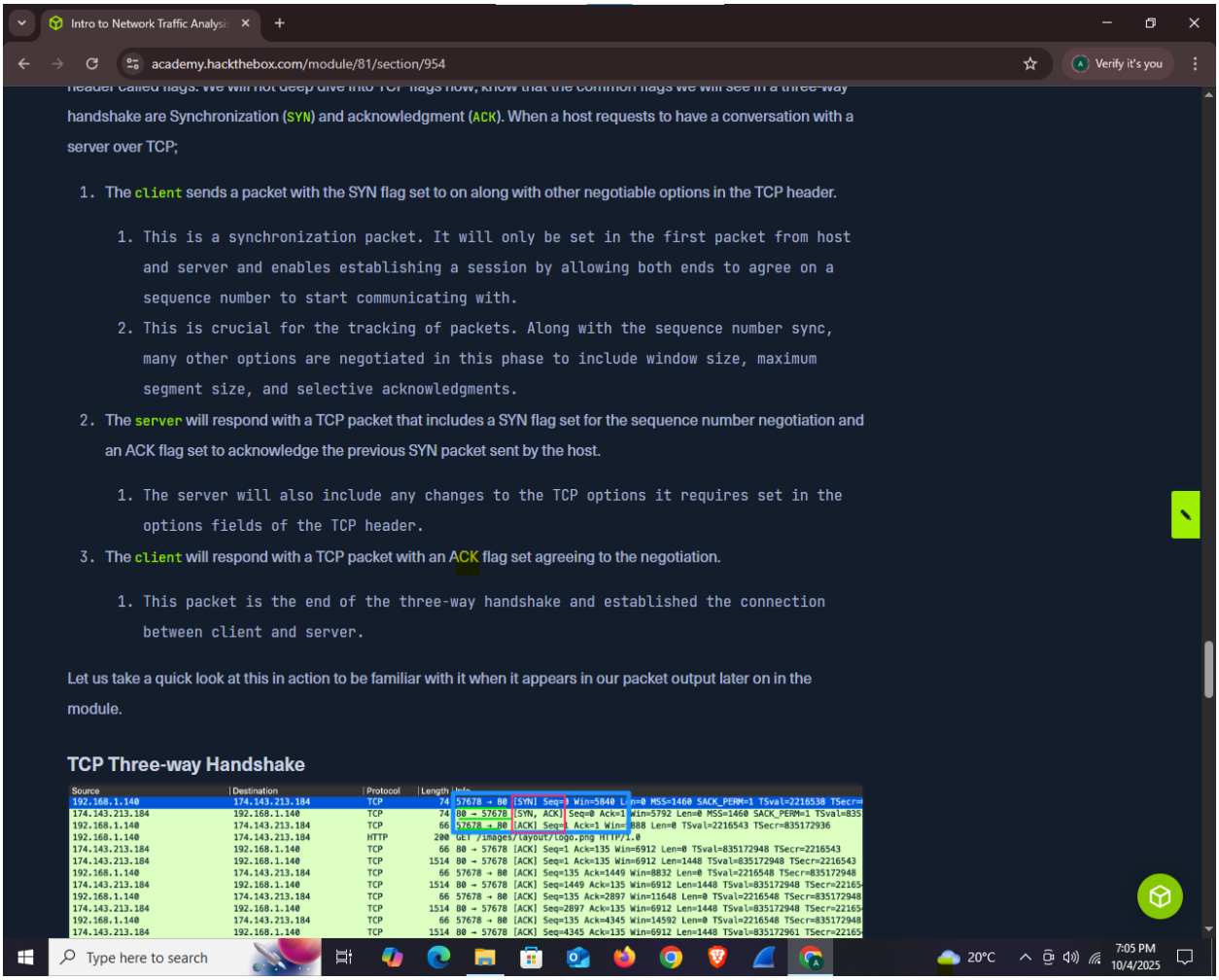

Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/954
Verify it's you
how it appears in theory and on the wire is required. A PDU is a data packet made up of control information and data encapsulated from each layer of the OSI model. The breakout below will show how the layers in the two models match up to a PDU.
PDU Example
The OSI Model
The TCP/IP Model
PDU
Host Layers
7. Layer Application FTP, HTTP
6. Layer Presentation JPG, PNG, SSL, TLS
5. Layer Session NetBIOS
4. Application
Data
4. Layer Transport TCP, UDP
3. Transport
Segment / Datagram
Media Layers
3. Layer Network Router, L3 Switch
2. Internet
Packet
2. Layer Data-Link Switch, Bridge
1. Link
Frame
1. Layer Physical Network Card
Bit
When inspecting a PDU, we need to keep the idea of encapsulation in mind. As our data moves down the protocol stack, each layer will wrap the previous layers' data in a new bubble we call encapsulation. This bubble adds the necessary information of that layer into the header of the PDU. This information can vary by level, but it includes what is held by the previous layer, operational flags, any options required to negotiate communications, the source and destination IP addresses, ports, transport, and application layer protocols.
PDU Packet Breakdown
Type here to search
20°C
6:43 PM
10/4/2025


Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/954
Verify it's you
by the red arrow.
Mac-Address
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
options=400<CHANNEL_IO>
ether 88:66:5a:11:bb:36
inet6 fe80::49f:e3c:bf36:9bb1%en0 prefixlen 64 secured scopeid 0x6
inet 192.168.86.243 netmask 0xffffff00 broadcast 192.168.86.255
nd6 options=201<PERFORMNUD,DAD>
media: autoselect
status: active
MAC-addressing is utilized in Layer two (the data-link or link-layer depending on which model you look at) communications between hosts. This works through host-to-host communication within a broadcast domain. If layer two traffic needs to cross a layer three interface, that PDU is sent to the layer three egress interface, and it is routed to the correct network. At layer two, this looks as though the PDU is addressed to the router interface, and the router will take the layer three address into account when determining where to send it next. Once it makes a choice, it strips the encapsulation at layer two and replaces it with new information that indicates the next physical address in the route.
IP Addressing
The Internet Protocol (IP) was developed to deliver data from one host to another across network boundaries. IP is responsible for routing packets, the encapsulation of data, and fragmentation and reassembly of datagrams when they reach the destination host. By nature, IP is a connectionless protocol that provides no assurances that data will reach its intended recipient. For the reliability and validation of data delivery, IP relies on upper-layer protocols such as TCP. Currently, there exist two main versions of IP. IPv4, which is the current dominant standard, and IPv6, which is intended to be the successor of IPv4.
IPv4
The most common addressing mechanism most are familiar with is the Internet Protocol address version 4 (IPv4). IPv4 addressing is the core method of routing packets across networks to hosts located outside our immediate vicinity. The image below shows us an example of an IPv4 address by the green arrow.
Type here to search
20°C
6:57 PM
10/4/2025

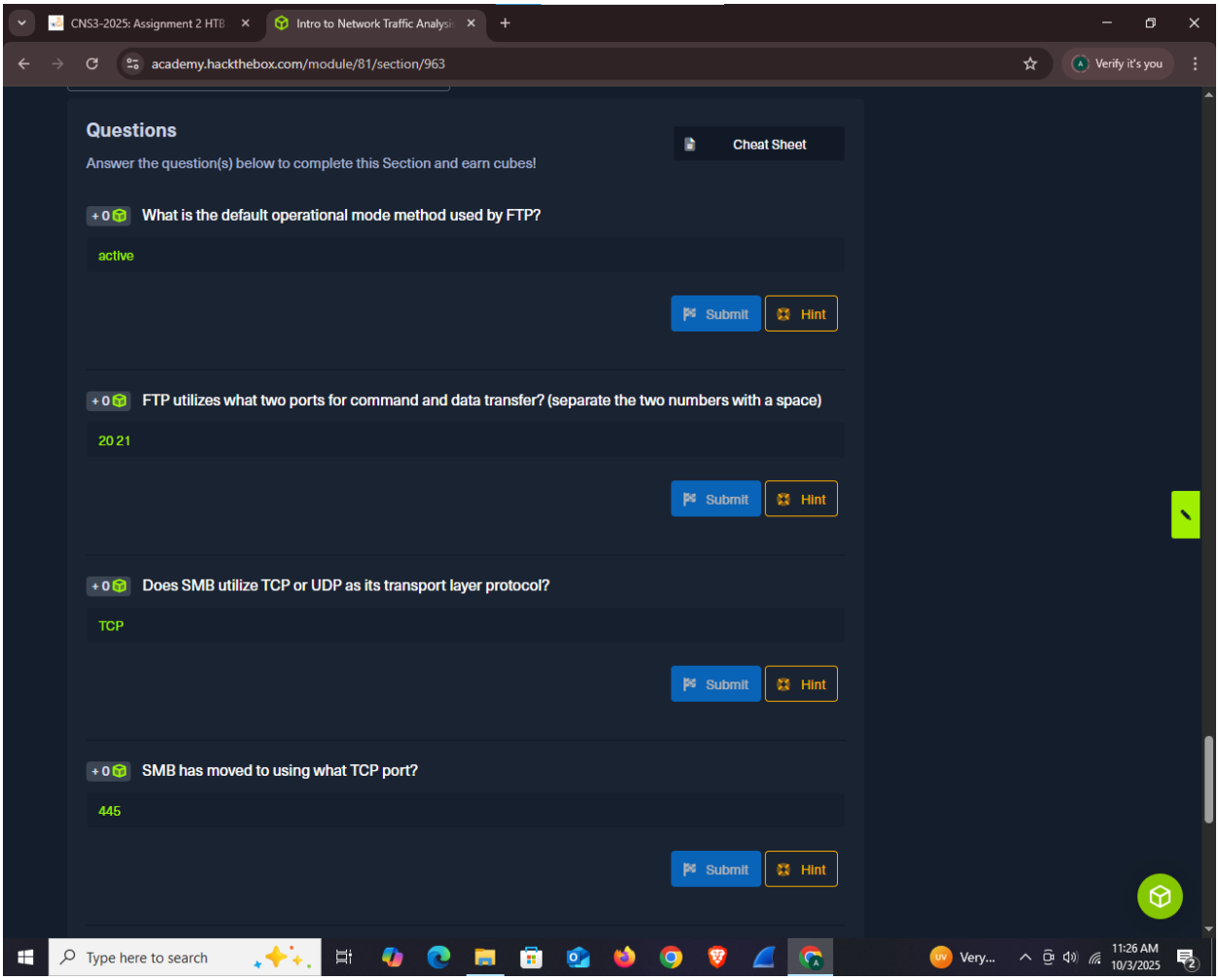

Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/954
Verify it's you
Start Instance
1 / 1 spawns left
end-user. Think of the OSI model as the theory behind how everything works, whereas the TCP-IP model is more closely aligned with the actual functionality of networking. The TCP-IP model is a bit more blended, and the rules are flexible. The TCP-IP model comprises four layers where layers five, six, and seven of the OSI model align with layer four of the TCP-IP model. Layer three deals with transportation, layer two is the internet layer which aligns with the network layer in OSI, and layer one is the link-layer which covers layers two and one of the OSI model.
Throughout this module, we will examine many different Protocol Data Units (PDU), so a functional understanding of how it appears in theory and on the wire is required. A PDU is a data packet made up of control information and data encapsulated from each layer of the OSI model. The breakout below will show how the layers in the two models match up to a PDU.
PDU Example
The OSI Model
The TCP/IP Model
PDU
Host Layers
7. Layer Application FTP, HTTP
6. Layer Presentation JPG, PNG, SSL, TLS
5. Layer Session NetBIOS
4. Application
Data
4. Layer Transport TCP, UDP
3. Transport
Segment / Datagram
Media Layers
3. Layer Network Router, L3 Switch
2. Internet
Packet
2. Layer Data-Link Switch, Bridge
1. Link
Frame
1. Layer Physical Network Card
Bit
Type here to search
20°C
6:58 PM
10/4/2025

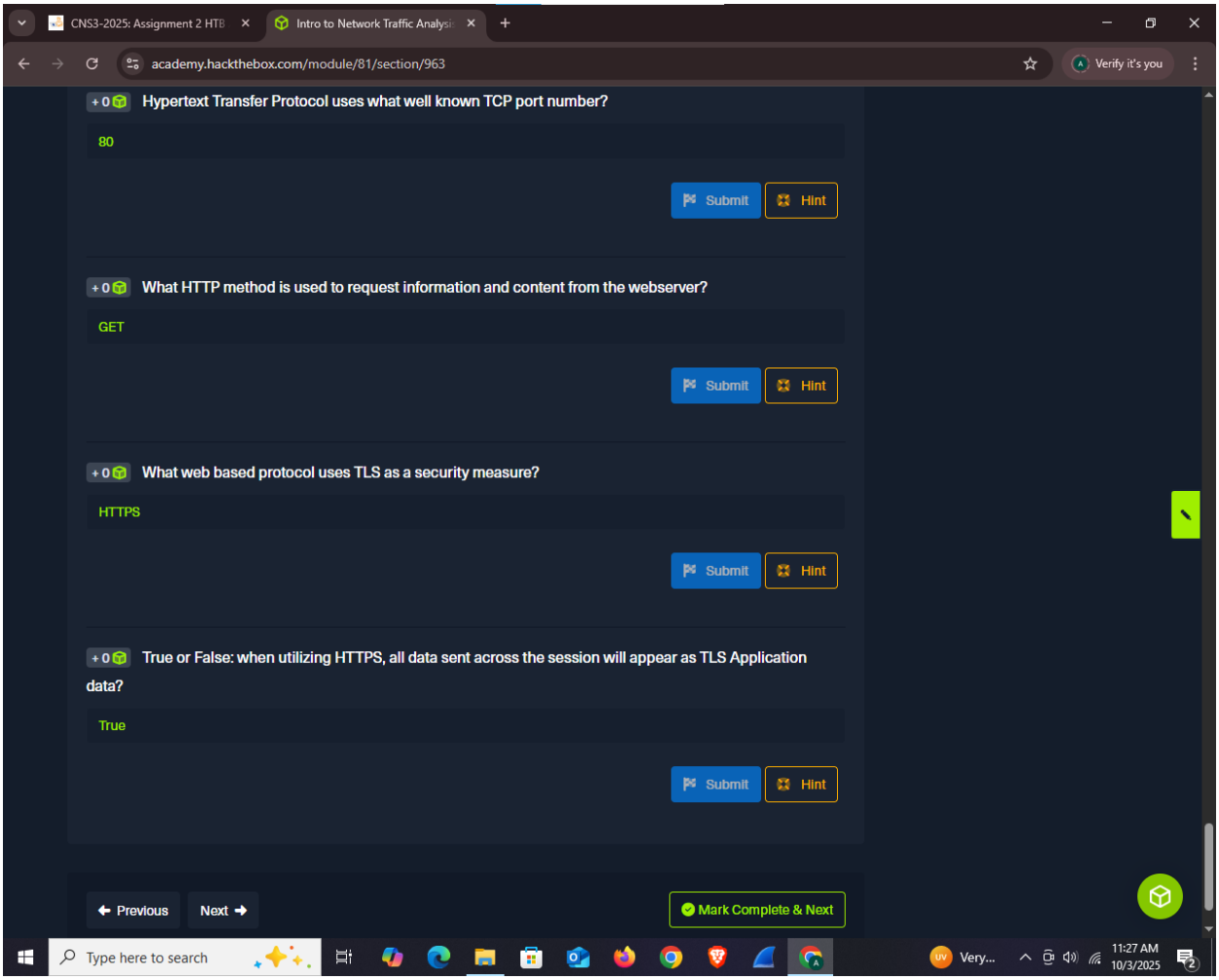

Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/954
Verify it's you
ether 88:66:5a:11:bb:36
inet6 fe80::49f:e3c:bf36:9bb1%en0 prefixlen 64 secured scopeid 0x6
inet 192.168.86.243 netmask 0xffffff00 broadcast 192.168.86.255
nd6 options=201<PERFORMNUD,DAD>
media: autoselect
status: active
An IPv4 address is made up of a 32-bit four octet number represented in decimal format. In our example, we can see the address 192.168.86.243. Each octet of an IP address can be represented by a number ranging from 0 to 255. When examining a PDU, we will find IP addresses in layer three (Network) of the OSI model and layer two (internet) of the TCP-IP model. We will not deep dive into IPv4 here, but for the sake of this module, understand what these addresses are, what they do for us, and at which layer they are used.
IPv6
After a little over a decade of utilizing IPv4, it was determined that we had quickly exhausted the pool of usable IP addresses. With such large chunks sectioned off for special use or private addressing, the world had quickly used up the available space. To help solve this issue, two things were done. The first was implementing variable-length subnet masks (VLSM) and Classless Inter-Domain Routing (CIDR). This allowed us to redefine the useable IP addresses in the v4 format changing how addresses were assigned to users. The second was the creation and continued development of IPv6 as a successor to IPv4.
IPv6 provides us a much larger address space that can be utilized for any networked purpose. IPv6 is a 128-bit address 16 octets represented in Hexadecimal format. We can see an example of a shortened IPv6 address in the image below by the blue arrow.
IPv6 Address
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
options=400<CHANNEL_IO>
ether 88:66:5a:11:bb:36
inet6 fe80::49f:e3c:bf36:9bb1%en0 prefixlen 64 secured scopeid 0x6
inet 192.168.86.243 netmask 0xffffff00 broadcast 192.168.86.255
nd6 options=201<PERFORMNUD,DAD>
media: autoselect
Type here to search
20°C
7:00 PM
10/4/2025

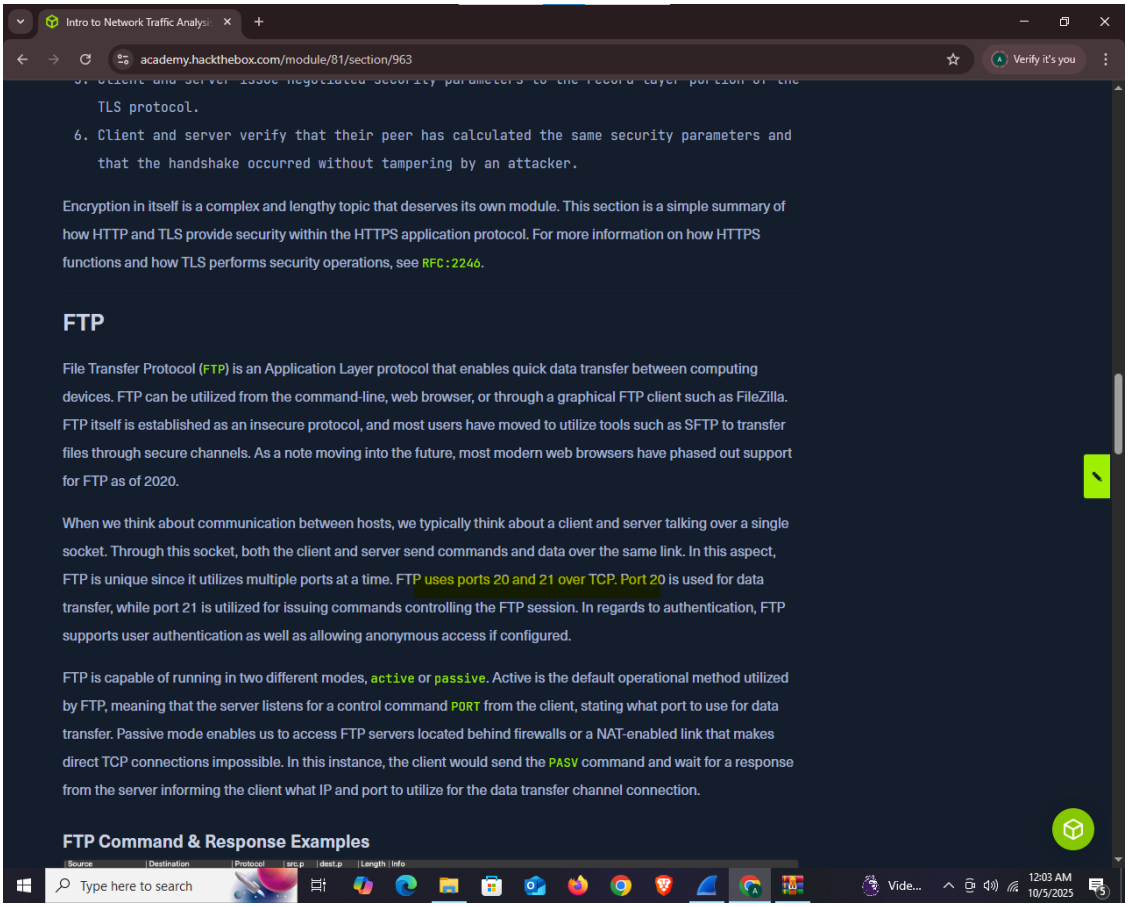

Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/954
Verify it's you
mechanisms used to accomplish this task are the Transmission Control (TCP) and the User Datagram Protocol (UDP).
TCP vs. UDP
Let us take a second to examine these two protocols side by side.
TCP VS. UDP
Characteristic
TCP
UDP
Transmission
Connection-oriented
Connectionless. Fire and forget.
Connection Establishment
TCP uses a three-way handshake to ensure that a connection is established.
UDP does not ensure the destination is listening.
Data Delivery
Stream-based conversations
packet by packet, the source does not care if the destination is active
Receipt of data
Sequence and Acknowledgement numbers are utilized to account for data.
UDP does not care.
Speed
TCP has more overhead and is slower because of its built-in functions.
UDP is fast but unreliable.
By looking at the table above, we can see that TCP and UDP provide two very different data transmission methods. TCP is considered a more reliable protocol since it allows for error checking and data acknowledgment as a normal function. In contrast, UDP is a quick, fire, and forget protocol best utilized when we care about speed over quality and validation.
To put this into perspective, TCP is utilized when moving data that requires completeness over speed. For example, when we use Secure Shell (SSH) to connect from one host to another, a connection is opened that stays active while you issue commands and perform actions. This is a function of TCP, ensuring our conversation with the distant host is not interrupted. If it does get interrupted for some reason, TCP will not reassemble a partial fragment of a packet and send it to the application. We can avoid errors this way. What would happen if we issued a command like sudo passwd user to change the user's password on a remote host, and during the change, part of the message drops. If this were
Type here to search
20°C
7:03 PM
10/4/2025



Question 2.networking premier layer 5-7.


CNS3-2025: Assignment 2 HTB
Intro to Network Traffic Analysi
academy.hackthebox.com/module/81/section/963
Verify it's you
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
+0 What is the default operational mode method used by FTP?
active
Submit
Hint
+0 FTP utilizes what two ports for command and data transfer? (separate the two numbers with a space)
20 21
Submit
Hint
+0 Does SMB utilize TCP or UDP as its transport layer protocol?
TCP
Submit
Hint
+0 SMB has moved to using what TCP port?
445
Submit
Hint
Type here to search
Very...
11:26 AM
10/3/2025


CNS3-2025: Assignment 2 HTB
Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/963
Verify it's you
+0 Hypertext Transfer Protocol uses what well known TCP port number?
80
Submit
Hint
+0 What HTTP method is used to request information and content from the webserver?
GET
Submit
Hint
+0 What web based protocol uses TLS as a security measure?
HTTPS
Submit
Hint
+0 True or False: when utilizing HTTPS, all data sent across the session will appear as TLS Application data?
True
Submit
Hint
Previous
Next
Mark Complete & Next
Type here to search
Very...
11:27 AM
10/3/2025


Intro to Network Traffic Analysi
academy.hackthebox.com/module/81/section/963
Verify it's you
TLS protocol.
6. Client and server verify that their peer has calculated the same security parameters and that the handshake occurred without tampering by an attacker.
Encryption in itself is a complex and lengthy topic that deserves its own module. This section is a simple summary of how HTTP and TLS provide security within the HTTPS application protocol. For more information on how HTTPS functions and how TLS performs security operations, see RFC:2246.
FTP
File Transfer Protocol (FTP) is an Application Layer protocol that enables quick data transfer between computing devices. FTP can be utilized from the command-line, web browser, or through a graphical FTP client such as FileZilla. FTP itself is established as an insecure protocol, and most users have moved to utilize tools such as SFTP to transfer files through secure channels. As a note moving into the future, most modern web browsers have phased out support for FTP as of 2020.
When we think about communication between hosts, we typically think about a client and server talking over a single socket. Through this socket, both the client and server send commands and data over the same link. In this aspect, FTP is unique since it utilizes multiple ports at a time. FTP uses ports 20 and 21 over TCP. Port 20 is used for data transfer, while port 21 is utilized for issuing commands controlling the FTP session. In regards to authentication, FTP supports user authentication as well as allowing anonymous access if configured.
FTP is capable of running in two different modes, active or passive. Active is the default operational method utilized by FTP, meaning that the server listens for a control command PORT from the client, stating what port to use for data transfer. Passive mode enables us to access FTP servers located behind firewalls or a NAT-enabled link that makes direct TCP connections impossible. In this instance, the client would send the PASV command and wait for a response from the server informing the client what IP and port to utilize for the data transfer channel connection.
FTP Command & Response Examples
Source
Destination
Protocol
src.p
dest.p
Length
Info
Type here to search
Vide...
12:03 AM
10/5/2025

Proof.

`QUIT`	ends the session.

This is not an exhaustive list of the possible FTP control commands that could be seen. These can vary based on the FTP application or shell in use. For more information on FTP, see `RFC:959`.

SMB

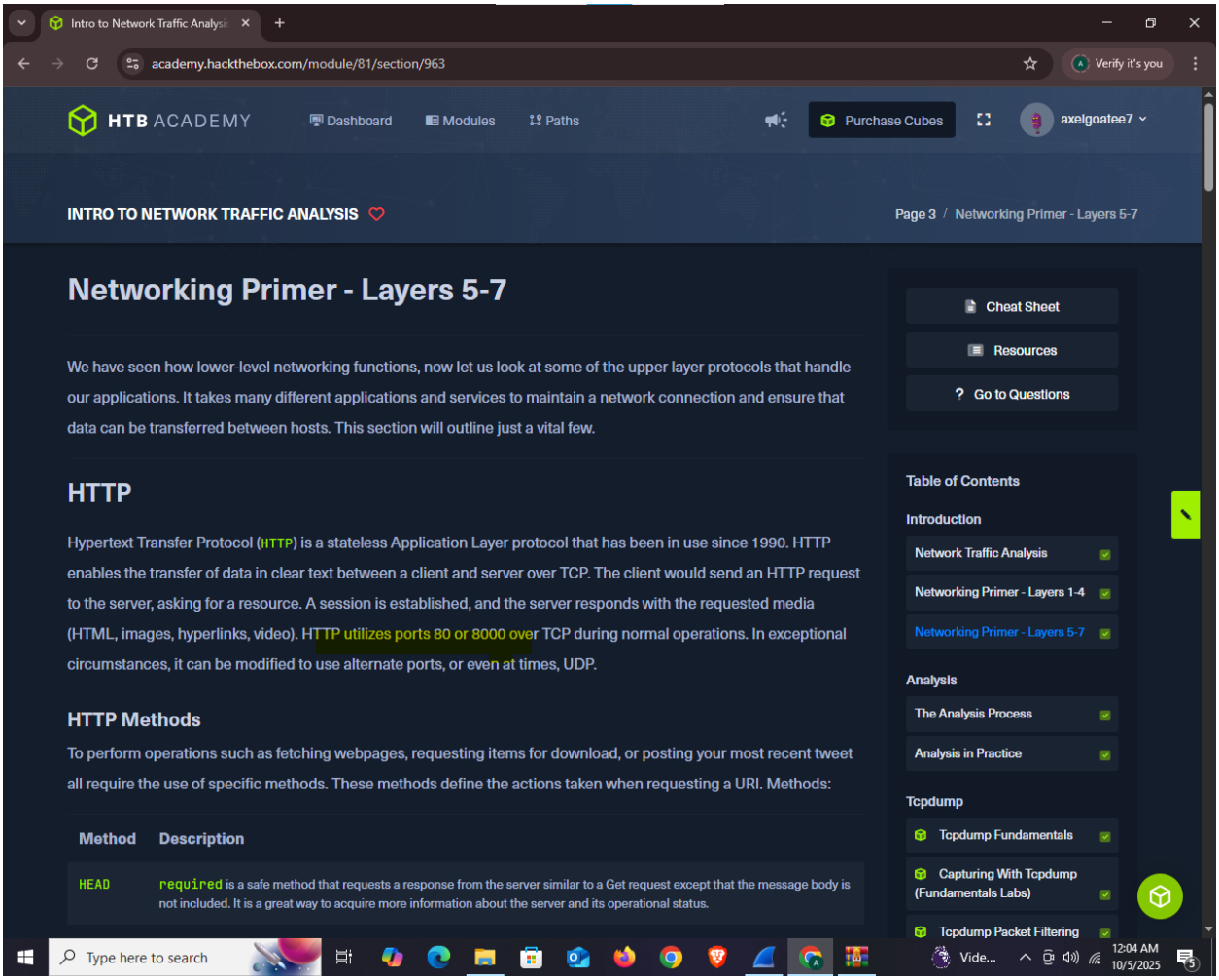
Server Message Block (`SMB`) is a protocol most widely seen in Windows enterprise environments that enables sharing resources between hosts over common networking architectures. SMB is a connection-oriented protocol that requires user authentication from the host to the resource to ensure the user has correct permissions to use that resource or perform actions. In the past, SMB utilized NetBIOS as its transport mechanism over UDP ports 137 and 138. Since modern changes, SMB now supports direct TCP transport over port 445, NetBIOS over TCP port 139, and even the QUIC protocol.

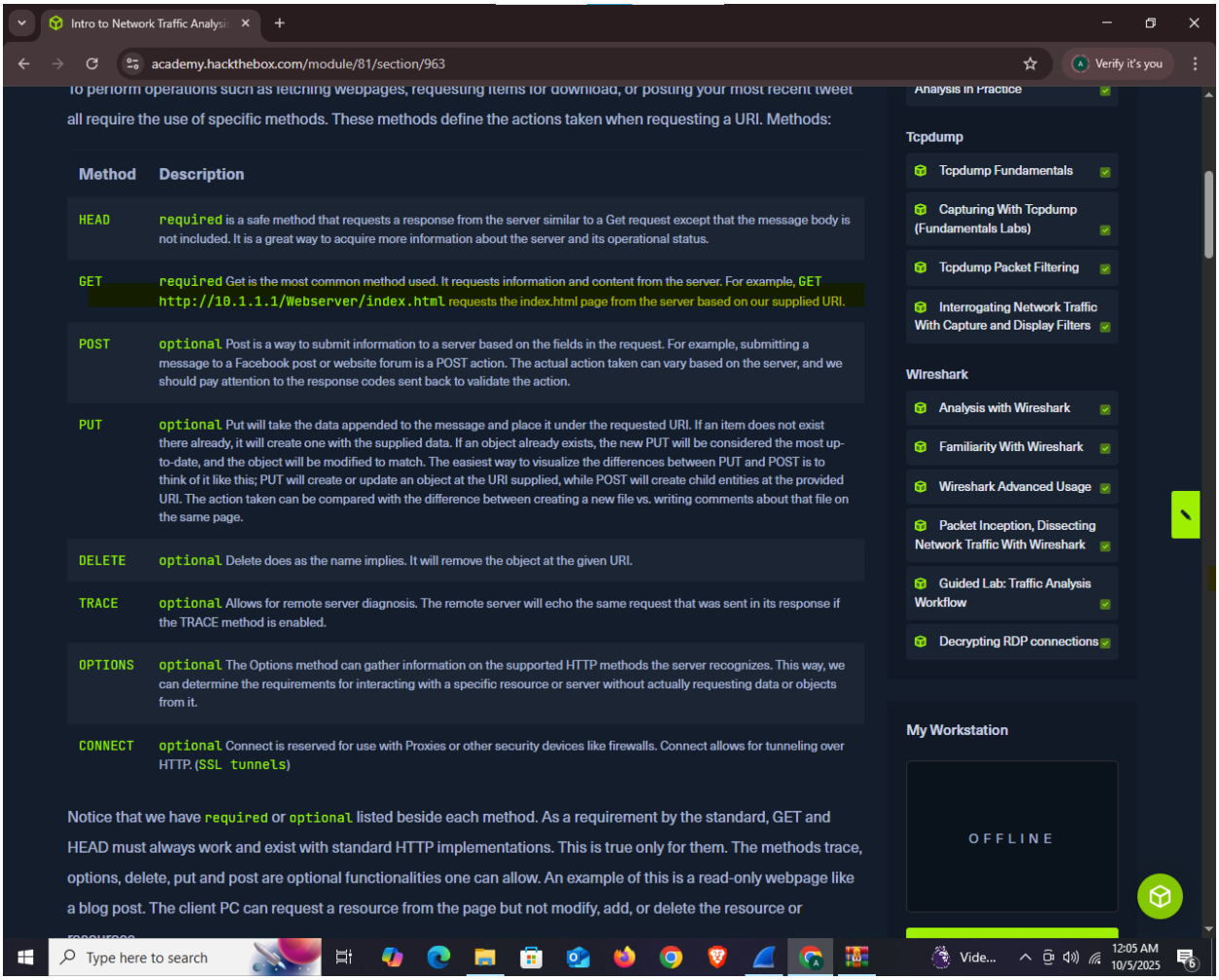
As a user, SMB provides us easy and convenient access to resources like printers, shared drives, authentication servers, and more. For this reason, SMB is very attractive to potential attackers as well.

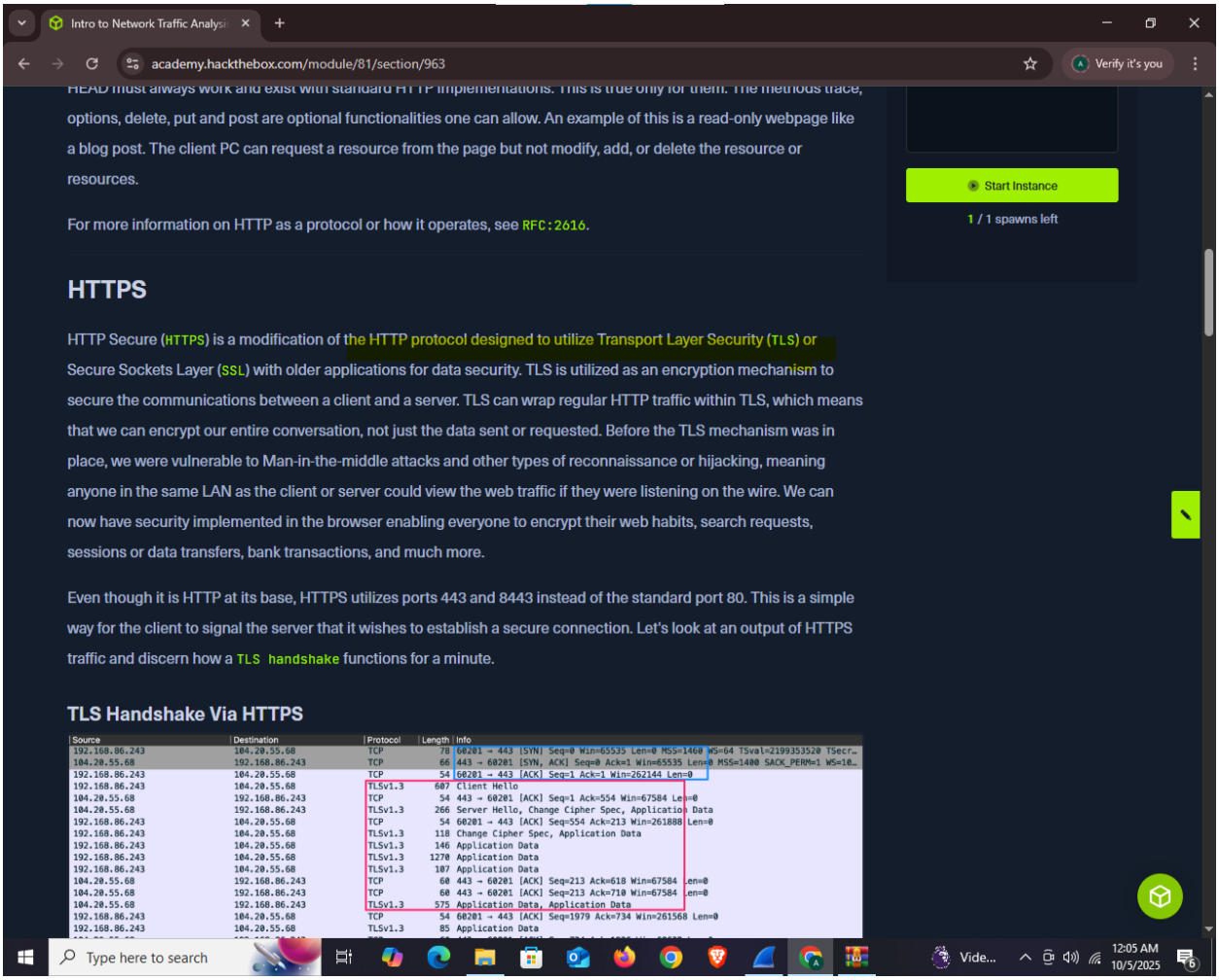
Like any other application that uses TCP as its transport mechanism, it will perform standard functions like the three-way handshake and acknowledging received packets. Let us take a second to look at some SMB traffic to familiarize ourselves.

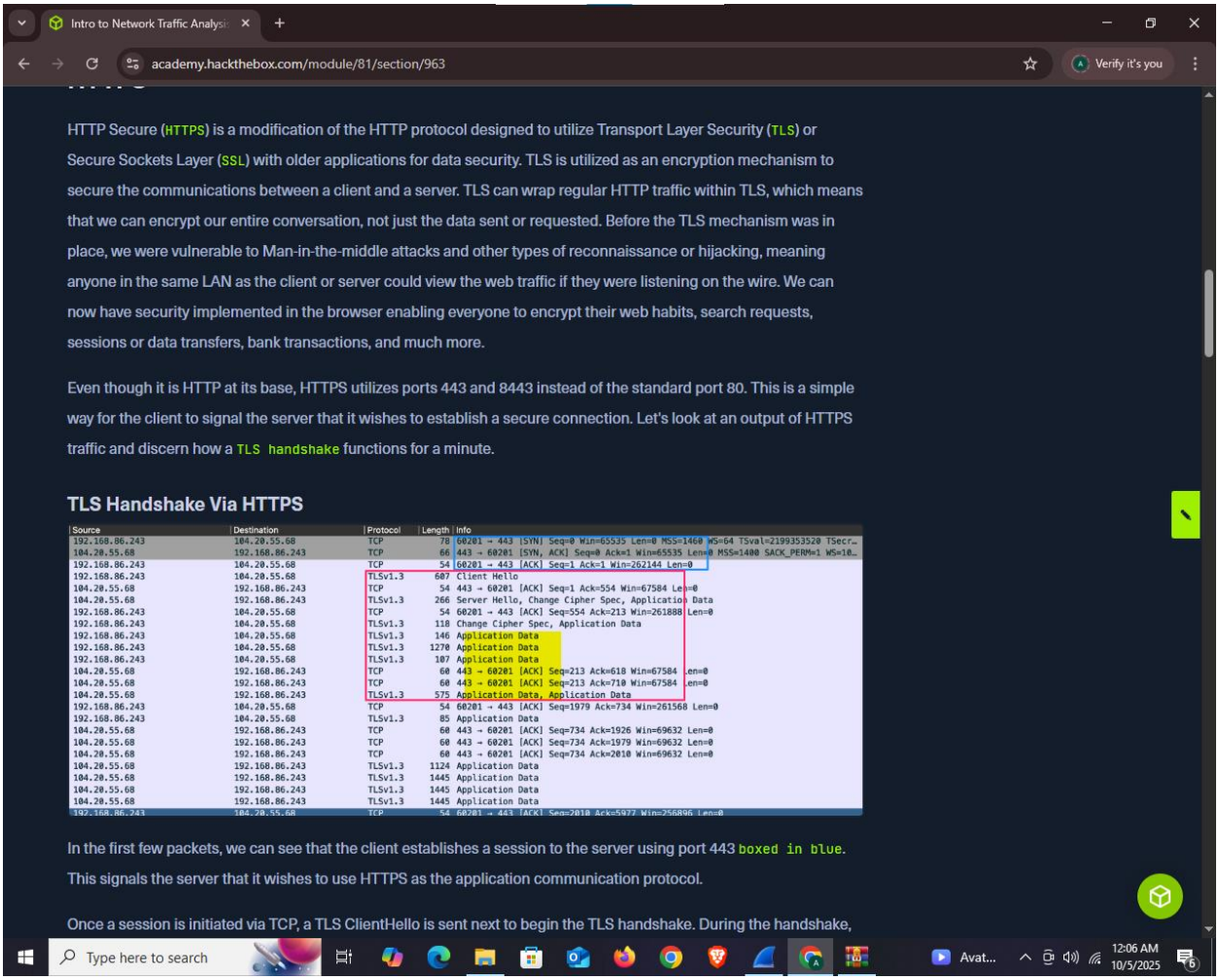
SMB On The Wire

Source	Destination	Protocol	src.p	dest.p	Length	Info
192.168.199.132	192.168.199.255	NBNS	137	137	92	Name query NB WPAD<00>
192.168.199.132	SCV	DNS	537…	53	86	Standard query 0x142e A officeclient.microsoft.com
192.168.199.133	SCV	DNS	643…	53	92	Standard query 0xfa7d A geo-prod.do.dsp.mp.microsoft.com
VMware_61:f5:5f	SCV	ARP			42	Who has 192.168.199.1? Tell 192.168.199.133
SCV	VMware_61:f5:5f	ARP			42	192.168.199.1 is at 00:50:56:c0:00:01
192.168.199.132	192.168.199.133	TCP	496…	445	66	49670 → 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.133	192.168.199.132	TCP	445	496…	66	445 → 49670 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.132	192.168.199.133	TCP	496…	445	54	49670 → 445 [ACK] Seq=1 Ack=1 Win=65536 Len=0
192.168.199.132	192.168.199.133	SMB	496…	445	213	Negotiate Protocol Request
192.168.199.133	192.168.199.132	SMB2	445	496…	506	Negotiate Protocol Response
192.168.199.132	192.168.199.133	SMB2	496…	445	232	Negotiate Protocol Request
192.168.199.133	192.168.199.132	SMB2	445	496…	566	Negotiate Protocol Response
192.168.199.132	192.168.199.133	SMB2	496…	445	220	Session Setup Request, NTLMSSP_NEGOTIATE
192.168.199.133	192.168.199.132	SMB2	445	496…	401	Session Setup Response, Error: STATUS_MORE_PROCESSING_REQUIRED, NTLMSSP_CHALLENGE
192.168.199.132	192.168.199.133	SMB2	496…	445	711	Session Setup Request, NTLMSSP_AUTH, User: DESKTOP-2AEFM7G\user
192.168.199.133	192.168.199.132	SMB2	445	496…	131	Session Setup Response, Error: STATUS_LOGON_FAILURE
192.168.199.132	192.168.199.133	TCP	496…	445	54	49670 → 445 [RST, ACK] Seq=1161 Ack=1389 Win=0 Len=0
192.168.199.132	192.168.199.133	TCP	496…	445	66	49671 → 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1

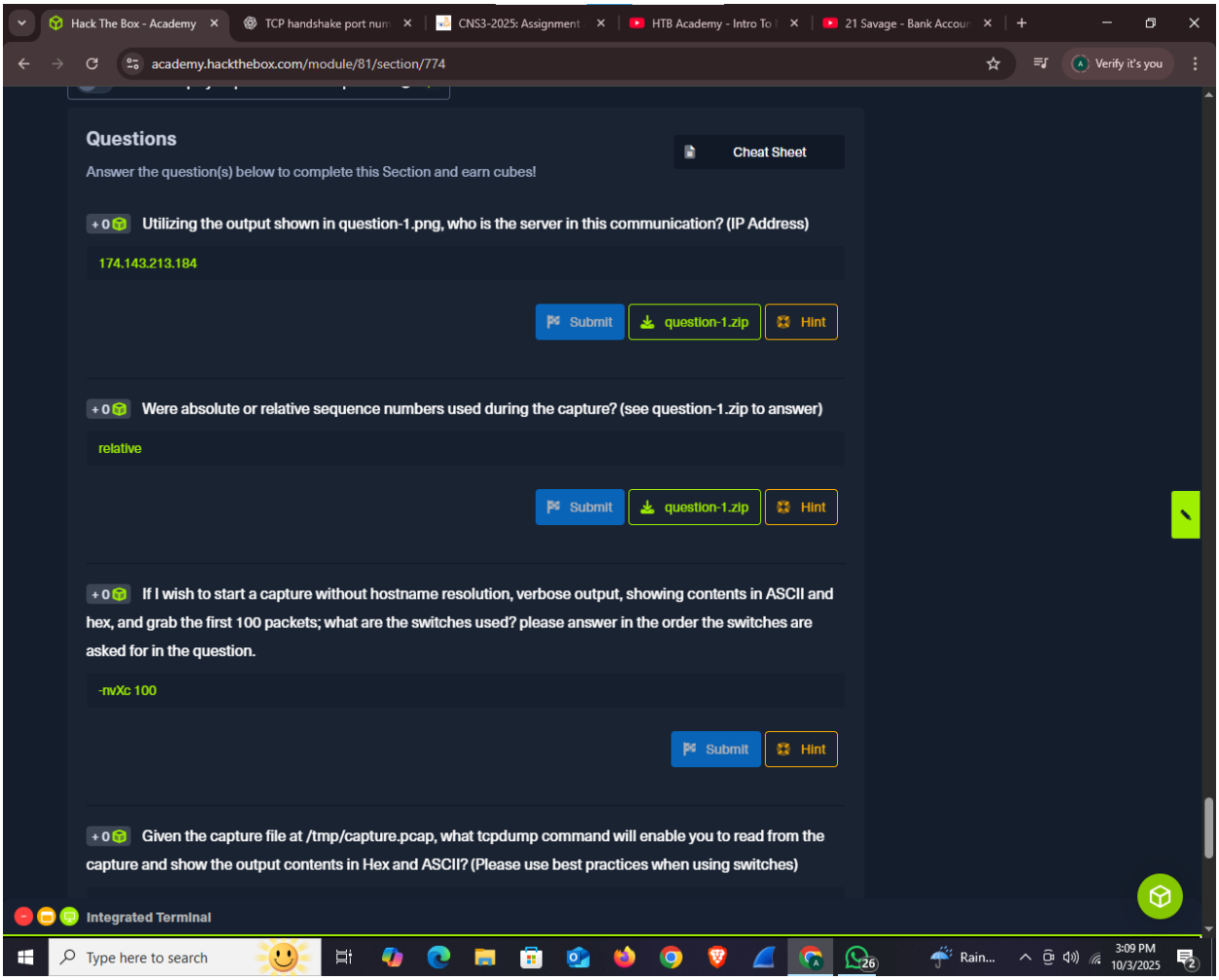

Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/963
Verify it's you
HTB ACADEMY
Dashboard
Modules
Paths
Purchase Cubes
axelgoatee7
INTRO TO NETWORK TRAFFIC ANALYSIS
Page 3 / Networking Primer - Layers 5-7
Networking Primer - Layers 5-7
We have seen how lower-level networking functions, now let us look at some of the upper layer protocols that handle our applications. It takes many different applications and services to maintain a network connection and ensure that data can be transferred between hosts. This section will outline just a vital few.
HTTP
Hypertext Transfer Protocol (HTTP) is a stateless Application Layer protocol that has been in use since 1990. HTTP enables the transfer of data in clear text between a client and server over TCP. The client would send an HTTP request to the server, asking for a resource. A session is established, and the server responds with the requested media (HTML, images, hyperlinks, video). HTTP utilizes ports 80 or 8000 over TCP during normal operations. In exceptional circumstances, it can be modified to use alternate ports, or even at times, UDP.
HTTP Methods
To perform operations such as fetching webpages, requesting items for download, or posting your most recent tweet all require the use of specific methods. These methods define the actions taken when requesting a URI. Methods:
Method
Description
HEAD
required is a safe method that requests a response from the server similar to a Get request except that the message body is not included. It is a great way to acquire more information about the server and its operational status.
Cheat Sheet
Resources
Go to Questions
Table of Contents
Introduction
Network Traffic Analysis
Networking Primer - Layers 1-4
Networking Primer - Layers 5-7
Analysis
The Analysis Process
Analysis in Practice
Tcpdump
Tcpdump Fundamentals
Capturing With Tcpdump (Fundamentals Labs)
Tcpdump Packet Filtering
Type here to search
12:04 AM
10/5/2025

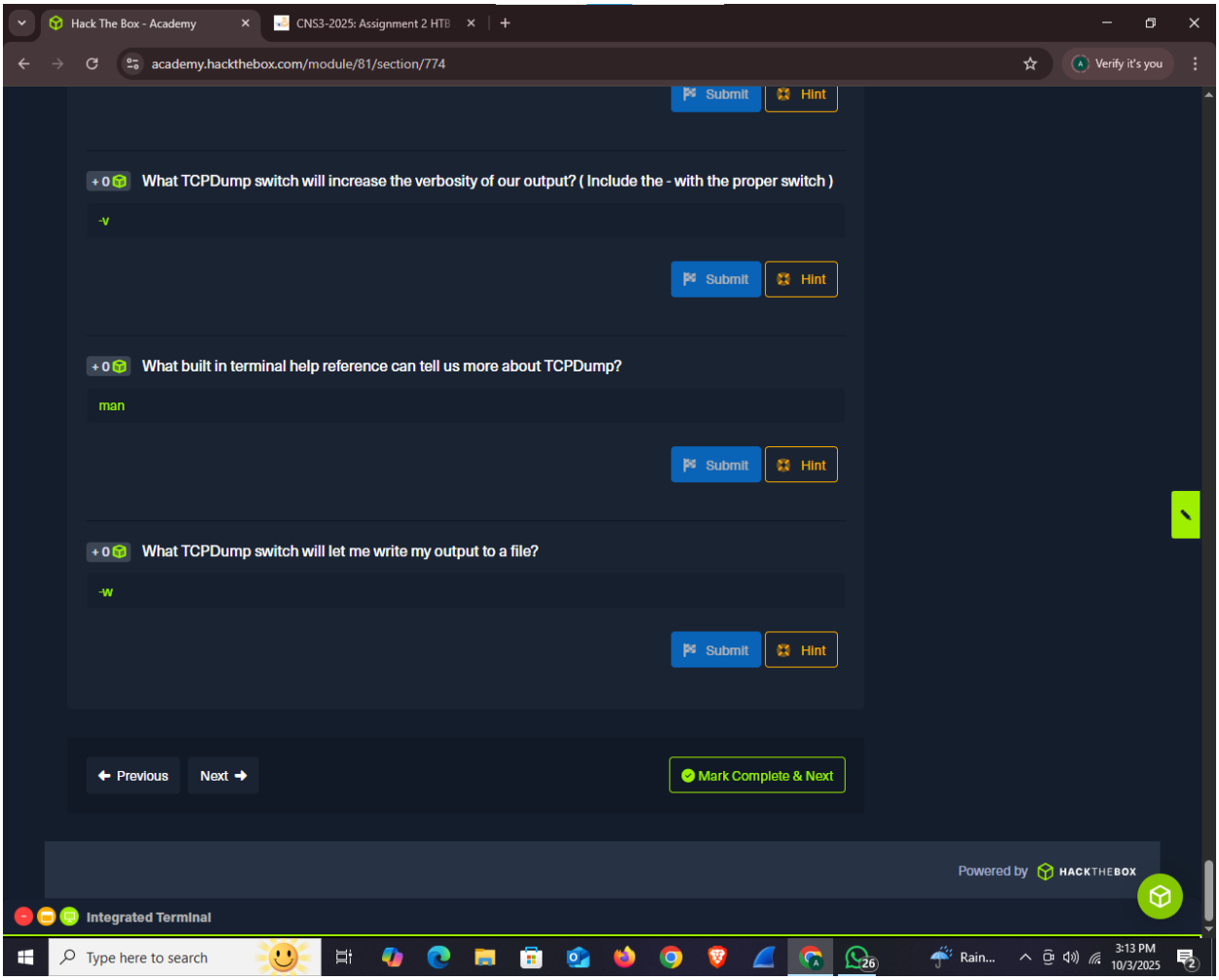

Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/963
Verify it's you
To perform operations such as fetching webpages, requesting items for download, or posting your most recent tweet all require the use of specific methods. These methods define the actions taken when requesting a URI. Methods:
Method
Description
HEAD
required is a safe method that requests a response from the server similar to a Get request except that the message body is not included. It is a great way to acquire more information about the server and its operational status.
GET
required Get is the most common method used. It requests information and content from the server. For example, GET http://10.1.1.1/Webserver/index.html requests the index.html page from the server based on our supplied URI.
POST
optional Post is a way to submit information to a server based on the fields in the request. For example, submitting a message to a Facebook post or website forum is a POST action. The actual action taken can vary based on the server, and we should pay attention to the response codes sent back to validate the action.
PUT
optional Put will take the data appended to the message and place it under the requested URI. If an item does not exist there already, it will create one with the supplied data. If an object already exists, the new PUT will be considered the most up-to-date, and the object will be modified to match. The easiest way to visualize the differences between PUT and POST is to think of it like this; PUT will create or update an object at the URI supplied, while POST will create child entities at the provided URI. The action taken can be compared with the difference between creating a new file vs. writing comments about that file on the same page.
DELETE
optional Delete does as the name implies. It will remove the object at the given URI.
TRACE
optional Allows for remote server diagnosis. The remote server will echo the same request that was sent in its response if the TRACE method is enabled.
OPTIONS
optional The Options method can gather information on the supported HTTP methods the server recognizes. This way, we can determine the requirements for interacting with a specific resource or server without actually requesting data or objects from it.
CONNECT
optional Connect is reserved for use with Proxies or other security devices like firewalls. Connect allows for tunneling over HTTP. (SSL tunnels)
Notice that we have required or optional listed beside each method. As a requirement by the standard, GET and HEAD must always work and exist with standard HTTP implementations. This is true only for them. The methods trace, options, delete, put and post are optional functionalities one can allow. An example of this is a read-only webpage like a blog post. The client PC can request a resource from the page but not modify, add, or delete the resource or
Analysis in Practice
Tcpdump
Tcpdump Fundamentals
Capturing With Tcpdump (Fundamentals Labs)
Tcpdump Packet Filtering
Interrogating Network Traffic With Capture and Display Filters
Wireshark
Analysis with Wireshark
Familiarity With Wireshark
Wireshark Advanced Usage
Packet Inception, Dissecting Network Traffic With Wireshark
Guided Lab: Traffic Analysis Workflow
Decrypting RDP connections
My Workstation
OFFLINE
Type here to search
Vide...
12:05 AM
10/5/2025


Intro to Network Traffic Analysis
academy.hackthebox.com/module/81/section/963
Verify it's you
HEAD must always work and exist with standard HTTP implementations. This is true only for them. The methods trace, options, delete, put and post are optional functionalities one can allow. An example of this is a read-only webpage like a blog post. The client PC can request a resource from the page but not modify, add, or delete the resource or resources.
For more information on HTTP as a protocol or how it operates, see RFC:2616.
Start Instance
1 / 1 spawns left
HTTPS
HTTP Secure (HTTPS) is a modification of the HTTP protocol designed to utilize Transport Layer Security (TLS) or Secure Sockets Layer (SSL) with older applications for data security. TLS is utilized as an encryption mechanism to secure the communications between a client and a server. TLS can wrap regular HTTP traffic within TLS, which means that we can encrypt our entire conversation, not just the data sent or requested. Before the TLS mechanism was in place, we were vulnerable to Man-in-the-middle attacks and other types of reconnaissance or hijacking, meaning anyone in the same LAN as the client or server could view the web traffic if they were listening on the wire. We can now have security implemented in the browser enabling everyone to encrypt their web habits, search requests, sessions or data transfers, bank transactions, and much more.
Even though it is HTTP at its base, HTTPS utilizes ports 443 and 8443 instead of the standard port 80. This is a simple way for the client to signal the server that it wishes to establish a secure connection. Let's look at an output of HTTPS traffic and discern how a TLS handshake functions for a minute.
TLS Handshake Via HTTPS
Source Destination Protocol Length Info
192.168.86.243 104.20.55.68 TCP 78 60201 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2199353520 TSecr…
104.20.55.68 192.168.86.243 TCP 66 443 → 60201 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1400 SACK_PERM=1 WS=10…
192.168.86.243 104.20.55.68 TCP 54 60201 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
192.168.86.243 104.20.55.68 TLSv1.3 607 Client Hello
104.20.55.68 192.168.86.243 TCP 54 443 → 60201 [ACK] Seq=1 Ack=554 Win=67584 Len=0
104.20.55.68 192.168.86.243 TLSv1.3 266 Server Hello, Change Cipher Spec, Application Data
192.168.86.243 104.20.55.68 TCP 54 60201 → 443 [ACK] Seq=554 Ack=213 Win=261888 Len=0
192.168.86.243 104.20.55.68 TLSv1.3 118 Change Cipher Spec, Application Data
192.168.86.243 104.20.55.68 TLSv1.3 146 Application Data
192.168.86.243 104.20.55.68 TLSv1.3 1270 Application Data
192.168.86.243 104.20.55.68 TLSv1.3 107 Application Data
104.20.55.68 192.168.86.243 TCP 60 443 → 60201 [ACK] Seq=213 Ack=618 Win=67584 Len=0
104.20.55.68 192.168.86.243 TCP 60 443 → 60201 [ACK] Seq=213 Ack=710 Win=67584 Len=0
104.20.55.68 192.168.86.243 TLSv1.3 575 Application Data, Application Data
192.168.86.243 104.20.55.68 TCP 54 60201 → 443 [ACK] Seq=1979 Ack=734 Win=261568 Len=0
192.168.86.243 104.20.55.68 TLSv1.3 85 Application Data
Type here to search
12:05 AM
10/5/2025

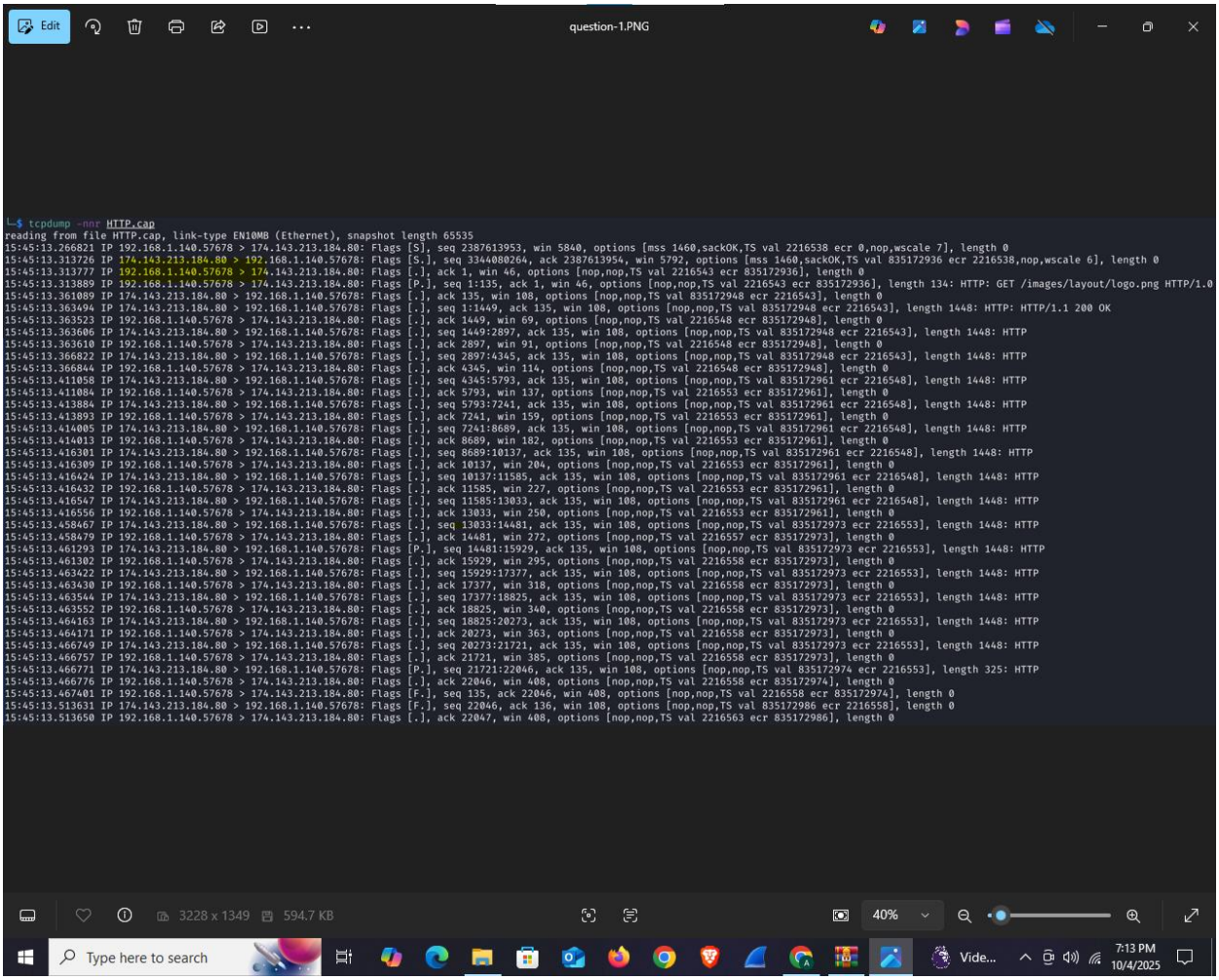


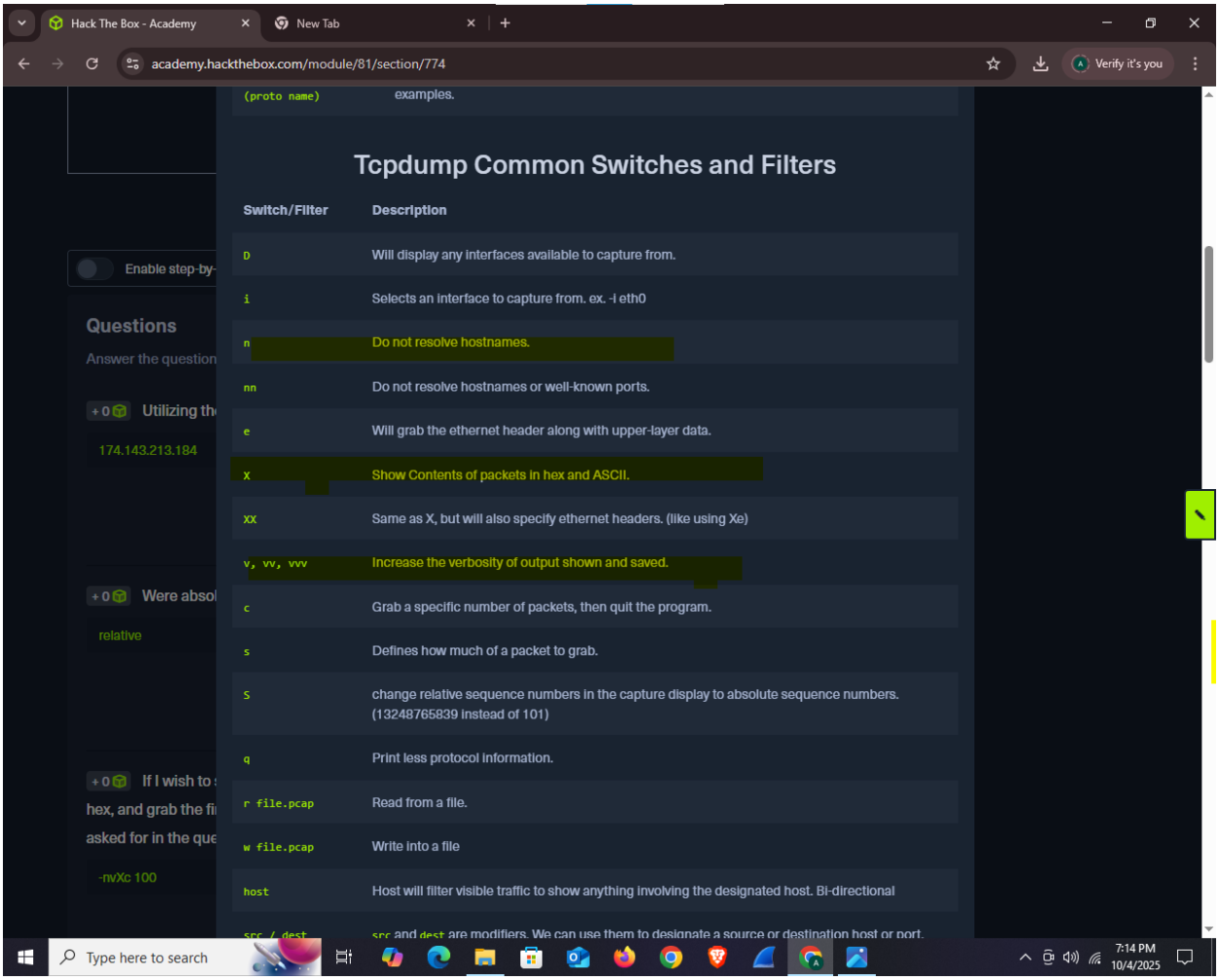
Question 3. TCPDUMP fundamentals.

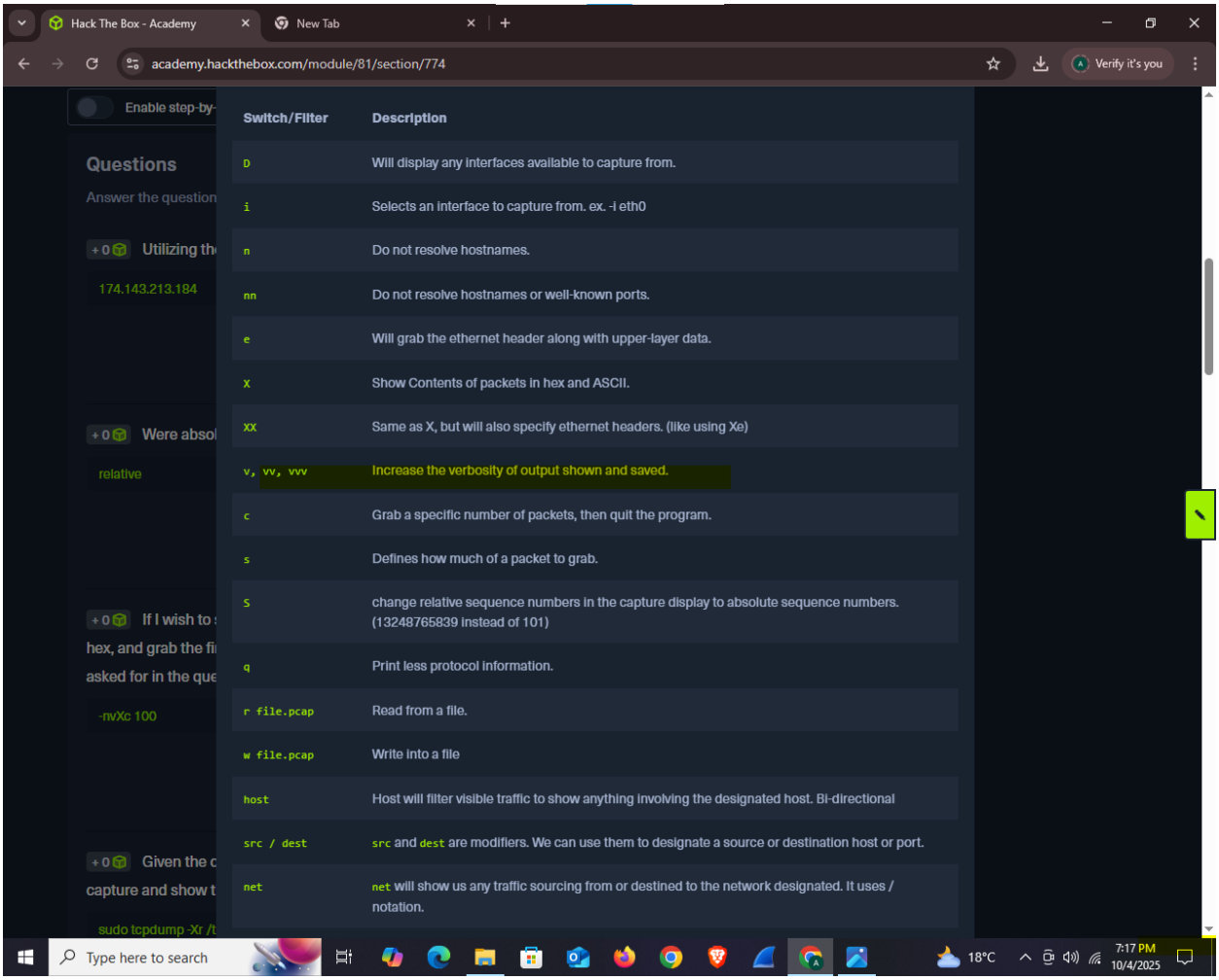

Hack The Box - Academy
TCP handshake port num
CNS3-2025: Assignment
HTB Academy - Intro To
21 Savage - Bank Accou
academy.hackthebox.com/module/81/section/774
Verify it's you
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
+0 Utilizing the output shown in question-1.png, who is the server in this communication? (IP Address)
174.143.213.184
Submit
question-1.zip
Hint
+0 Were absolute or relative sequence numbers used during the capture? (see question-1.zip to answer)
relative
Submit
question-1.zip
Hint
+0 If I wish to start a capture without hostname resolution, verbose output, showing contents in ASCII and hex, and grab the first 100 packets; what are the switches used? please answer in the order the switches are asked for in the question.
-nvXc 100
Submit
Hint
+0 Given the capture file at /tmp/capture.pcap, what tcpdump command will enable you to read from the capture and show the output contents in Hex and ASCII? (Please use best practices when using switches)
Integrated Terminal
Type here to search
Rain...
3:09 PM
10/3/2025


Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/774
Verify it's you
Submit
Hint
+0 What TCPDump switch will increase the verbosity of our output? (Include the - with the proper switch)
-v
Submit
Hint
+0 What built in terminal help reference can tell us more about TCPDump?
man
Submit
Hint
+0 What TCPDump switch will let me write my output to a file?
-w
Submit
Hint
Previous
Next
Mark Complete & Next
Powered by HACKTHEBOX
Integrated Terminal
Type here to search
Rain...
3:13 PM
10/3/2025

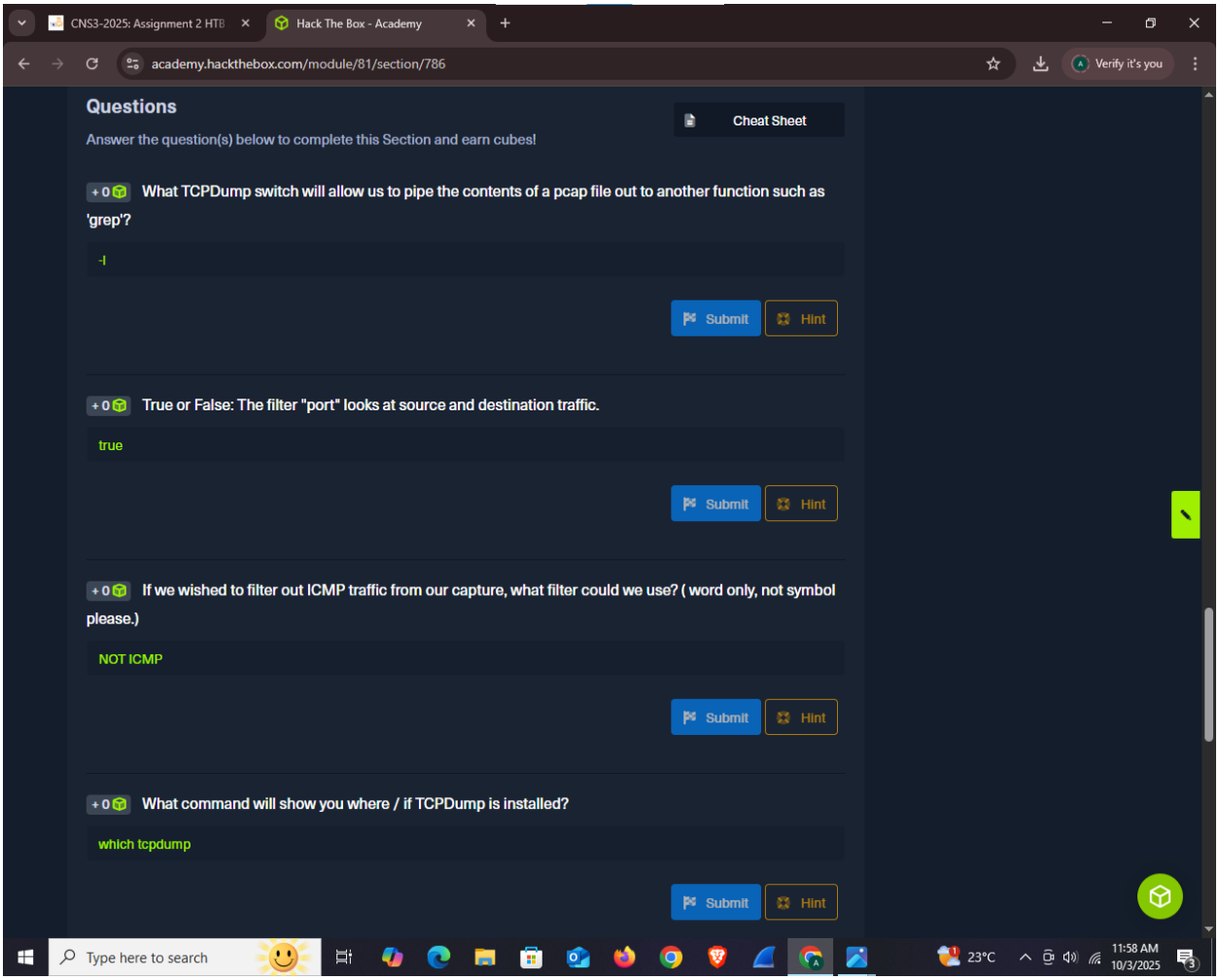
Proof.

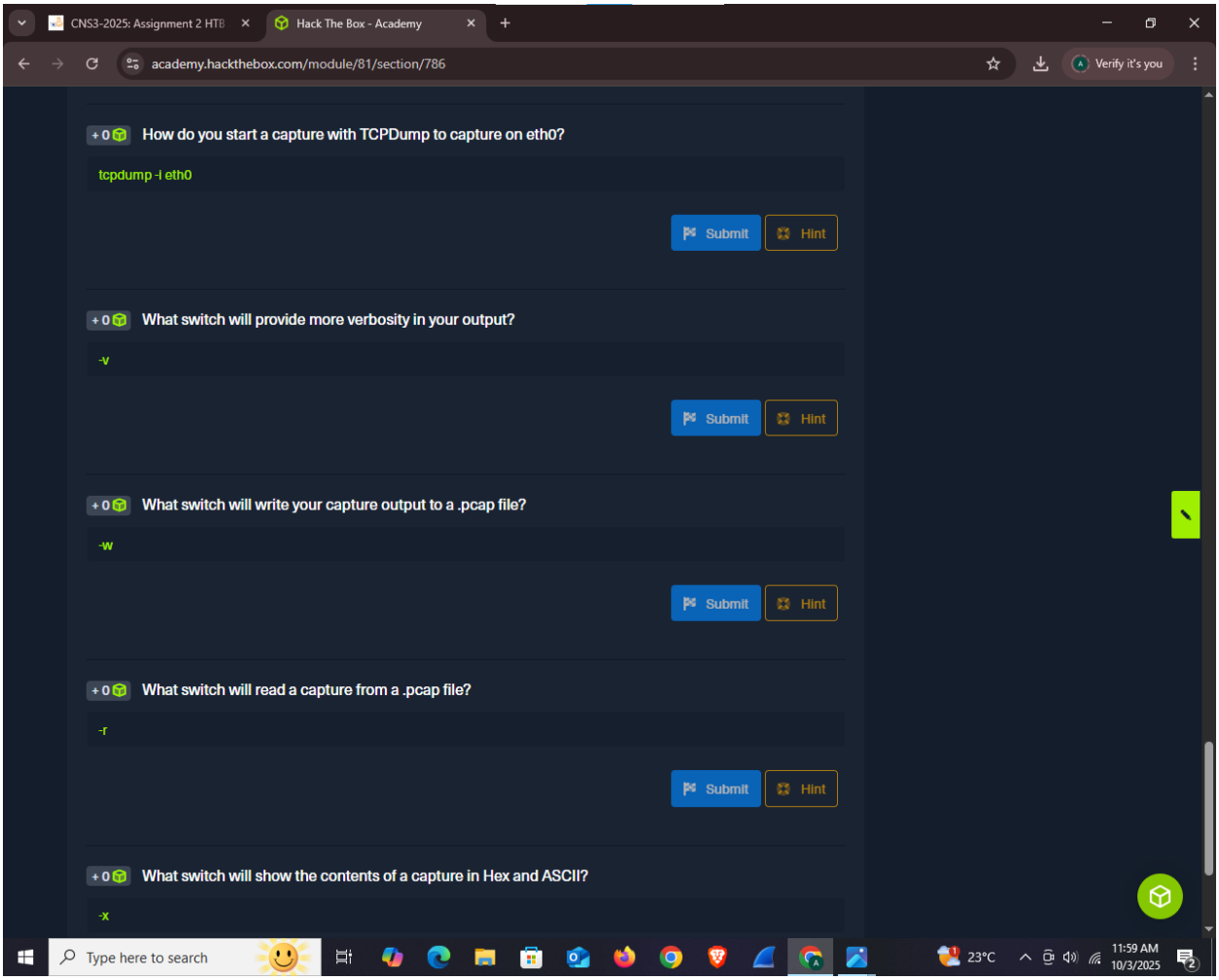

Edit
question-1.PNG
└─$ tcpdump -nnr HTTP.cap
reading from file HTTP.cap, link-type EN10MB (Ethernet), snapshot length 65535
15:45:13.266821 IP 192.168.1.140.57678 > 174.143.213.184.80: Flags [S], seq 2387613953, win 5840, options [mss 1460,sackOK,TS val 2216538 ecr 0,nop,wscale 7], length 0
15:45:13.313726 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
3228 x 1349
594.7 KB
40%
Type here to search
Vide...
7:13 PM
10/4/2025


Hack The Box - Academy
New Tab
academy.hackthebox.com/module/81/section/774
Verify it's you
(proto name) examples.
Tcpdump Common Switches and Filters
Switch/Filter
Description
D
Will display any interfaces available to capture from.
i
Selects an interface to capture from. ex. -i eth0
n
Do not resolve hostnames.
nn
Do not resolve hostnames or well-known ports.
e
Will grab the ethernet header along with upper-layer data.
X
Show Contents of packets in hex and ASCII.
XX
Same as X, but will also specify ethernet headers. (like using Xe)
v, vv, vvv
Increase the verbosity of output shown and saved.
c
Grab a specific number of packets, then quit the program.
s
Defines how much of a packet to grab.
S
change relative sequence numbers in the capture display to absolute sequence numbers. (13248765839 instead of 101)
q
Print less protocol information.
r file.pcap
Read from a file.
w file.pcap
Write into a file
host
Host will filter visible traffic to show anything involving the designated host. Bi-directional
src / dest
src and dest are modifiers. We can use them to designate a source or destination host or port.
Enable step-by-
Questions
Answer the question
+0 Utilizing th
174.143.213.184
+0 Were absol
relative
+0 If I wish to
hex, and grab the fi
asked for in the que
-nvXc 100
Type here to search
7:14 PM
10/4/2025

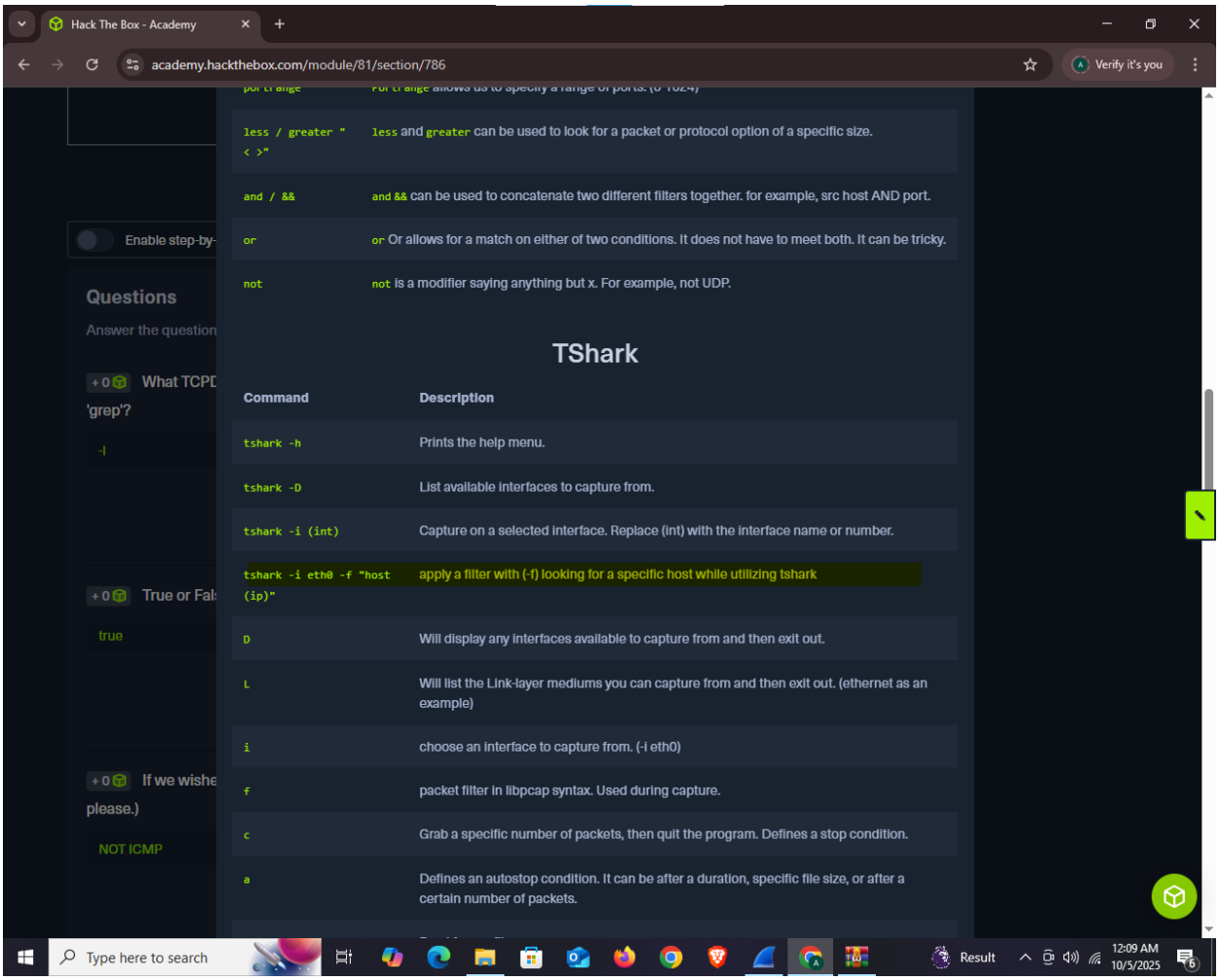


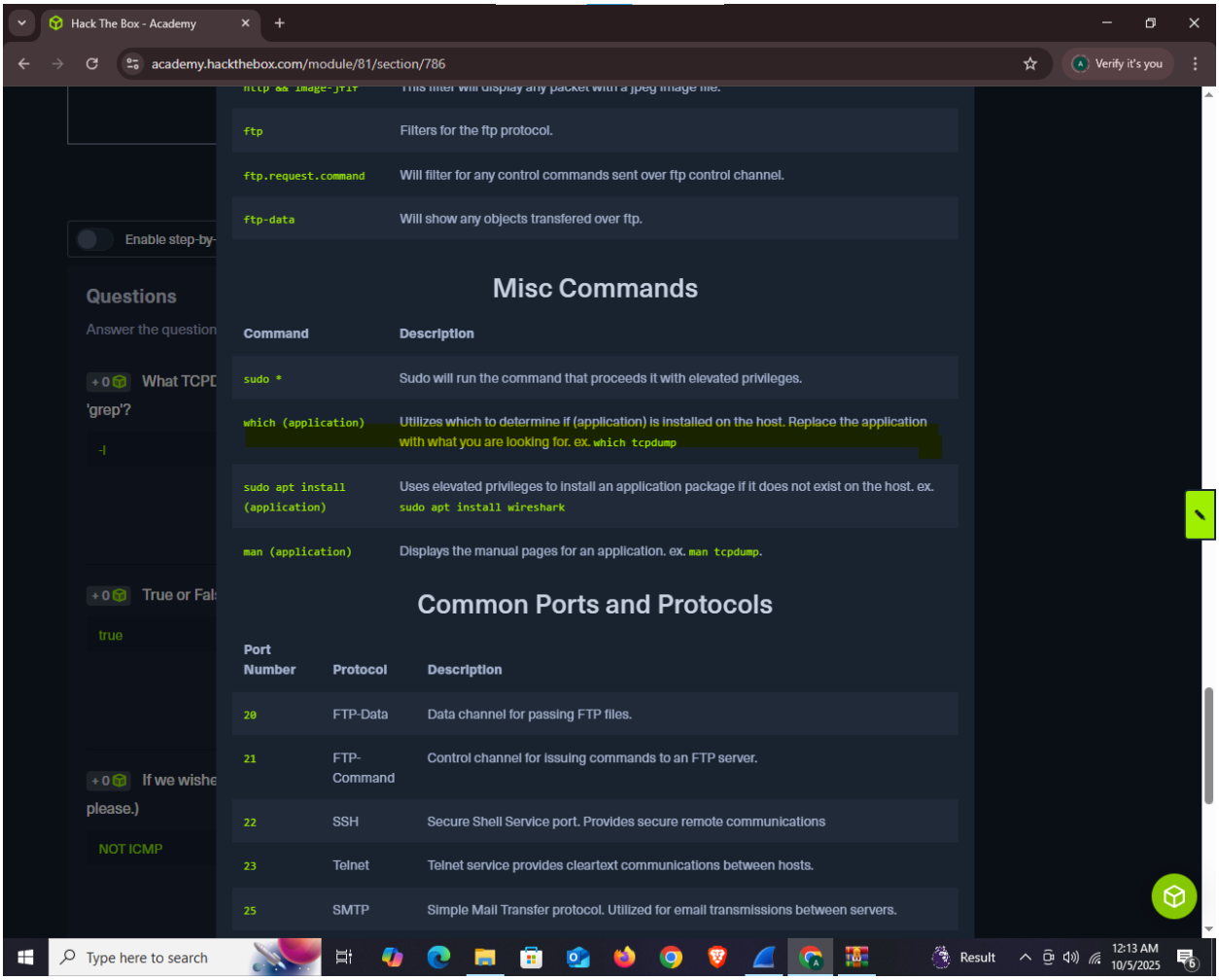
Question 4 capturing with tcpdump .

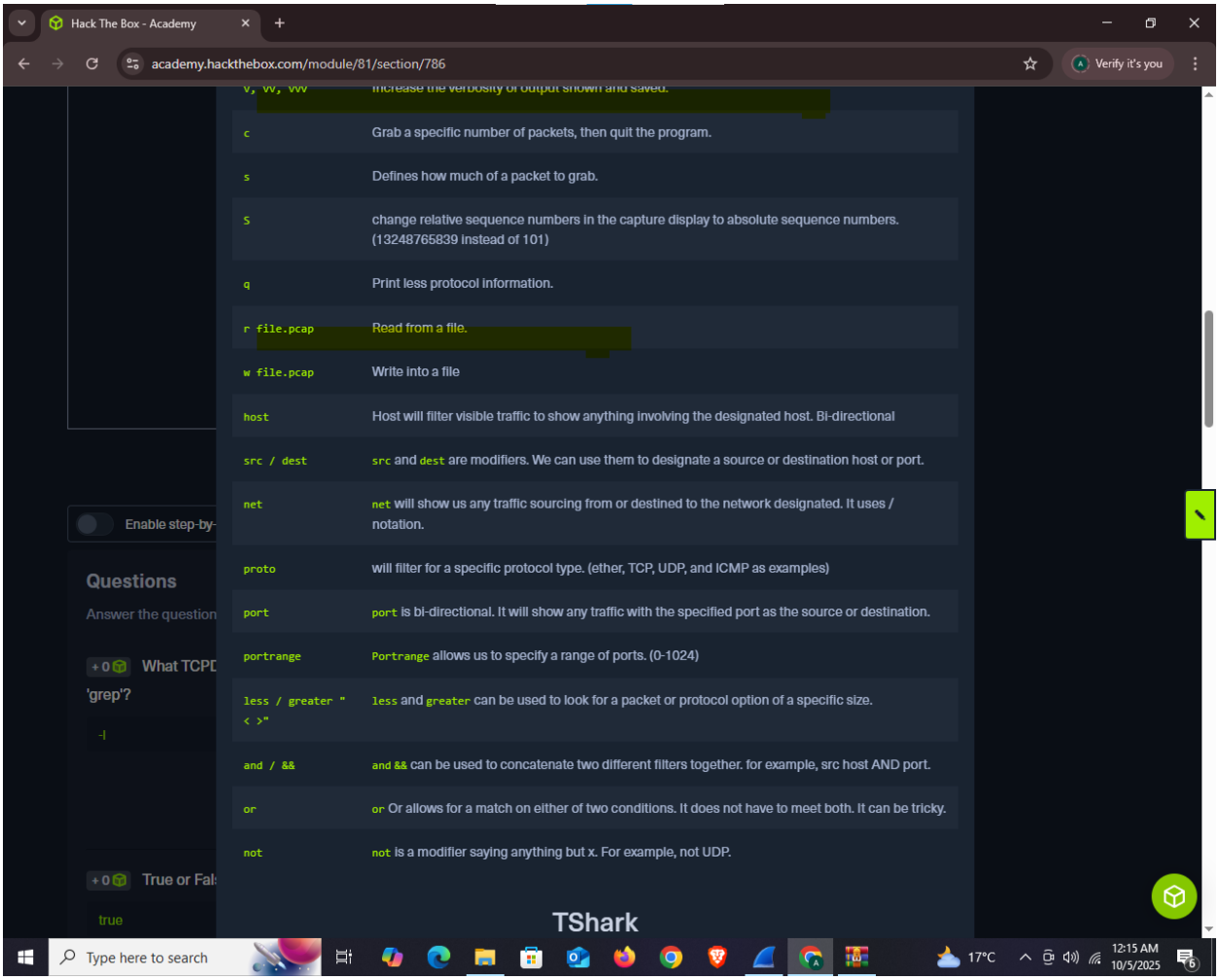

CNS3-2025: Assignment 2 HTB
Hack The Box - Academy
academy.hackthebox.com/module/81/section/786
Verify it's you
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
+0 What TCPDump switch will allow us to pipe the contents of a pcap file out to another function such as 'grep'?
-l
Submit
Hint
+0 True or False: The filter "port" looks at source and destination traffic.
true
Submit
Hint
+0 If we wished to filter out ICMP traffic from our capture, what filter could we use? (word only, not symbol please.)
NOT ICMP
Submit
Hint
+0 What command will show you where / if TCPDump is installed?
which tcpdump
Submit
Hint
Type here to search
23°C
11:58 AM
10/3/2025

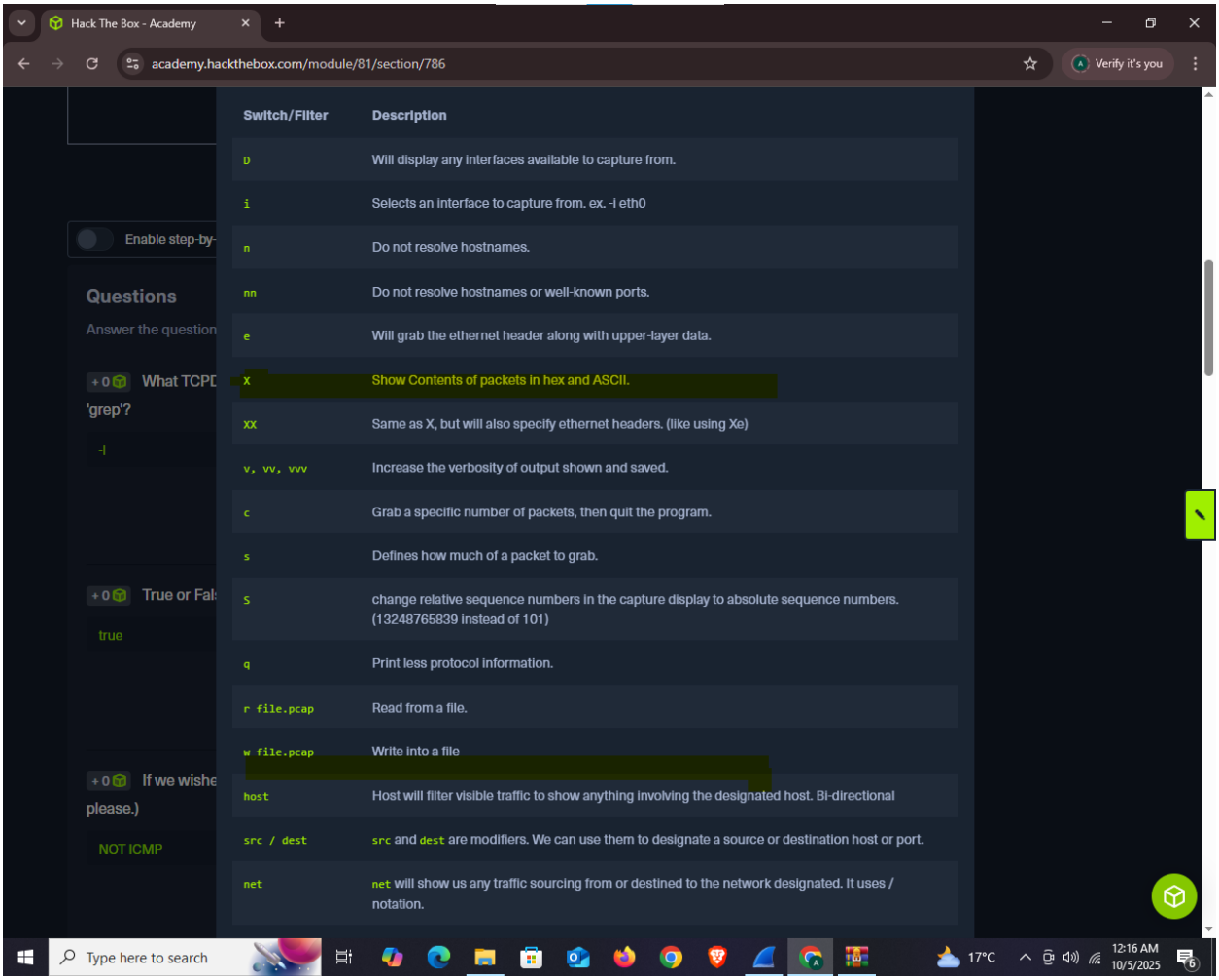

CNS3-2025: Assignment 2 HTB
Hack The Box - Academy
academy.hackthebox.com/module/81/section/786
Verify it's you
+0 How do you start a capture with TCPDump to capture on eth0?
tcpdump -i eth0
Submit
Hint
+0 What switch will provide more verbosity in your output?
-v
Submit
Hint
+0 What switch will write your capture output to a .pcap file?
-w
Submit
Hint
+0 What switch will read a capture from a .pcap file?
-r
Submit
Hint
+0 What switch will show the contents of a capture in Hex and ASCII?
-x
Type here to search
23°C
11:59 AM
10/3/2025

Proof.

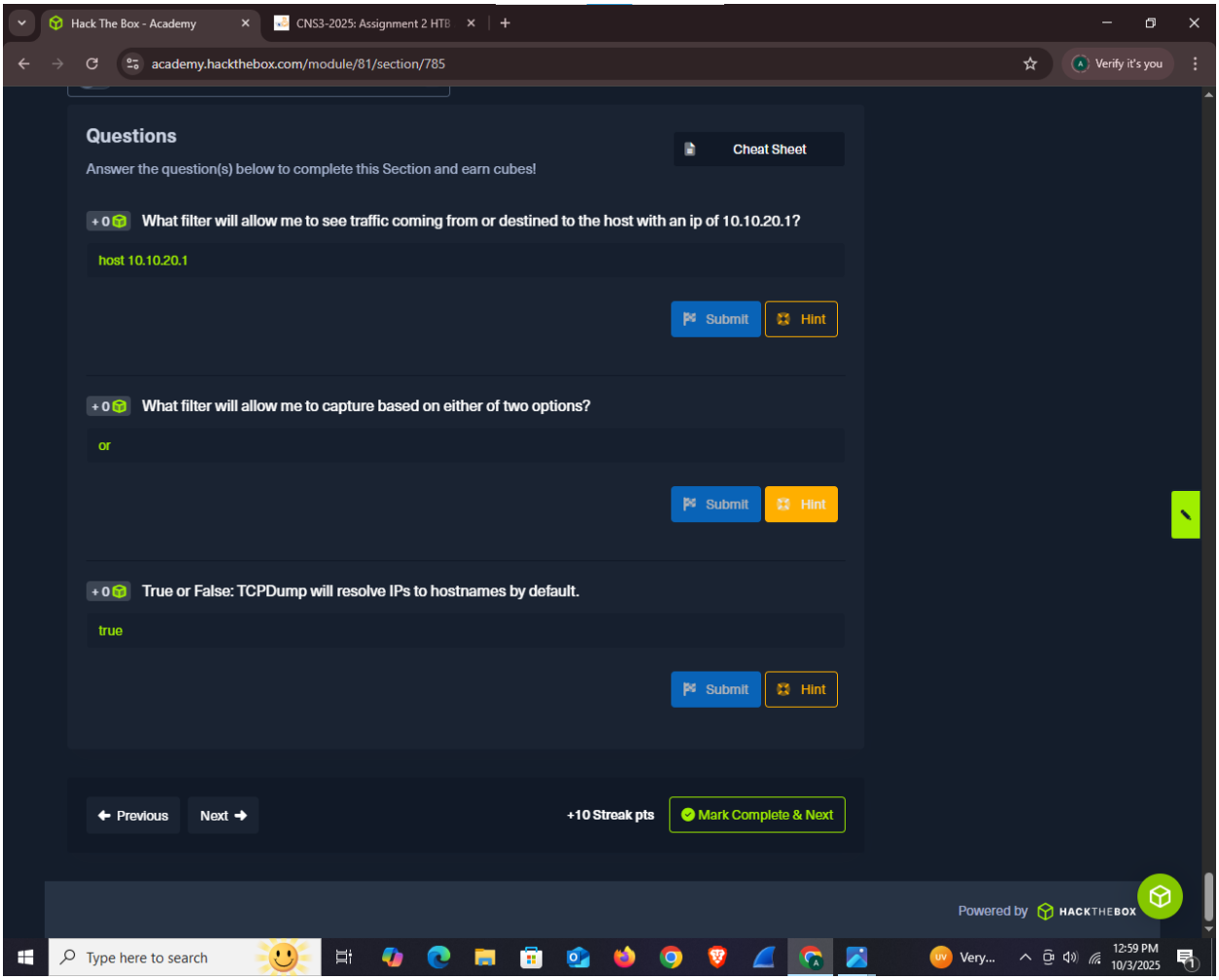

Hack The Box - Academy
academy.hackthebox.com/module/81/section/786
Verify it's you
less / greater " < >"
less and greater can be used to look for a packet or protocol option of a specific size.
and / &&
and && can be used to concatenate two different filters together. for example, src host AND port.
or
or Or allows for a match on either of two conditions. It does not have to meet both. It can be tricky.
not
not is a modifier saying anything but x. For example, not UDP.
TShark
Command
Description
tshark -h
Prints the help menu.
tshark -D
List available interfaces to capture from.
tshark -i (int)
Capture on a selected interface. Replace (int) with the interface name or number.
tshark -i eth0 -f "host (ip)"
apply a filter with (-f) looking for a specific host while utilizing tshark
D
Will display any interfaces available to capture from and then exit out.
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.
c
Grab a specific number of packets, then quit the program. Defines a stop condition.
a
Defines an autostop condition. It can be after a duration, specific file size, or after a certain number of packets.
Enable step-by-
Questions
Answer the question
What TCPD
'grep'?
-l
True or Fal
true
If we wishe
please.)
NOT ICMP
Type here to search
Result
12:09 AM
10/5/2025

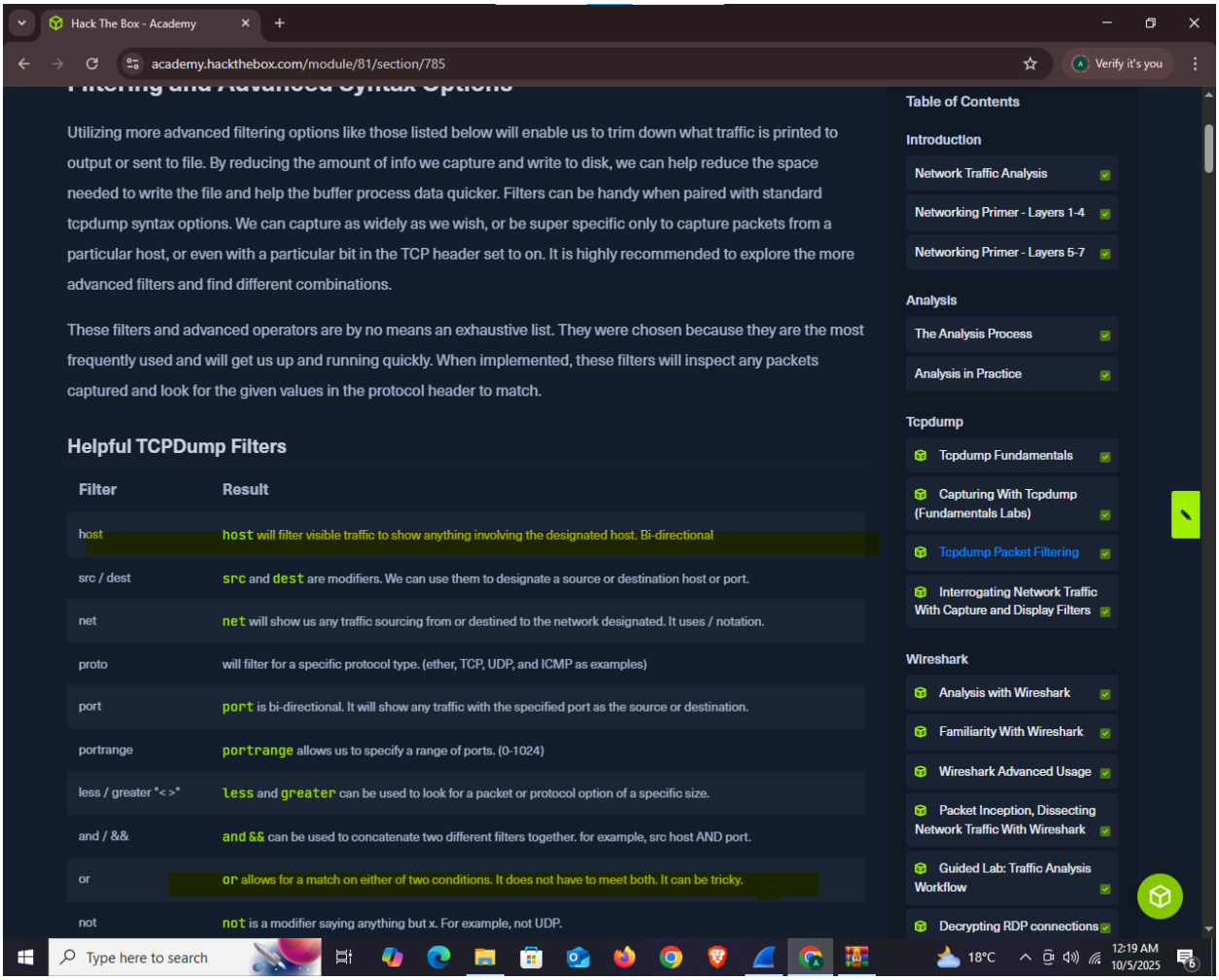

Hack The Box - Academy
academy.hackthebox.com/module/81/section/786
Verify it's you
ftp
Filters for the ftp protocol.
ftp.request.command
Will filter for any control commands sent over ftp control channel.
ftp-data
Will show any objects transfered over ftp.
Misc Commands
Command
Description
sudo *
Sudo will run the command that proceeds it with elevated privileges.
which (application)
Utilizes which to determine if (application) is installed on the host. Replace the application with what you are looking for. ex. which tcpdump
sudo apt install (application)
Uses elevated privileges to install an application package if it does not exist on the host. ex. sudo apt install wireshark
man (application)
Displays the manual pages for an application. ex. man tcpdump.
Common Ports and Protocols
Port Number
Protocol
Description
20
FTP-Data
Data channel for passing FTP files.
21
FTP-Command
Control channel for issuing commands to an FTP server.
22
SSH
Secure Shell Service port. Provides secure remote communications
23
Telnet
Telnet service provides cleartext communications between hosts.
25
SMTP
Simple Mail Transfer protocol. Utilized for email transmissions between servers.
Enable step-by-
Questions
Answer the question
What TCPD
'grep'?
-l
True or Fal
true
If we wishe
please.)
NOT ICMP
Type here to search
Result
12:13 AM
10/5/2025


Hack The Box - Academy
academy.hackthebox.com/module/81/section/786
Verify it's you
v, vv, vvv
Increase the verbosity of output shown and saved.
c
Grab a specific number of packets, then quit the program.
s
Defines how much of a packet to grab.
S
change relative sequence numbers in the capture display to absolute sequence numbers. (13248765839 instead of 101)
q
Print less protocol information.
r file.pcap
Read from a file.
w file.pcap
Write into a file
host
Host will filter visible traffic to show anything involving the designated host. Bi-directional
src / dest
src and dest are modifiers. We can use them to designate a source or destination host or port.
net
net will show us any traffic sourcing from or destined to the network designated. It uses / notation.
proto
will filter for a specific protocol type. (ether, TCP, UDP, and ICMP as examples)
port
port is bi-directional. It will show any traffic with the specified port as the source or destination.
portrange
Portrange allows us to specify a range of ports. (0-1024)
less / greater " < >"
less and greater can be used to look for a packet or protocol option of a specific size.
and / &&
and && can be used to concatenate two different filters together. for example, src host AND port.
or
or Or allows for a match on either of two conditions. It does not have to meet both. It can be tricky.
not
not is a modifier saying anything but x. For example, not UDP.
TShark
Enable step-by-
Questions
Answer the question
+0 What TCPD
'grep'?
-l
+0 True or Fal
true
Type here to search
17°C
12:15 AM
10/5/2025

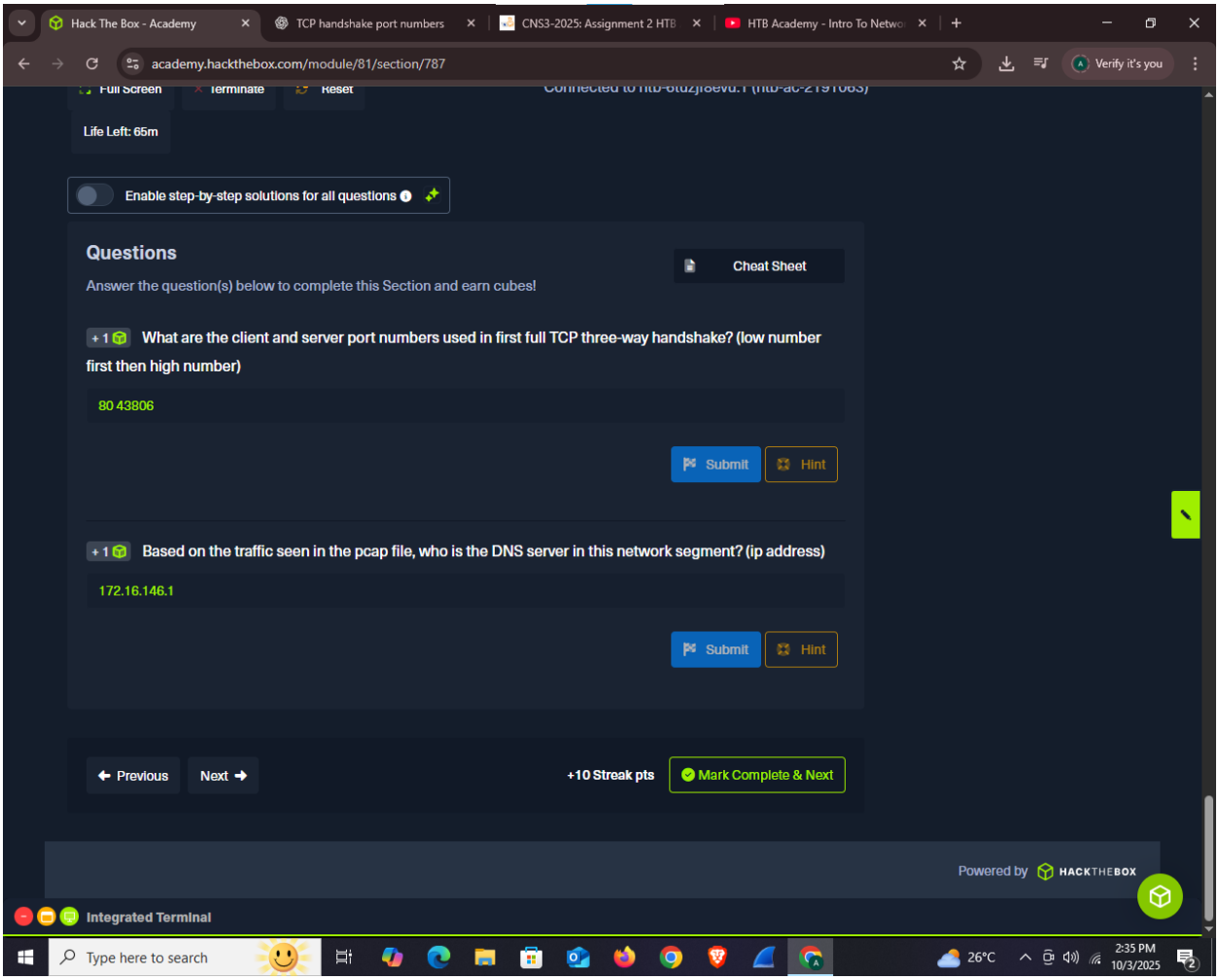


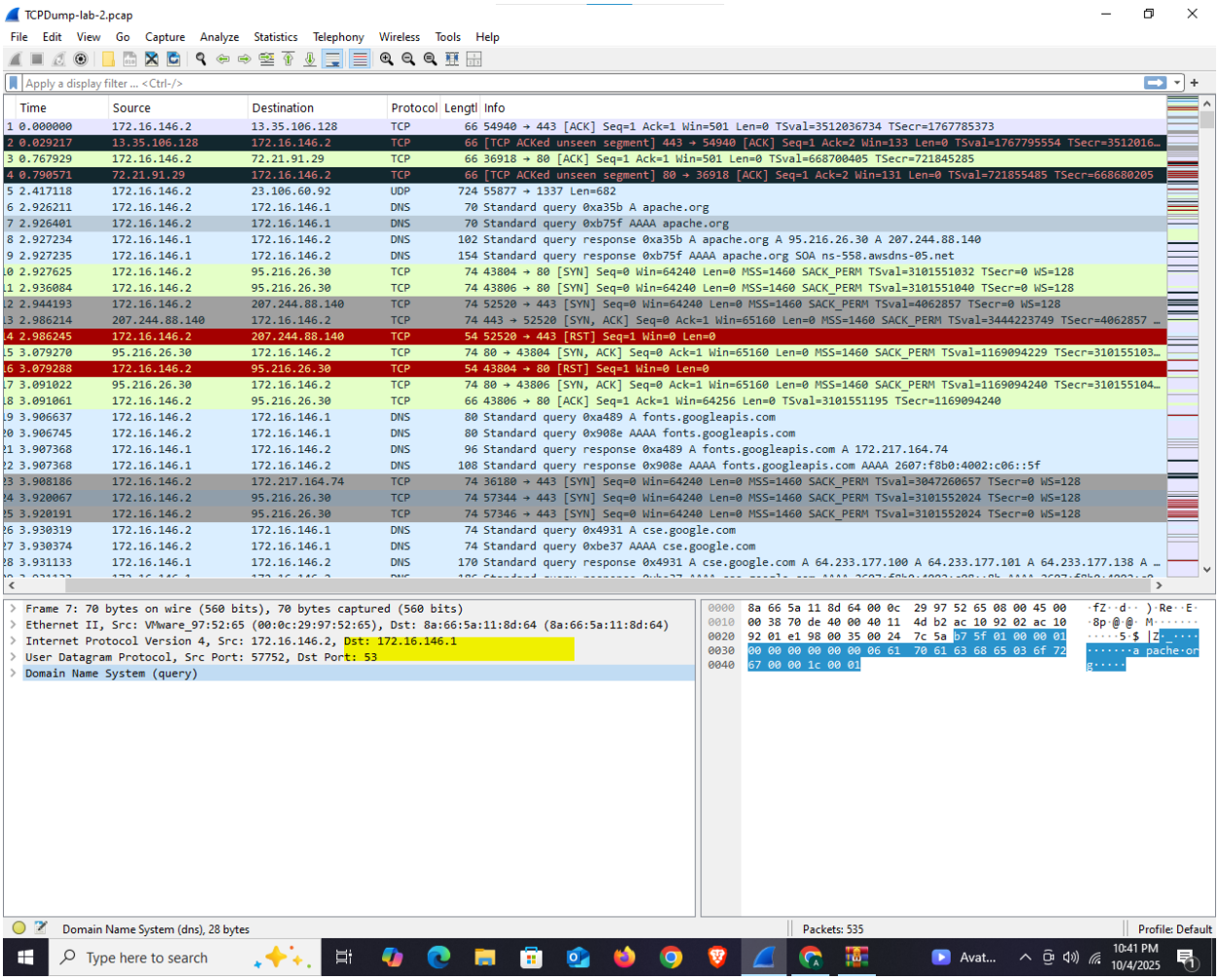
Question 5 . Tcpdump packet filtering.


Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/785
Verify it's you
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
+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
Submit
Hint
+0 What filter will allow me to capture based on either of two options?
or
Submit
Hint
+0 True or False: TCPDump will resolve IPs to hostnames by default.
true
Submit
Hint
Previous
Next
+10 Streak pts
Mark Complete & Next
Powered by HACKTHEBOX
Type here to search
Very...
12:59 PM
10/3/2025

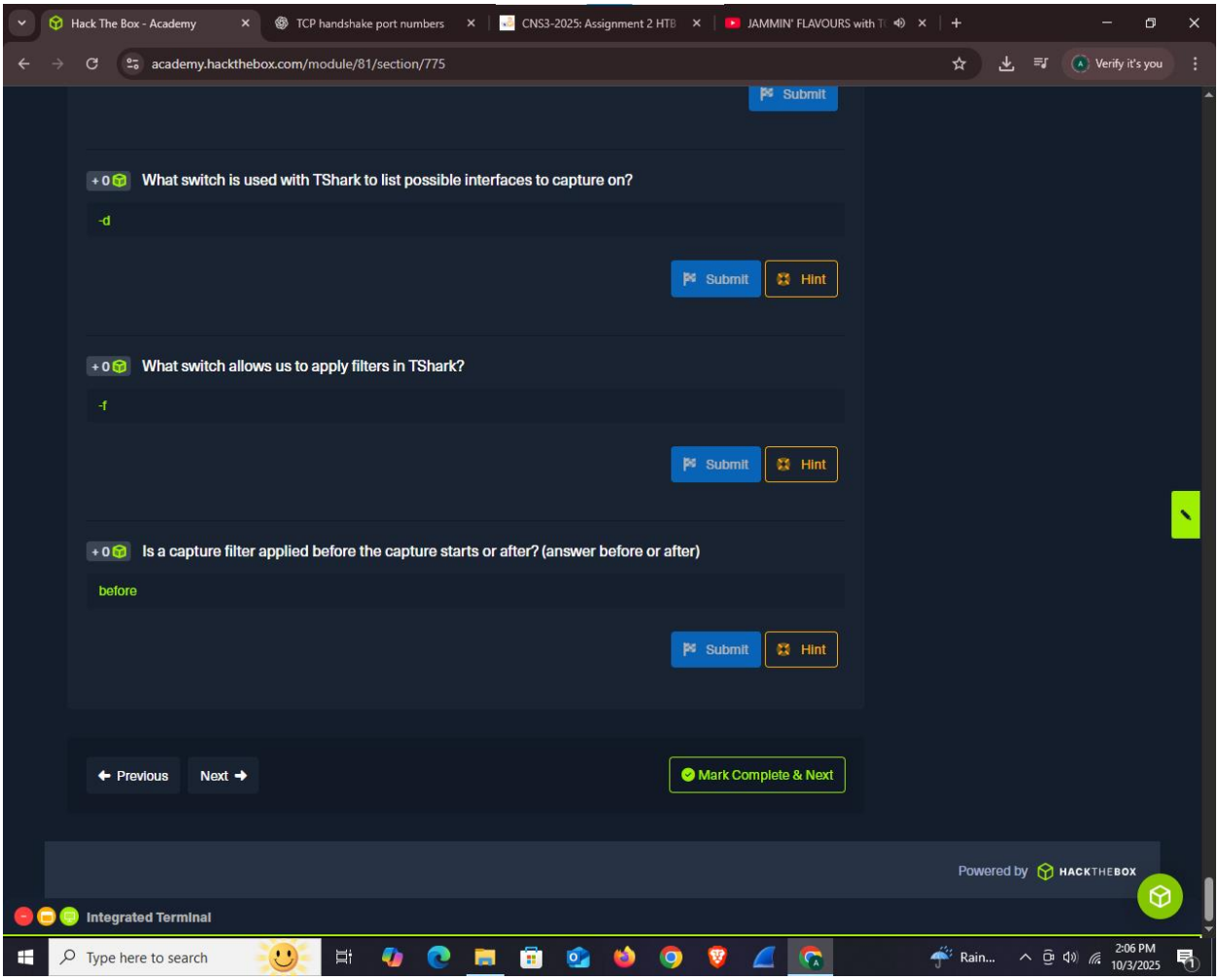


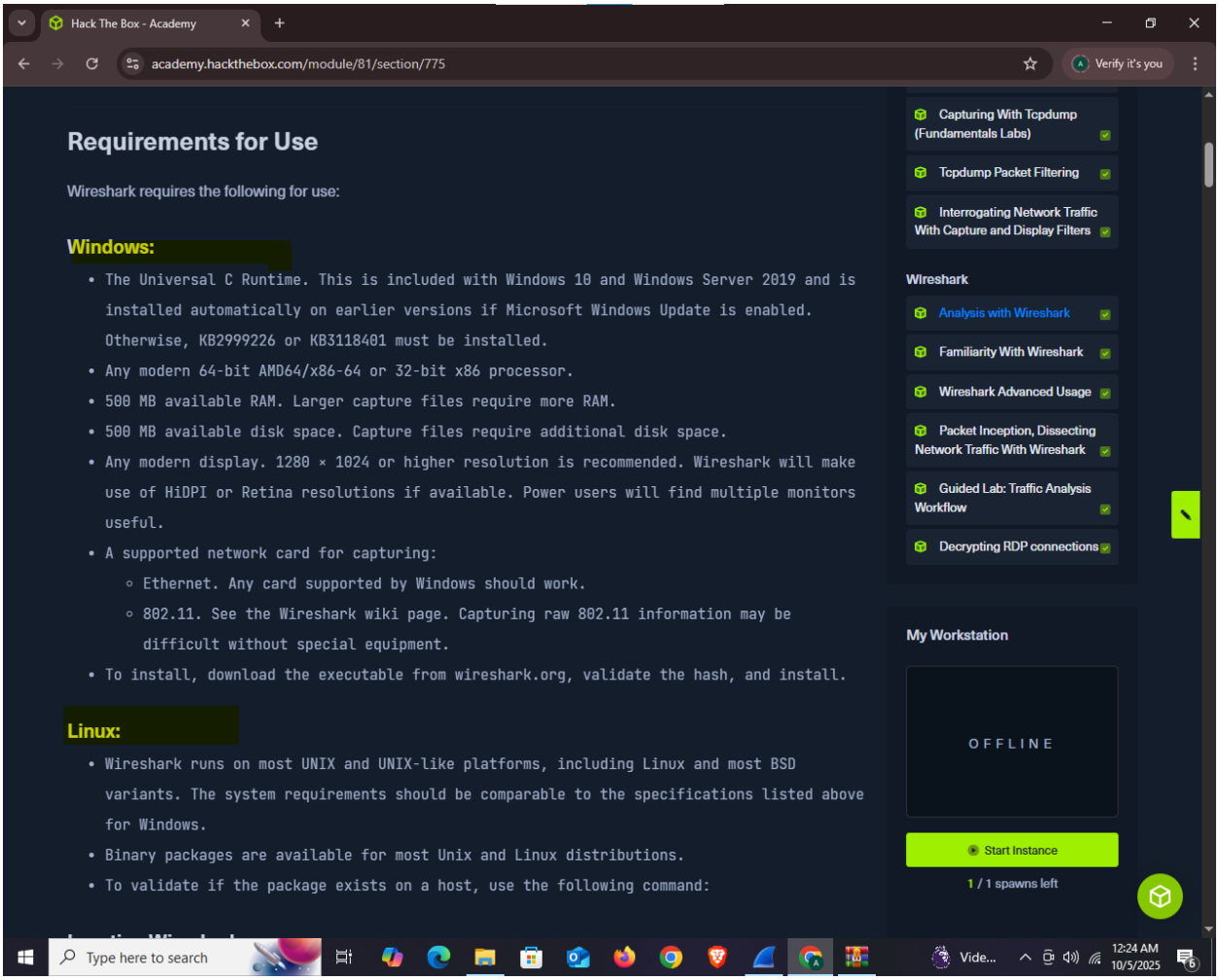
Question 6.Interogating Network Traffic with capture and display.

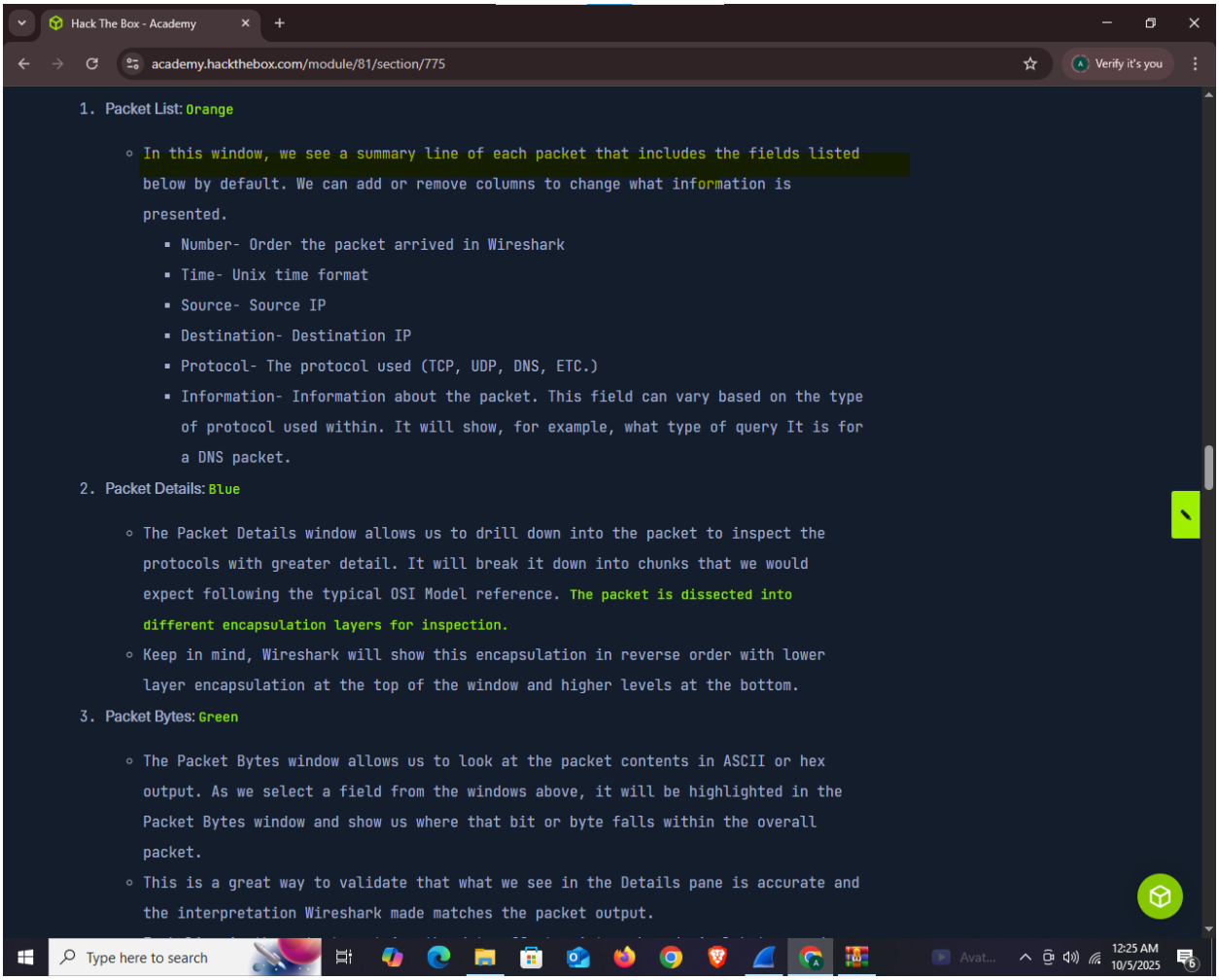

Hack The Box - Academy
TCP handshake port numbers
CNS3-2025: Assignment 2 HTB
HTB Academy - Intro To Netwo
academy.hackthebox.com/module/81/section/787
Verify it's you
Full Screen
Terminate
Reset
Life Left: 65m
Enable step-by-step solutions for all questions
Questions
Cheat Sheet
Answer the question(s) 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
Submit
Hint
+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
Submit
Hint
Previous
Next
+10 Streak pts
Mark Complete & Next
Powered by HACKTHEBOX
Integrated Terminal
Type here to search
26°C
2:35 PM
10/3/2025

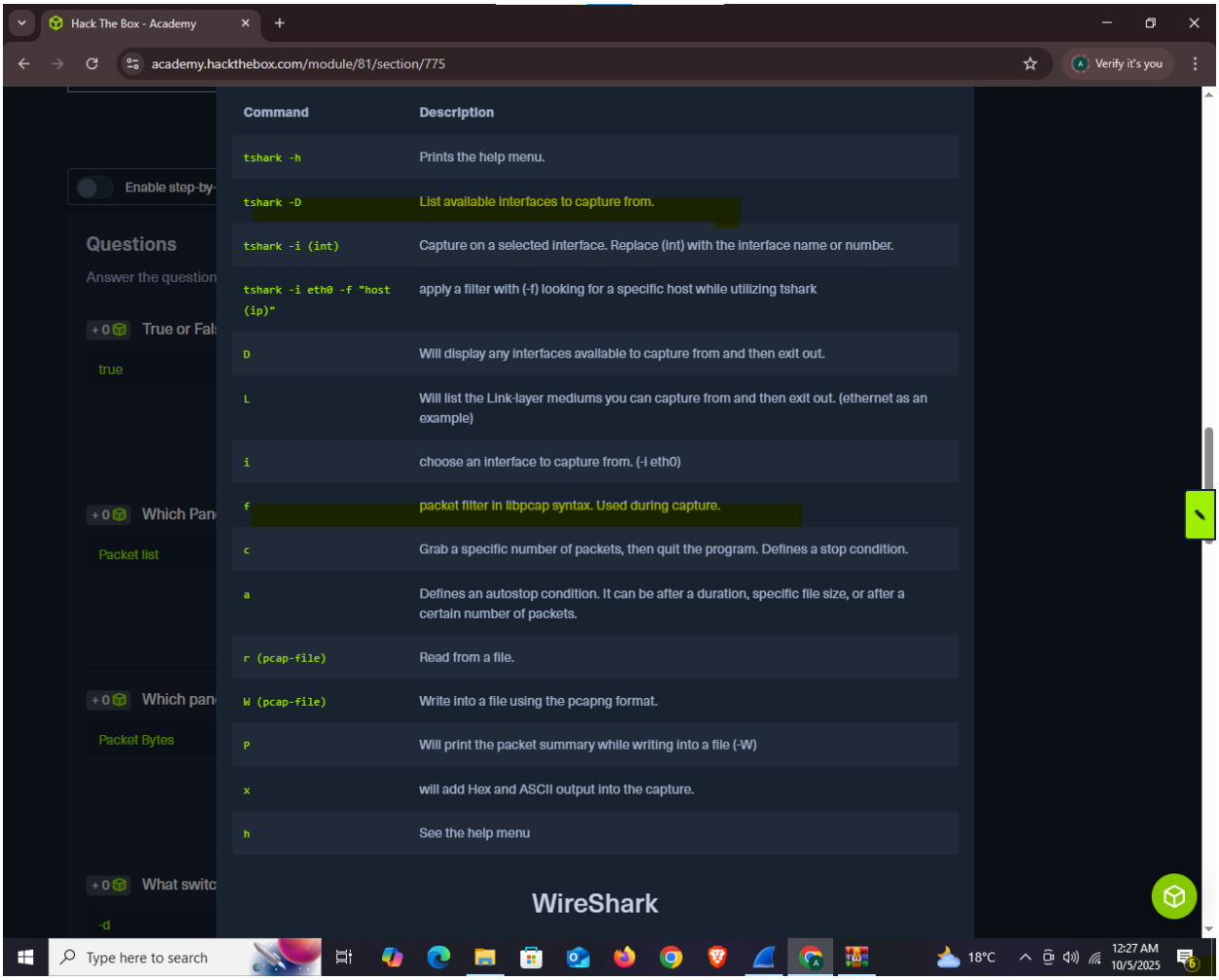


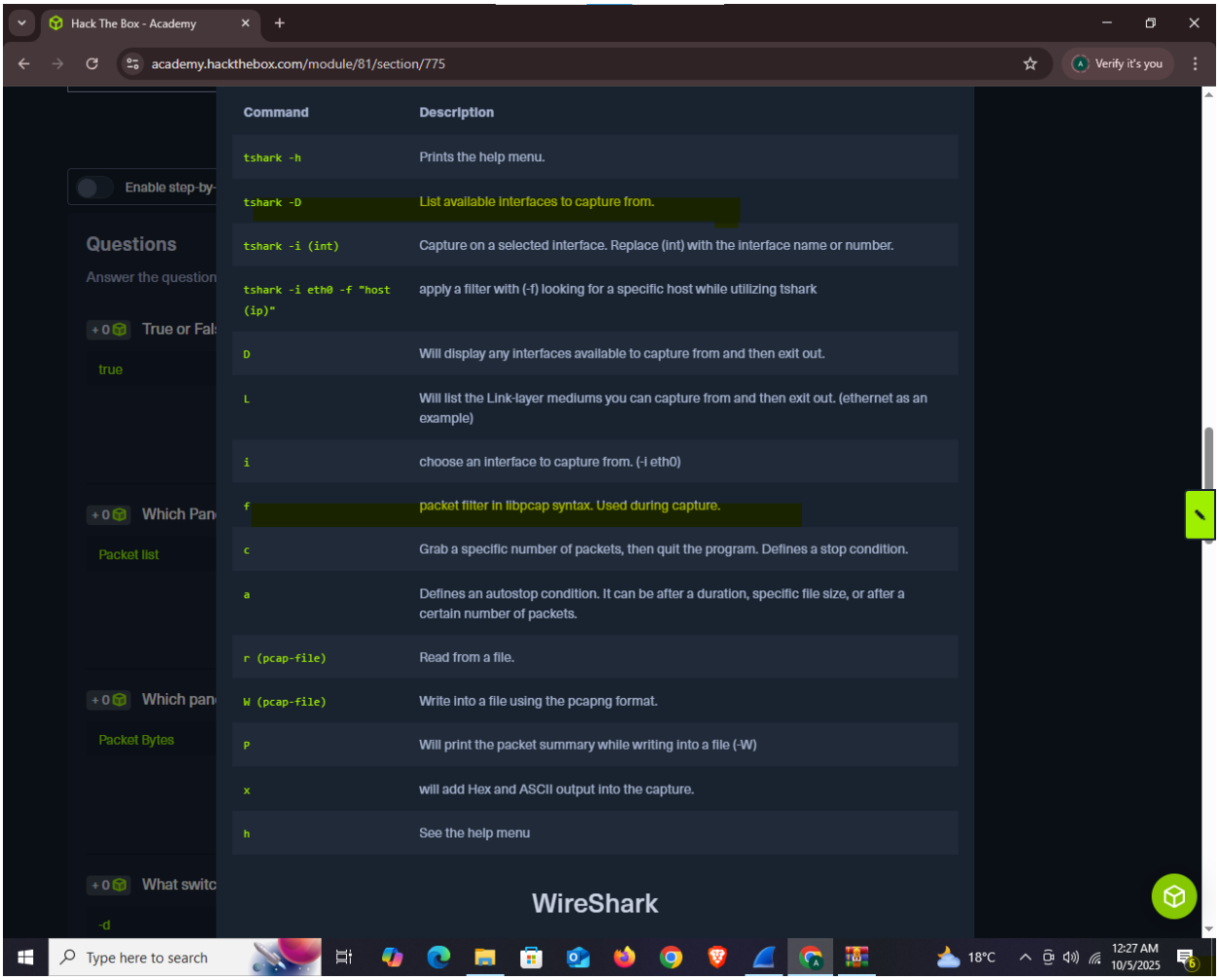
Question 7 .Analysis with wireshark.

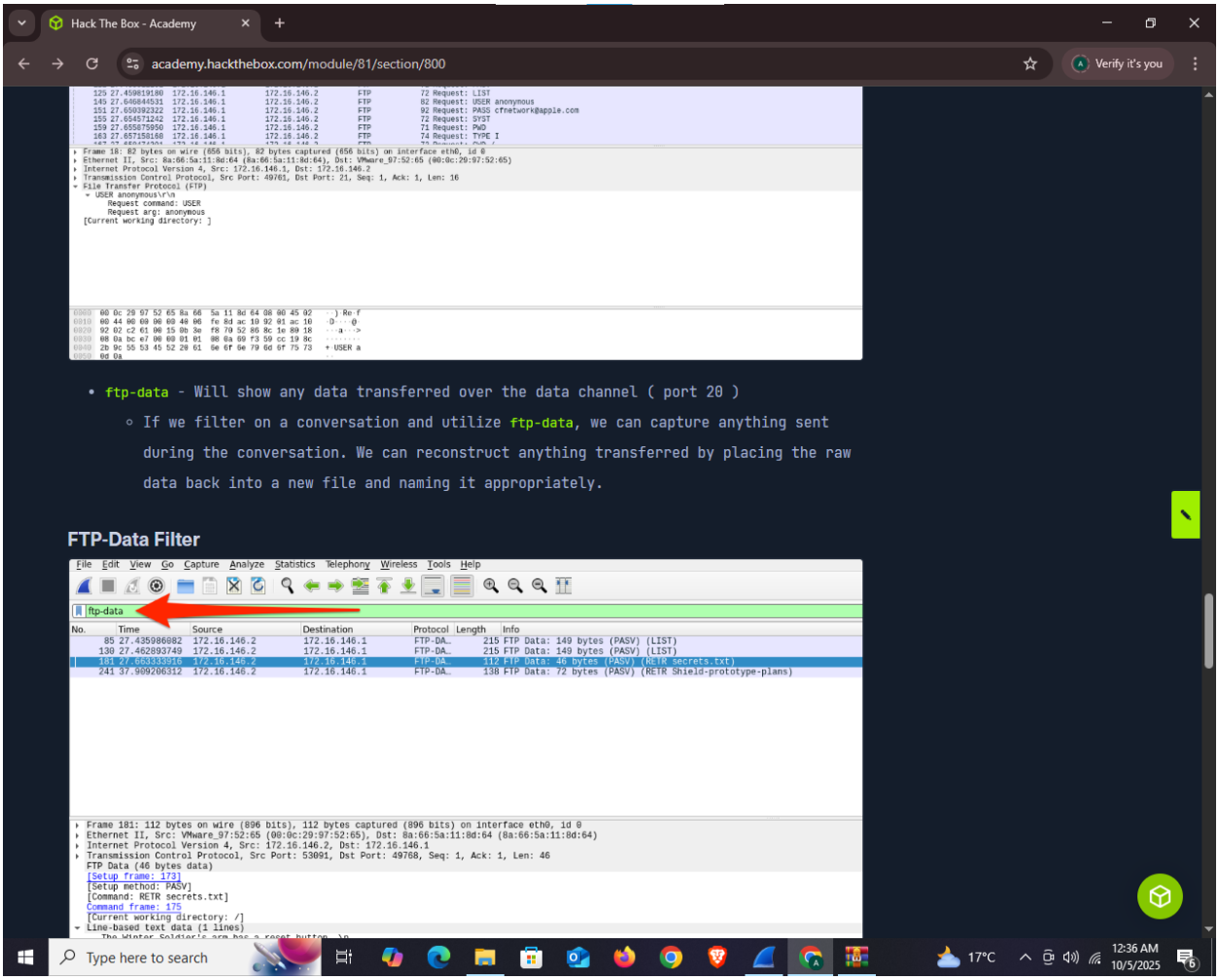

Hack The Box - Academy
TCP handshake port numbers
CNS3-2025: Assignment 2 HTB
JAMMIN' FLAVOURS with T
academy.hackthebox.com/module/81/section/775
Verify it's you
Submit
+0 What switch is used with TShark to list possible interfaces to capture on?
-d
Submit
Hint
+0 What switch allows us to apply filters in TShark?
-f
Submit
Hint
+0 Is a capture filter applied before the capture starts or after? (answer before or after)
before
Submit
Hint
Previous
Next
Mark Complete & Next
Powered by HACKTHEBOX
Integrated Terminal
Type here to search
Rain...
2:06 PM
10/3/2025


Hack The Box - Academy
academy.hackthebox.com/module/81/section/775
Verify it's you
Requirements for Use
Wireshark requires the following for use:
Windows:
The Universal C Runtime. This is included with Windows 10 and Windows Server 2019 and is installed automatically on earlier versions if Microsoft Windows Update is enabled. Otherwise, KB2999226 or KB3118401 must be installed.
Any modern 64-bit AMD64/x86-64 or 32-bit x86 processor.
500 MB available RAM. Larger capture files require more RAM.
500 MB available disk space. Capture files require additional disk space.
Any modern display. 1280 × 1024 or higher resolution is recommended. Wireshark will make use of HiDPI or Retina resolutions if available. Power users will find multiple monitors useful.
A supported network card for capturing:
Ethernet. Any card supported by Windows should work.
802.11. See the Wireshark wiki page. Capturing raw 802.11 information may be difficult without special equipment.
To install, download the executable from wireshark.org, validate the hash, and install.
Linux:
Wireshark runs on most UNIX and UNIX-like platforms, including Linux and most BSD variants. The system requirements should be comparable to the specifications listed above for Windows.
Binary packages are available for most Unix and Linux distributions.
To validate if the package exists on a host, use the following command:
Capturing With Tcpdump (Fundamentals Labs)
Tcpdump Packet Filtering
Interrogating Network Traffic With Capture and Display Filters
Wireshark
Analysis with Wireshark
Familiarity With Wireshark
Wireshark Advanced Usage
Packet Inception, Dissecting Network Traffic With Wireshark
Guided Lab: Traffic Analysis Workflow
Decrypting RDP connections
My Workstation
OFFLINE
Start Instance
1 / 1 spawns left
Type here to search
12:24 AM
10/5/2025

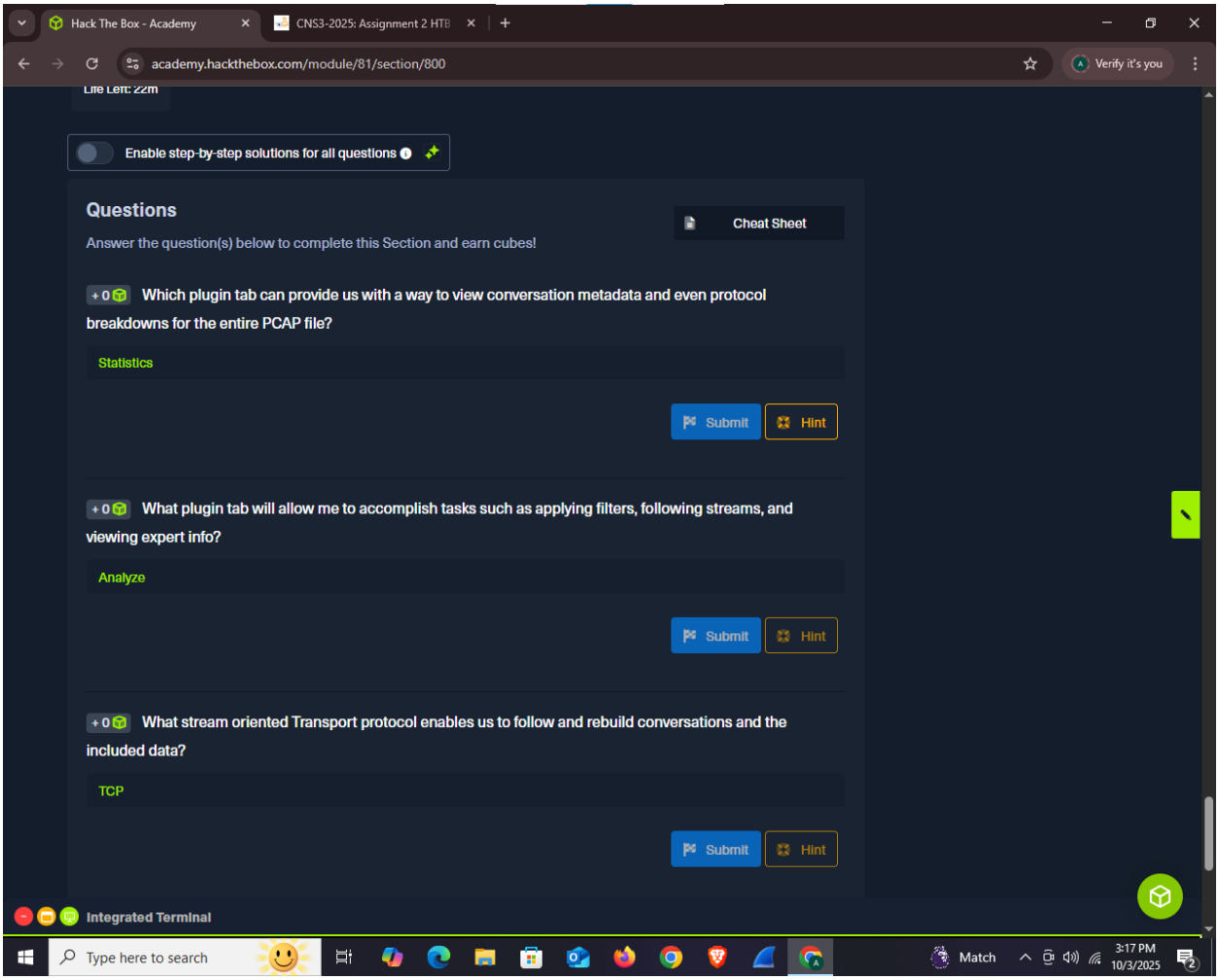

Hack The Box - Academy
academy.hackthebox.com/module/81/section/775
Verify it's you
1. Packet List: Orange
In this window, we see a summary line of each packet that includes the fields listed below by default. We can add or remove columns to change what information is presented.
Number- Order the packet arrived in Wireshark
Time- Unix time format
Source- Source IP
Destination- Destination IP
Protocol- The protocol used (TCP, UDP, DNS, ETC.)
Information- Information about the packet. This field can vary based on the type of protocol used within. It will show, for example, what type of query It is for a DNS packet.
2. Packet Details: Blue
The Packet Details 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 is dissected into different encapsulation layers for inspection.
Keep in mind, Wireshark will show this encapsulation in reverse order with lower layer encapsulation at the top of the window and higher levels at the bottom.
3. Packet Bytes: Green
The Packet Bytes 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.
This is a great way to validate that what we see in the Details pane is accurate and the interpretation Wireshark made matches the packet output.
Type here to search
12:25 AM
10/5/2025

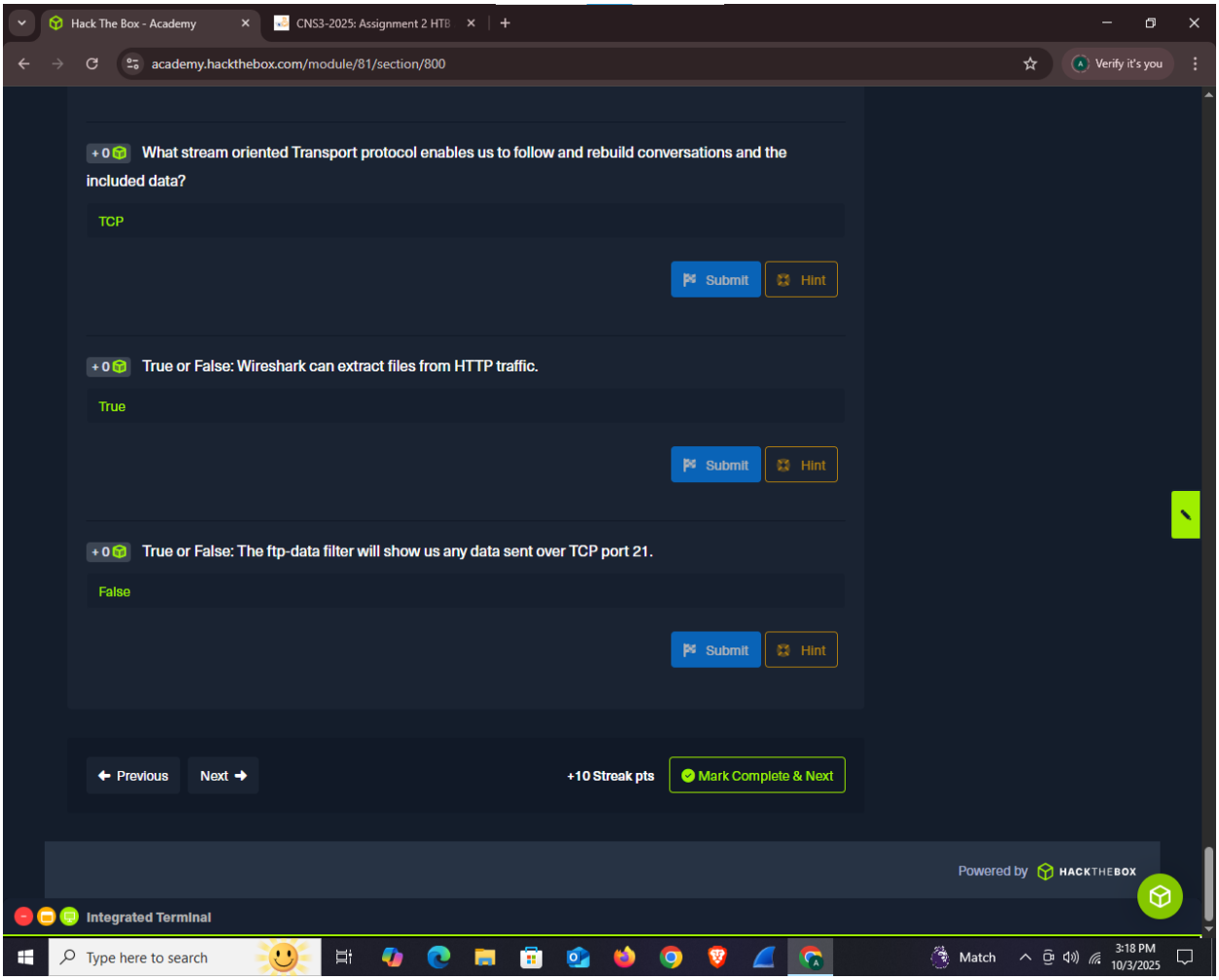

Hack The Box - Academy
academy.hackthebox.com/module/81/section/775
Verify it's you
Enable step-by-
Questions
Answer the question
+0 True or Fal
true
+0 Which Pan
Packet list
+0 Which pan
Packet Bytes
+0 What switc
-d
Command
Description
tshark -h
Prints the help menu.
tshark -D
List available interfaces to capture from.
tshark -i (int)
Capture on a selected interface. Replace (int) with the interface name or number.
tshark -i eth0 -f "host (ip)"
apply a filter with (-f) looking for a specific host while utilizing tshark
D
Will display any interfaces available to capture from and then exit out.
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.
c
Grab a specific number of packets, then quit the program. Defines a stop condition.
a
Defines an autostop condition. It can be after a duration, specific file size, or after a certain number of packets.
r (pcap-file)
Read from a file.
W (pcap-file)
Write into a file using the pcapng format.
P
Will print the packet summary while writing into a file (-W)
x
will add Hex and ASCII output into the capture.
h
See the help menu
WireShark
Type here to search
18°C
12:27 AM
10/5/2025

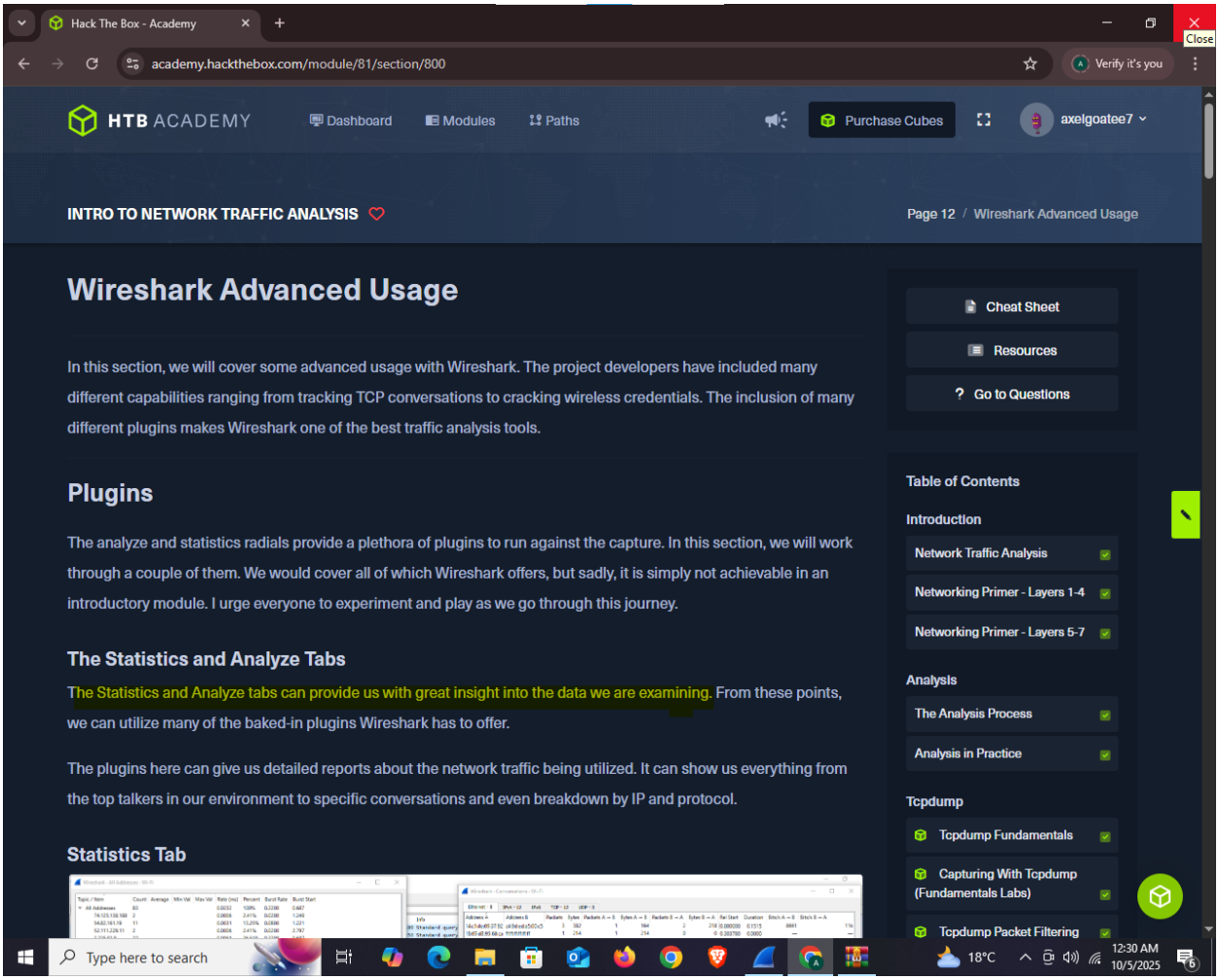

Hack The Box - Academy
academy.hackthebox.com/module/81/section/775
Verify it's you
Enable step-by-
Questions
Answer the question
+0 True or Fal
true
+0 Which Pan
Packet list
+0 Which pan
Packet Bytes
+0 What switc
-d
Command
Description
tshark -h
Prints the help menu.
tshark -D
List available interfaces to capture from.
tshark -i (int)
Capture on a selected interface. Replace (int) with the interface name or number.
tshark -i eth0 -f "host (ip)"
apply a filter with (-f) looking for a specific host while utilizing tshark
D
Will display any interfaces available to capture from and then exit out.
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.
c
Grab a specific number of packets, then quit the program. Defines a stop condition.
a
Defines an autostop condition. It can be after a duration, specific file size, or after a certain number of packets.
r (pcap-file)
Read from a file.
W (pcap-file)
Write into a file using the pcapng format.
P
Will print the packet summary while writing into a file (-W)
x
will add Hex and ASCII output into the capture.
h
See the help menu
WireShark
Type here to search
18°C
12:27 AM
10/5/2025



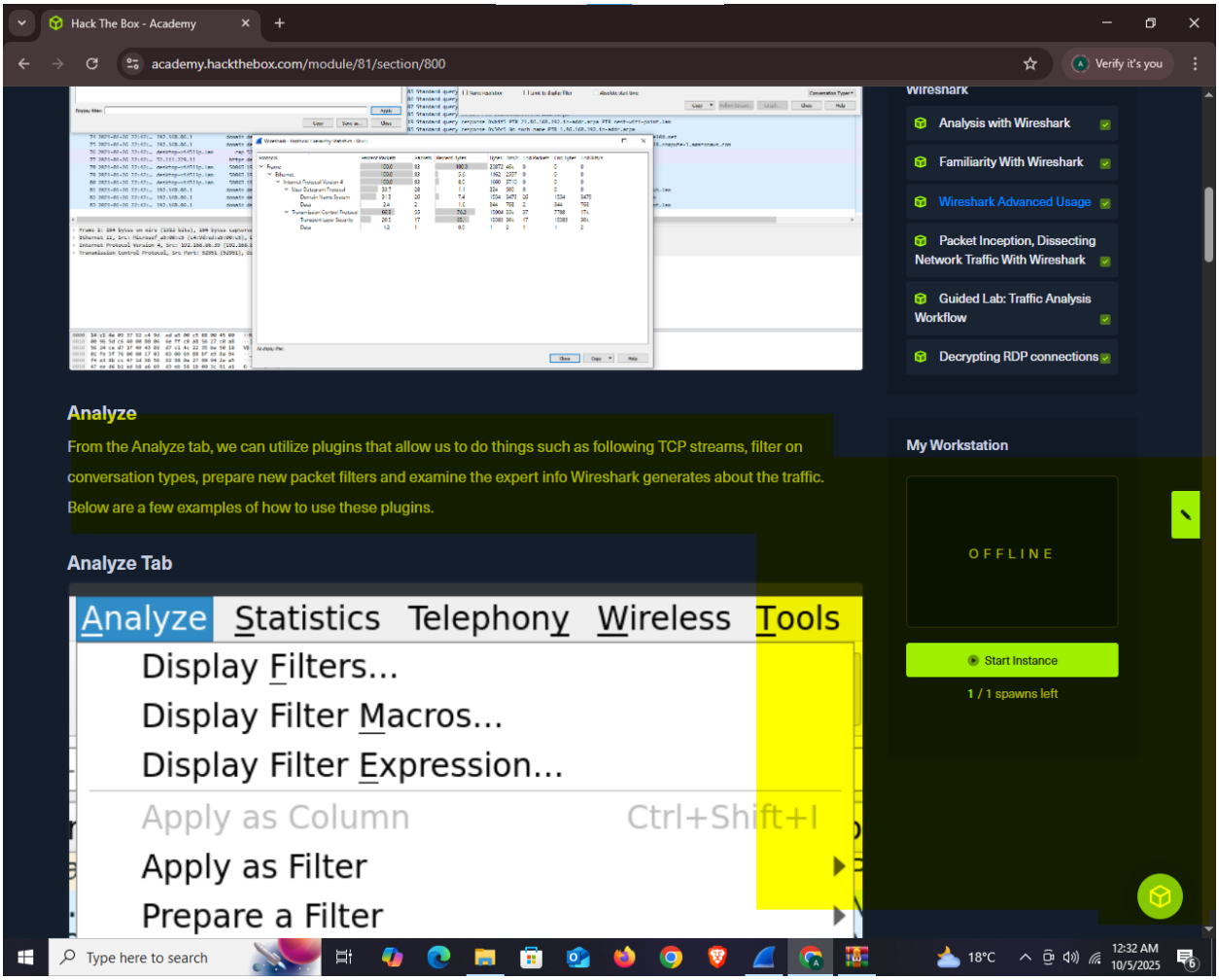
Question 8 Wireshark advanced usage.

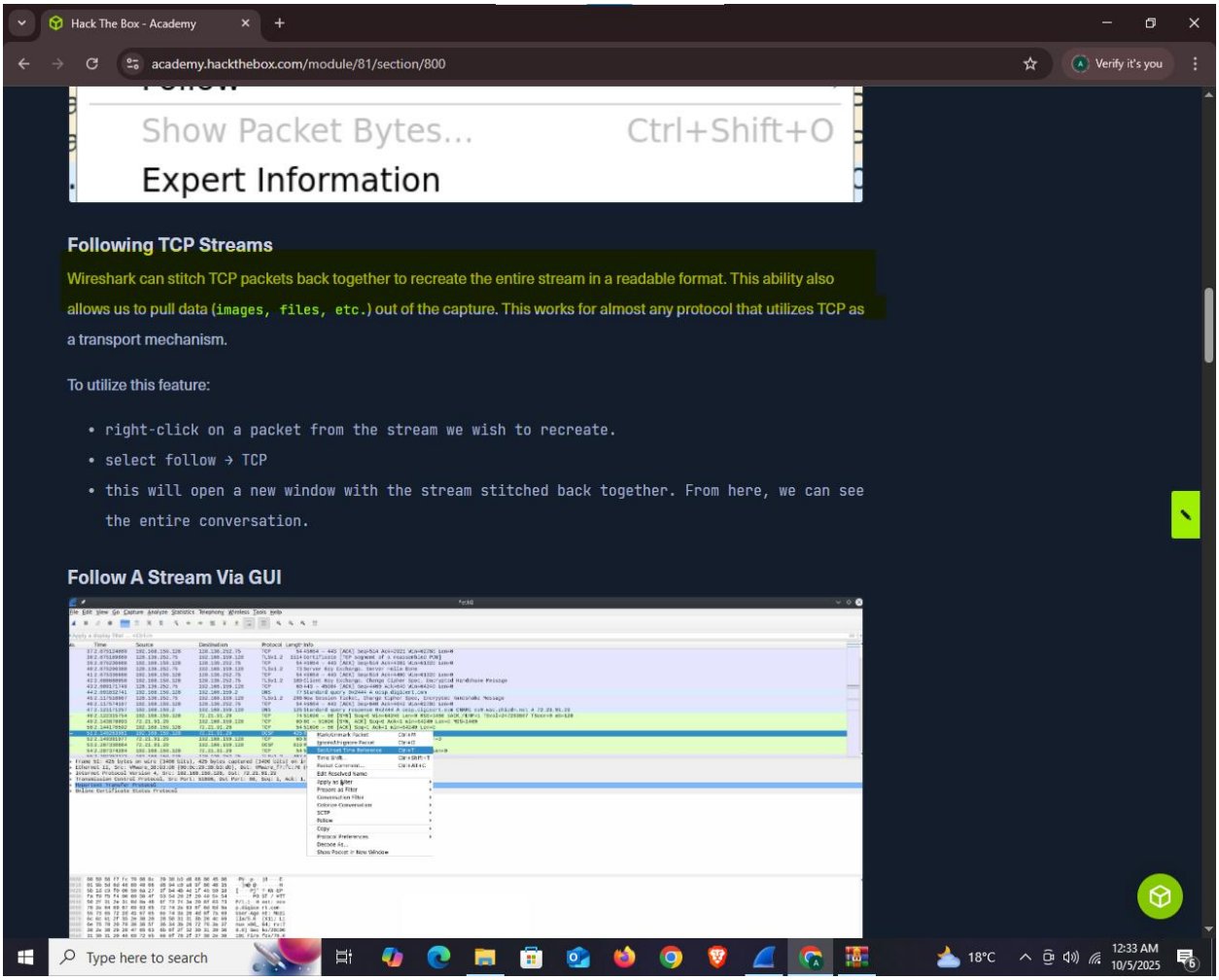

Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/800
Verify it's you
Life Left: 22m
Enable step-by-step solutions for all questions
Questions
Cheat Sheet
Answer the question(s) below to complete this Section and earn cubes!
+0 Which plugin tab can provide us with a way to view conversation metadata and even protocol breakdowns for the entire PCAP file?
Statistics
Submit
Hint
+0 What plugin tab will allow me to accomplish tasks such as applying filters, following streams, and viewing expert info?
Analyze
Submit
Hint
+0 What stream oriented Transport protocol enables us to follow and rebuild conversations and the included data?
TCP
Submit
Hint
Integrated Terminal
Type here to search
Match
3:17 PM
10/3/2025


Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/800
Verify it's you
+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
Previous
Next
+10 Streak pts
Mark Complete & Next
Powered by HACKTHEBOX
Integrated Terminal
Type here to search
Match
3:18 PM
10/3/2025

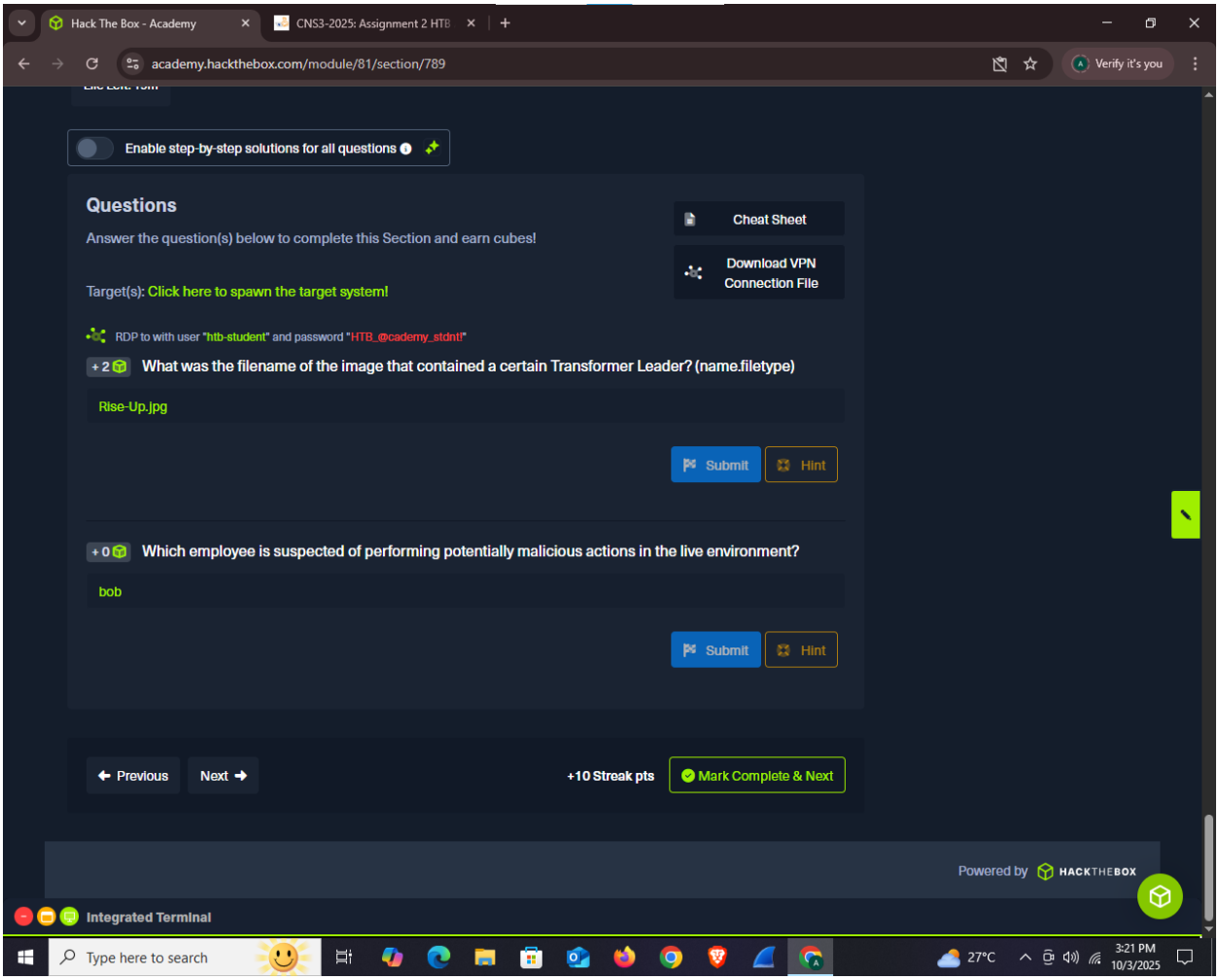

Hack The Box - Academy
academy.hackthebox.com/module/81/section/800
Verify it's you
HTB ACADEMY
Dashboard
Modules
Paths
Purchase Cubes
axelgoatee7
INTRO TO NETWORK TRAFFIC ANALYSIS
Page 12 / Wireshark Advanced Usage
Wireshark Advanced Usage
In this section, we will cover some advanced usage with Wireshark. The project developers have included many different capabilities ranging from tracking TCP conversations to cracking wireless credentials. The inclusion of many different plugins makes Wireshark one of the best traffic analysis tools.
Plugins
The analyze and statistics radials provide a plethora of plugins to run against the capture. In this section, we will work through a couple of them. We would cover all of which Wireshark offers, but sadly, it is simply not achievable in an introductory module. I urge everyone to experiment and play as we go through this journey.
The Statistics and Analyze Tabs
The Statistics and Analyze tabs can provide us with great insight into the data we are examining. From these points, we can utilize many of the baked-in plugins Wireshark has to offer.
The plugins here can give us detailed reports about the network traffic being utilized. It can show us everything from the top talkers in our environment to specific conversations and even breakdown by IP and protocol.
Statistics Tab
Cheat Sheet
Resources
Go to Questions
Table of Contents
Introduction
Network Traffic Analysis
Networking Primer - Layers 1-4
Networking Primer - Layers 5-7
Analysis
The Analysis Process
Analysis in Practice
Tcpdump
Tcpdump Fundamentals
Capturing With Tcpdump (Fundamentals Labs)
Tcpdump Packet Filtering
Type here to search
18°C
12:30 AM
10/5/2025

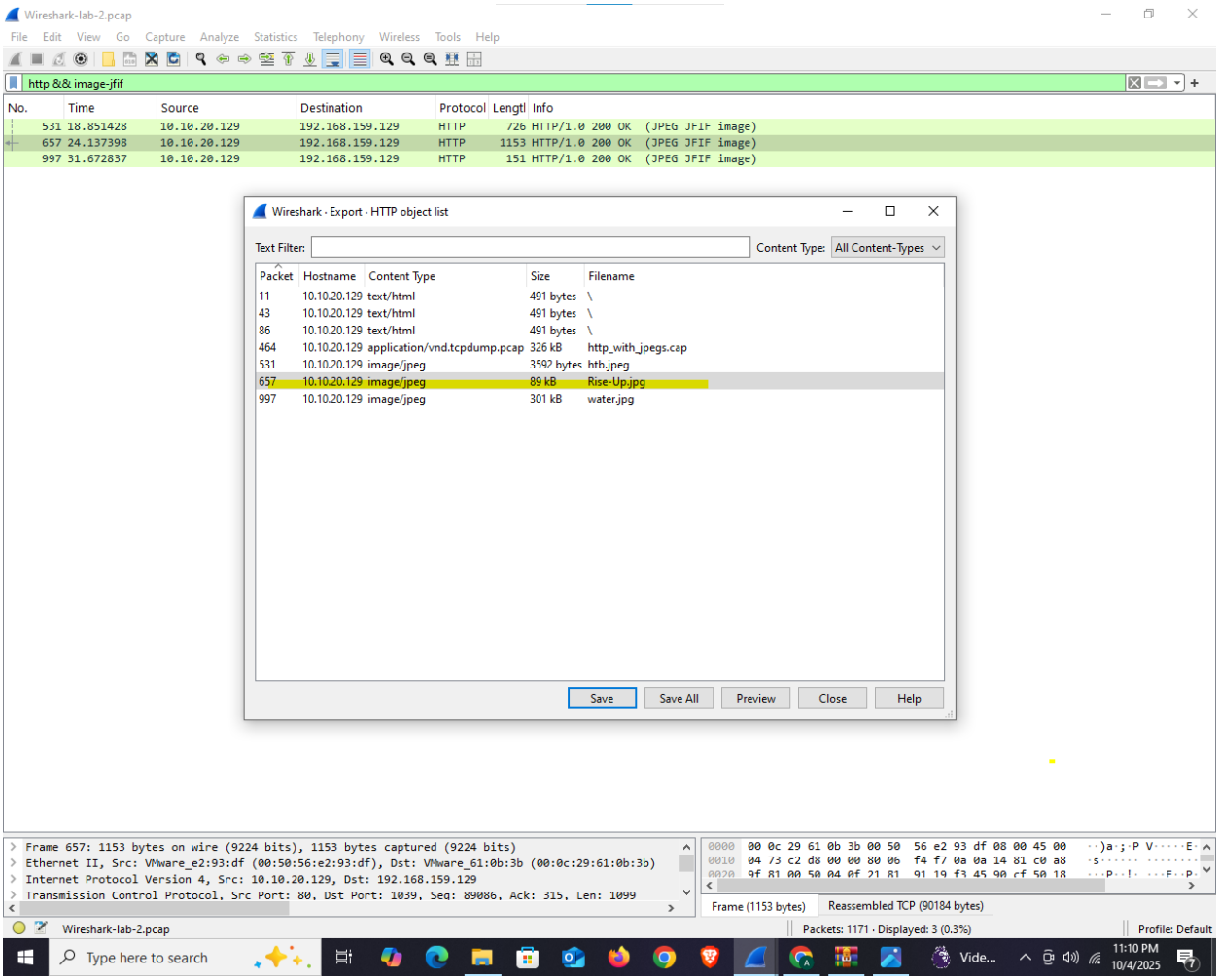
Proof.

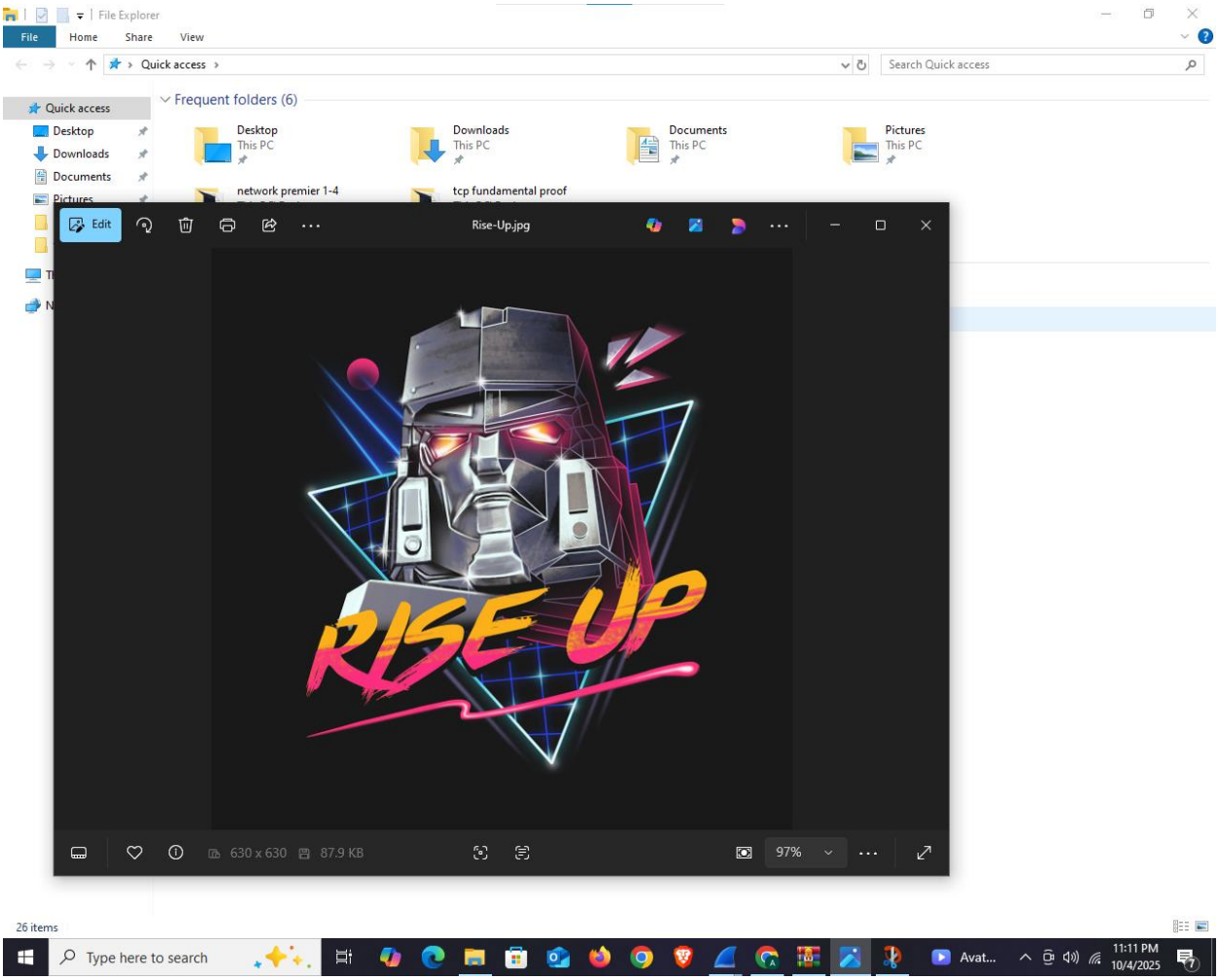

Hack The Box - Academy
academy.hackthebox.com/module/81/section/800
Verify it's you
Wireshark
Analysis with Wireshark
Familiarity With Wireshark
Wireshark Advanced Usage
Packet Inception, Dissecting Network Traffic With Wireshark
Guided Lab: Traffic Analysis Workflow
Decrypting RDP connections
Analyze
From the 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. Below are a few examples of how to use these plugins.
Analyze Tab
Analyze Statistics Telephony Wireless Tools
Display Filters...
Display Filter Macros...
Display Filter Expression...
Apply as Column Ctrl+Shift+I
Apply as Filter
Prepare a Filter
My Workstation
OFFLINE
Start Instance
1 / 1 spawns left
Type here to search
18°C
12:32 AM
10/5/2025



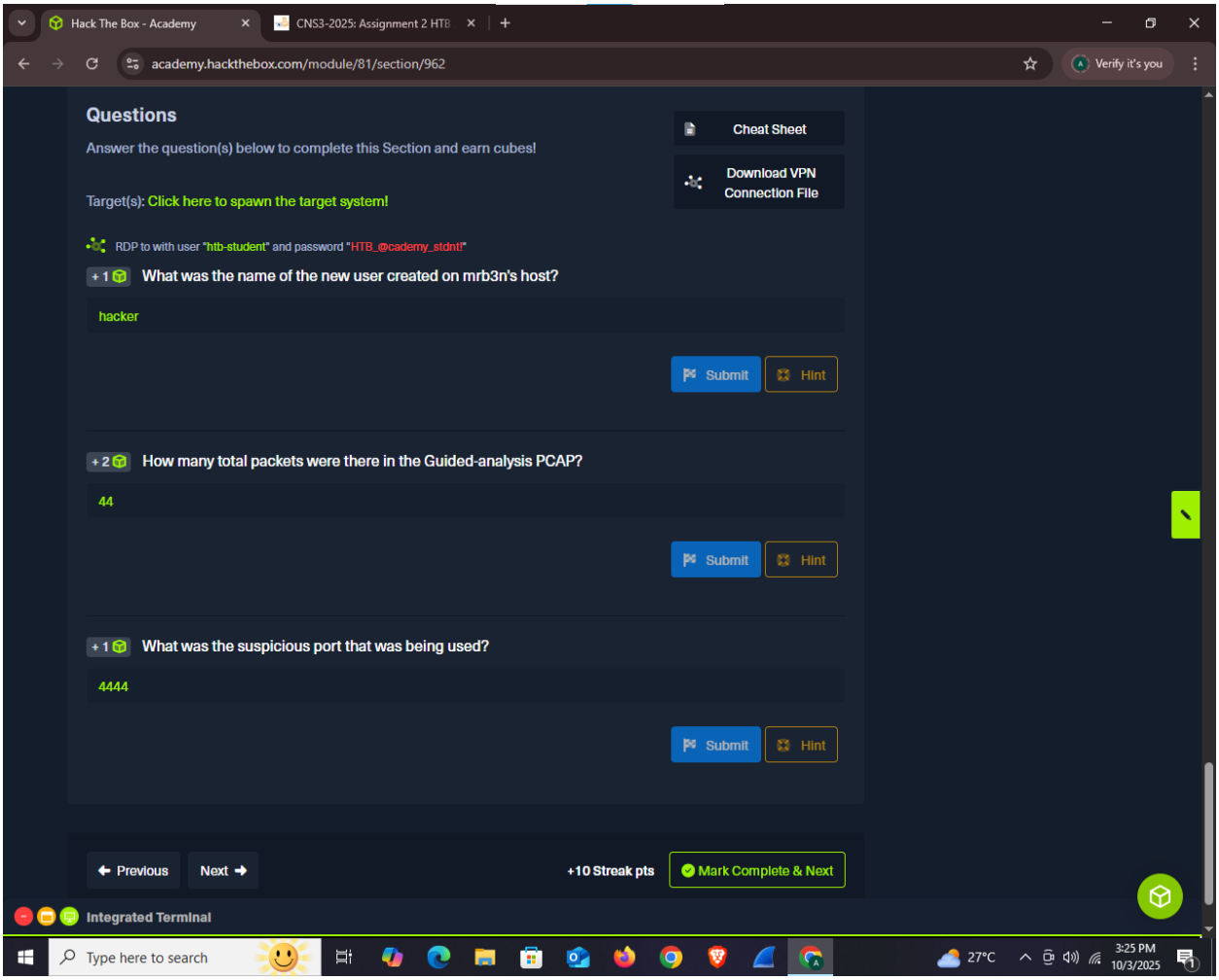
Question 9 packet inception ,dissecting network traffic with wireshark.

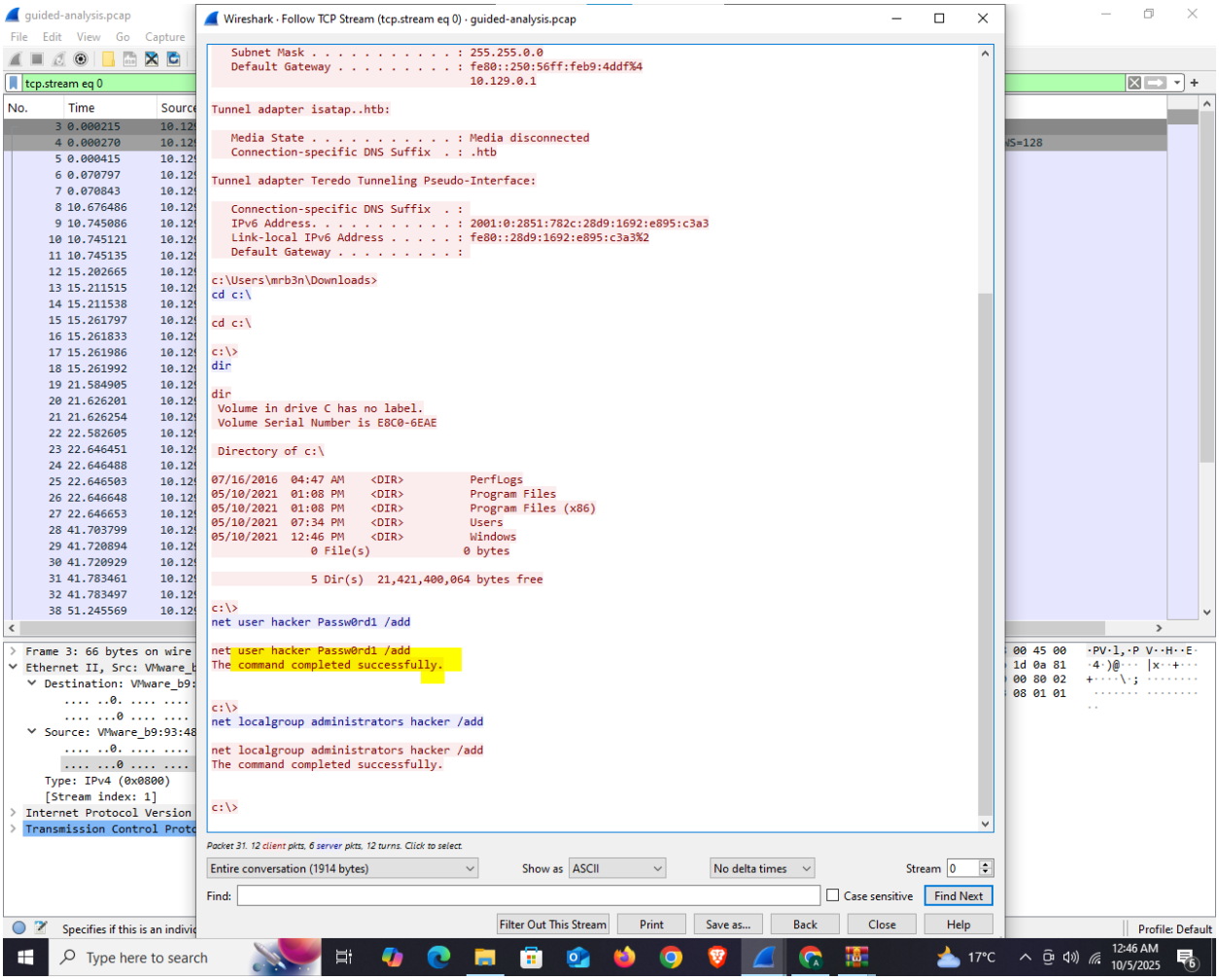

Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/789
Verify it's you
Enable step-by-step solutions for all questions
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
Download VPN Connection File
Target(s): Click here to spawn the target system!
RDP to 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
Submit
Hint
+0 Which employee is suspected of performing potentially malicious actions in the live environment?
bob
Submit
Hint
Previous
Next
+10 Streak pts
Mark Complete & Next
Powered by HACKTHEBOX
Integrated Terminal
Type here to search
27°C
3:21 PM
10/3/2025

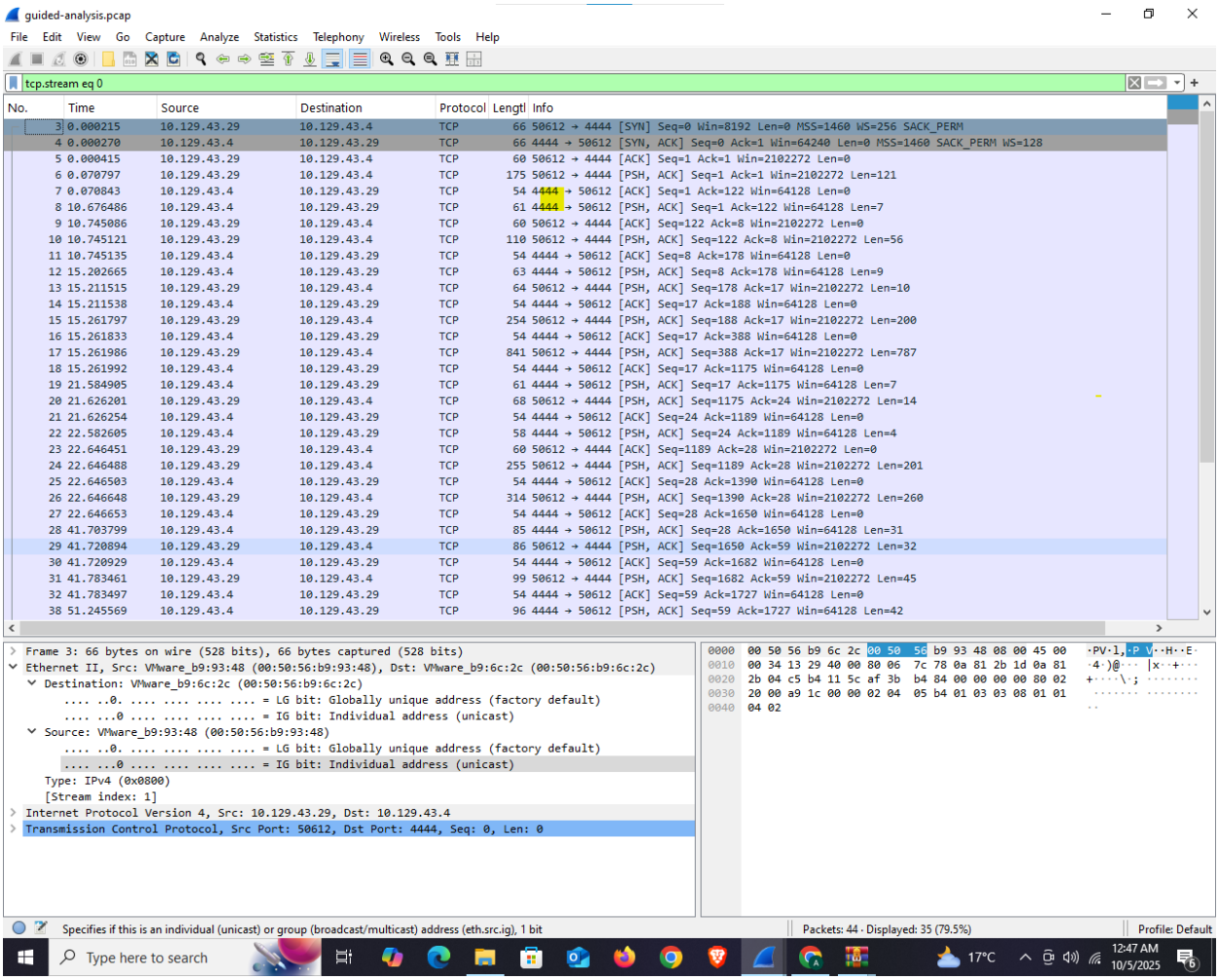

Wireshark-lab-2.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http && image-jfif
No. Time Source Destination Protocol Lengtl Info
531 18.851428 10.10.20.129 192.168.159.129 HTTP 726 HTTP/1.0 200 OK (JPEG JFIF image)
657 24.137398 10.10.20.129 192.168.159.129 HTTP 1153 HTTP/1.0 200 OK (JPEG JFIF image)
997 31.672837 10.10.20.129 192.168.159.129 HTTP 151 HTTP/1.0 200 OK (JPEG JFIF image)
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
Save Save All Preview Close Help
Frame 657: 1153 bytes on wire (9224 bits), 1153 bytes captured (9224 bits)
Ethernet II, Src: VMware_e2:93:df (00:50:56:e2:93:df), Dst: VMware_61:0b:3b (00:0c:29:61:0b:3b)
Internet Protocol Version 4, Src: 10.10.20.129, Dst: 192.168.159.129
Transmission Control Protocol, Src Port: 80, Dst Port: 1039, Seq: 89086, Ack: 315, Len: 1099
Frame (1153 bytes)
Reassembled TCP (90184 bytes)
Wireshark-lab-2.pcap
Packets: 1171 · Displayed: 3 (0.3%)
Profile: Default
Type here to search
11:10 PM
10/4/2025



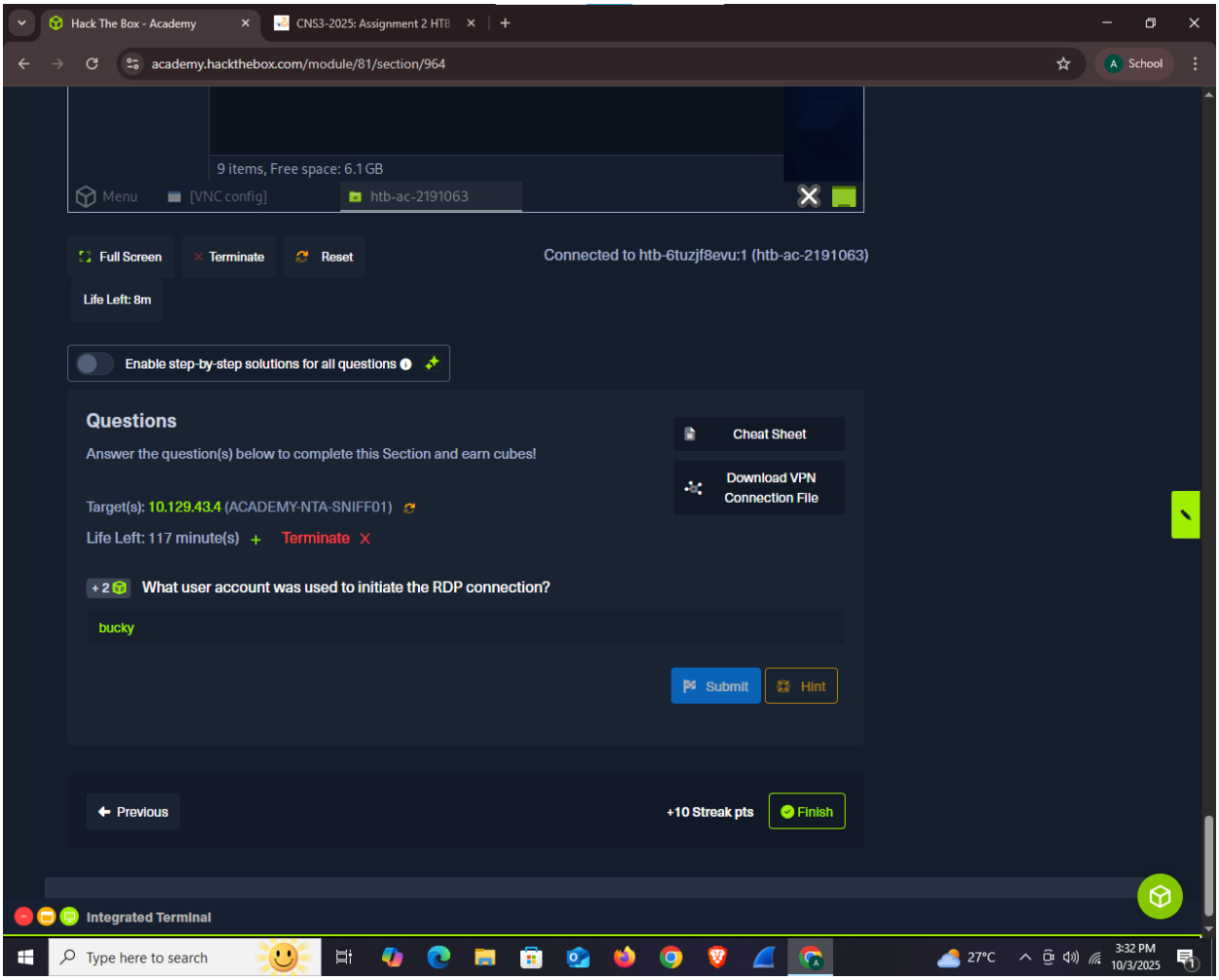
Question 10. Guided lab traffic analysis work flow.

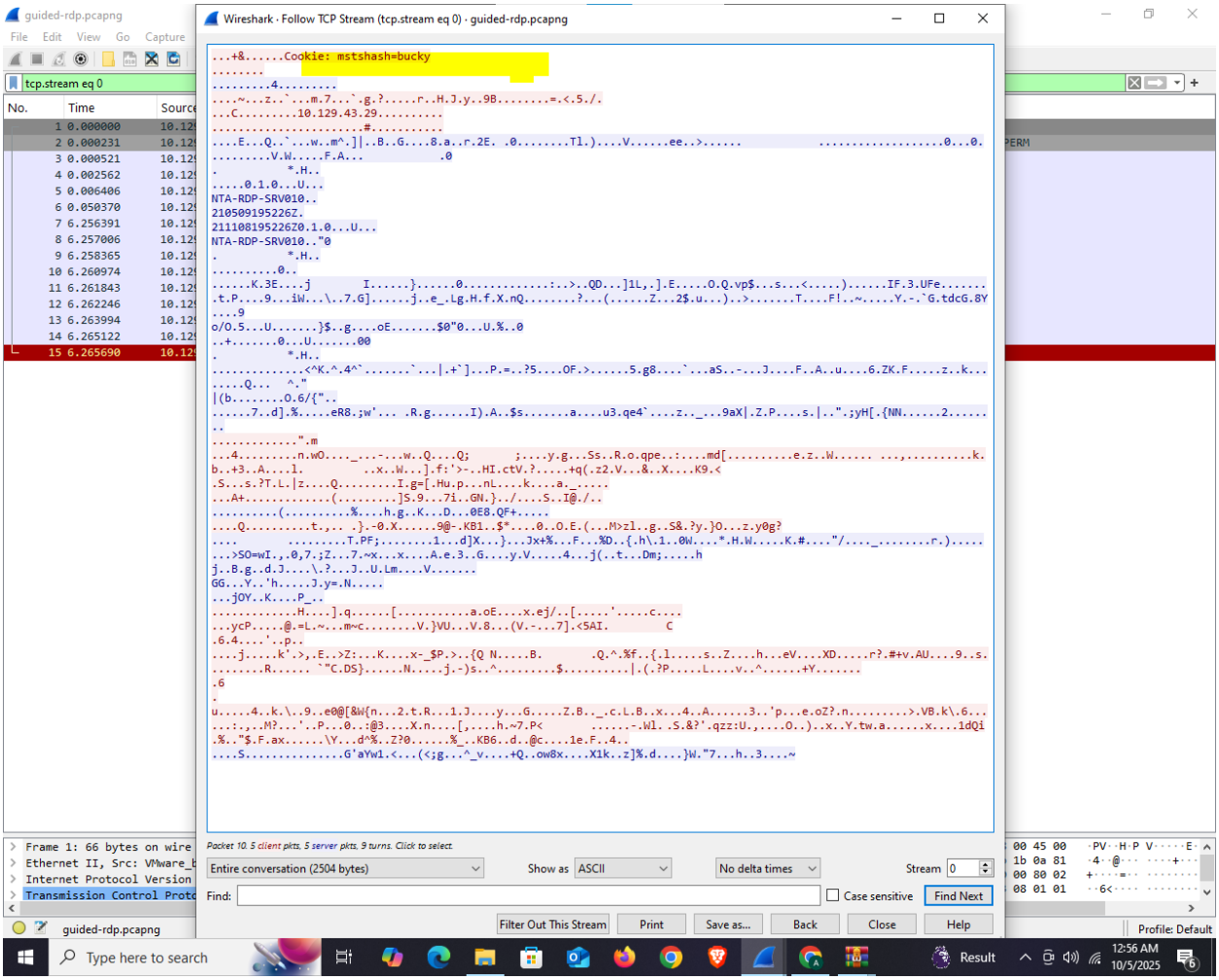

Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/962
Verify it's you
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
Download VPN Connection File
Target(s): Click here to spawn the target system!
RDP to with user "htb-student" 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
Previous
Next
+10 Streak pts
Mark Complete & Next
Integrated Terminal
Type here to search
27°C
3:25 PM
10/3/2025

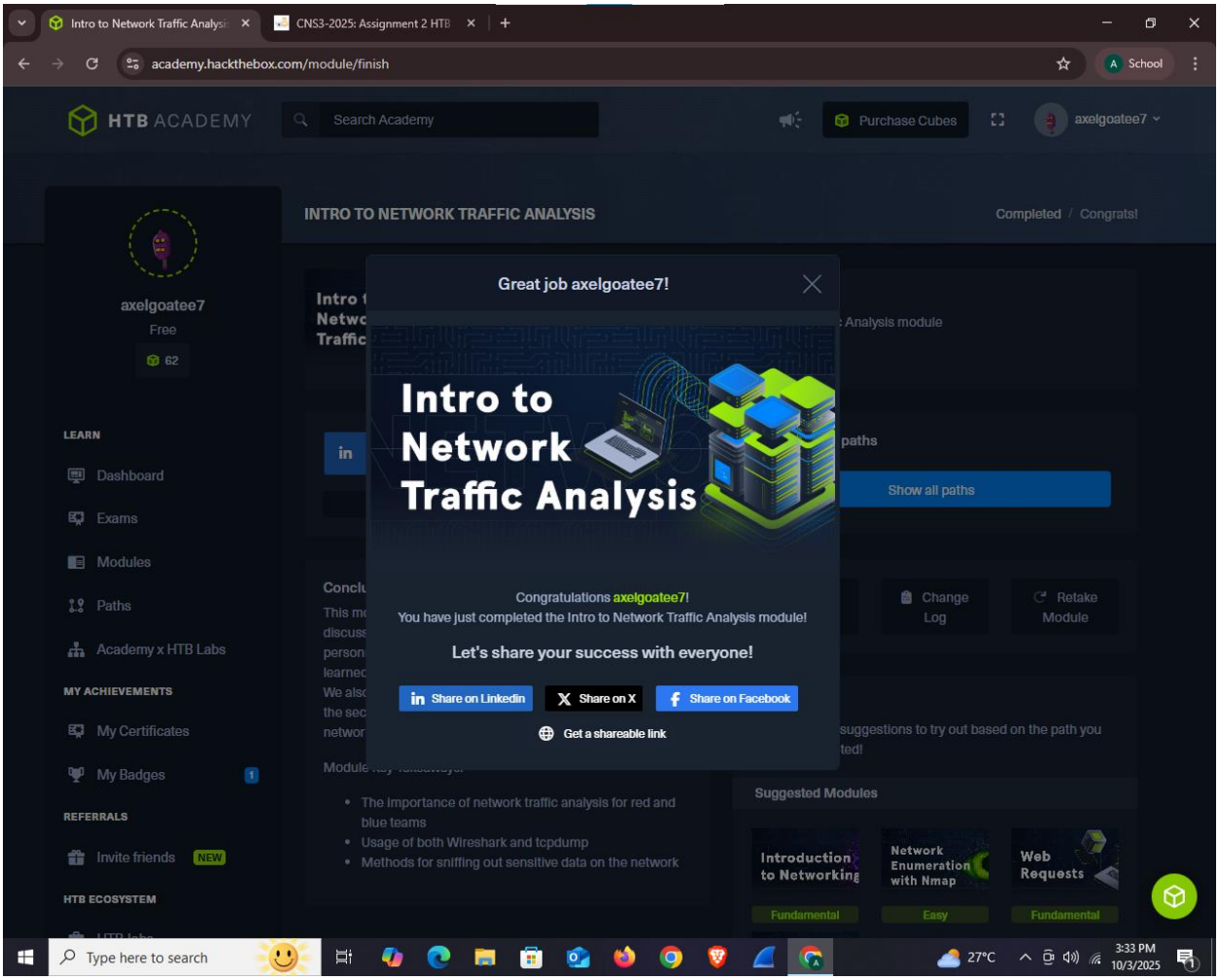

guided-analysis.pcap
File Edit View Go Capture
tcp.stream eq 0
No. Time Source
Wireshark · Follow TCP Stream (tcp.stream eq 0) · guided-analysis.pcap
Subnet Mask . . . . . : 255.255.0.0
Default Gateway . . . . . : fe80::250:56ff:feb9:4ddf%4
10.129.0.1
Tunnel adapter isatap..htb:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . : .htb
Tunnel adapter Teredo Tunneling Pseudo-Interface:
Connection-specific DNS Suffix . :
IPv6 Address. . . . . : 2001:0:2851:782c:28d9:1692:e895:c3a3
Link-local IPv6 Address . . . . . : fe80::28d9:1692:e895:c3a3%2
Default Gateway . . . . . :
c:\Users\mrb3n\Downloads>
cd c:\
cd c:\
c:\>
dir
dir
Volume in drive C has no label.
Volume Serial Number is E8C0-6EAE
Directory of c:\
07/16/2016 04:47 AM <DIR> PerfLogs
05/10/2021 01:08 PM <DIR> Program Files
05/10/2021 01:08 PM <DIR> Program Files (x86)
05/10/2021 07:34 PM <DIR> Users
05/10/2021 12:46 PM <DIR> Windows
0 File(s) 0 bytes
5 Dir(s) 21,421,400,064 bytes free
c:\>
net user hacker Passw0rd1 /add
net user hacker Passw0rd1 /add
The command completed successfully.
c:\>
net localgroup administrators hacker /add
net localgroup administrators hacker /add
The command completed successfully.
c:\>
Packet 31. 12 client pkts, 6 server pkts, 12 turns. Click to select.
Entire conversation (1914 bytes)
Show as ASCII
No delta times
Stream 0
Find:
Case sensitive
Find Next
Filter Out This Stream
Print
Save as...
Back
Close
Help
Frame 3: 66 bytes on wire
Ethernet II, Src: VMware_b
Destination: VMware_b9:
Source: VMware_b9:93:48
Type: IPv4 (0x0800)
[Stream index: 1]
Internet Protocol Version
Transmission Control Proto
Profile: Default
Type here to search
17°C
12:46 AM
10/5/2025



Question 11. Decrypting RDP connections.


Hack The Box - Academy
CNS3-2025: Assignment 2 HTB
academy.hackthebox.com/module/81/section/964
School
9 items, Free space: 6.1 GB
Menu
[VNC config]
htb-ac-2191063
Full Screen
Terminate
Reset
Connected to htb-6tuzjf8evu:1 (htb-ac-2191063)
Life Left: 8m
Enable step-by-step solutions for all questions
Questions
Answer the question(s) below to complete this Section and earn cubes!
Cheat Sheet
Download VPN Connection File
Target(s): 10.129.43.4 (ACADEMY-NTA-SNIFF01)
Life Left: 117 minute(s)
Terminate
+2
What user account was used to initiate the RDP connection?
bucky
Submit
Hint
Previous
+10 Streak pts
Finish
Integrated Terminal
Type here to search
27°C
3:32 PM
10/3/2025


guided-rdp.pcapng
File Edit View Go Capture
tcp.stream eq 0
No. Time Source
1 0.000000 10.12
2 0.000231 10.12
3 0.000521 10.12
4 0.002562 10.12
5 0.006406 10.12
6 0.050370 10.12
7 6.256391 10.12
8 6.257006 10.12
9 6.258365 10.12
10 6.260974 10.12
11 6.261843 10.12
12 6.262246 10.12
13 6.263994 10.12
14 6.265122 10.12
15 6.265690 10.12
Wireshark · Follow TCP Stream (tcp.stream eq 0) · guided-rdp.pcapng
...+&.....Cookie: mstshash=bucky
NTA-RDP-SRV010..
NTA-RDP-SRV010.."0
Packet 10. 5 client pkts, 5 server pkts, 9 turns. Click to select.
Entire conversation (2504 bytes)
Show as ASCII
No delta times
Stream 0
Find:
Case sensitive
Find Next
Filter Out This Stream
Print
Save as...
Back
Close
Help
Frame 1: 66 bytes on wire
Ethernet II, Src: VMware_b
Internet Protocol Version
Transmission Control Proto
guided-rdp.pcapng
Profile: Default
Type here to search
Result
12:56 AM
10/5/2025



Conclusion.

In this activity I have learnt networking primer layer 1-7 , the analysis process , tcpdump fundamentals and capturing with tcp dump , analysis with wireshark , wireshark advanced usage and decrypting with RDP connections