Oakland University
Department of Computer Science and Engineering

Lab5: Packet Sniffing and Privacy Analysis

Tools: Kali Linux, Wireshark, Tor Browser, `gpg`

Total Points: 100

Purpose

Packet sniffing involves capturing and analyzing network traffic to observe data packets as they move across a network. It is a key technique for identifying security vulnerabilities, unauthorized activity, and misconfigurations. Ethical hackers use tools like Wireshark to inspect live or recorded traffic, revealing details such as protocols, IP addresses, port numbers, and unencrypted data.

The primary goals of this lab are to detect insecure data transmissions, understand network behavior, and strengthen overall security. Through this lab, students will:

- Understand how personal data can leak through unencrypted communications.

- Use Kali Linux tools to analyze network traffic and identify privacy risks.

- Apply privacy-preserving techniques, including encryption.

- Evaluate the effectiveness of anonymity tools such as Tor.

Tasks

Task 1— Packet sniffing

- Use Wireshark to capture live network traffic and perform a detailed analysis of the packets.
- **Submission:** Screenshot illustrating the flow graph of captured traffic and TCP, IP ，Ethernet header

Task 2—Privacy Leaks in Network Traffic

- Demonstrate how unencrypted HTTP traffic exposes sensitive metadata and content.
- Capture and show HTTP GET requests revealing host, path, User Agent headers, and Cookie headers.
- **Submission:** Screenshot illustrating selected packet's HTTP section or HTTP stream.

Task 3— Encryption for Privacy

- Encrypt a plaintext file using symmetric GnuPG encryption.
- Verify decryption restores the original file.
- **Submission:** Screenshot showing encrypted file and successful decryption.

Task 4 — Anonymity with Tor

- Compare your direct connection's public IP with the Tor IP.
- **Submission:** Screenshots showing original public IP and Tor exit IP with confirmation message.

Assessment Criteria

1. **Structure of Submission and Clarity (5%)**
    - Name file: Assignment#X_YourFullName (e.g., Assignment#6_SolmazSalehian)
2. **Response Clarity and Correctness (95%)**
    - Organized work, clear screenshots, labeled images, relevant observations.

Part 1: Packet sniffing with Wireshark

Step 1: Install and Launch Wireshark

- Wireshark is available on Kali Linux and you can use it. It can also be installed on Windows/macOS.
- You can download from Wireshark.org if needed.

Wireshark is an open source application freely available on the Internet that allows you to capture packets as they appear at the network adaptor card. This means that you will be able to see all header information on the packet from each of the OSI layers. (Normally these headers are stripped off so that the only portion remaining is the data payload.) You will use the software to view complete packets and locate each layer's header, from the physical layer to the application layer. Doing so will help you to better understand network traffic and identify things that are "out of order." Using this program, you will:

- 1) Analyze simple protocols and learn about the software interface and the information it contains.
- 2) Observe, analyze and reconstruct specific packet interchanges between a computer and a server.

Step 2: Analyzing Simple Protocols

Start the Wireshark. The initial screen will resemble Figure 1. (the view might be different based the version that you have installed). Notice that your local interface is listed (if you have multiple interfaces, you may see more than one entry; the names may vary). You can click the interface and press "Start" to begin packet capture.

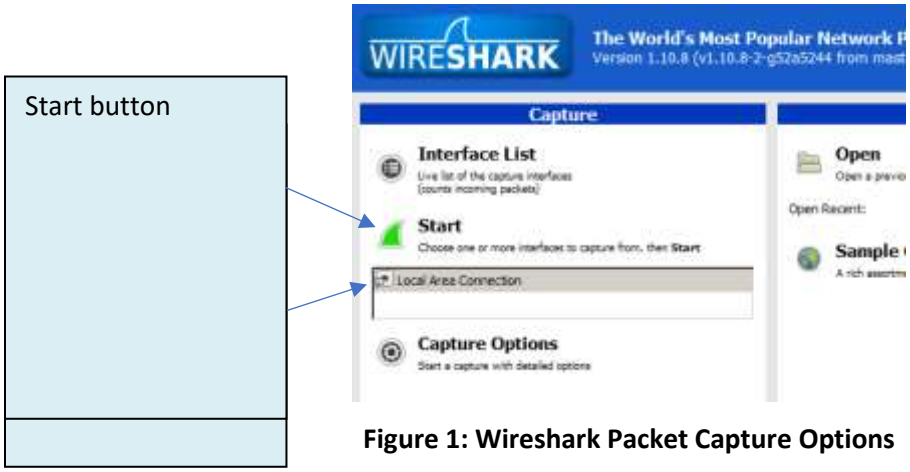


Figure 1: Wireshark Packet Capture Options

Below the menu, the capture window is divided into three distinct areas. The top is a listing of all packets received—the packet list pane; the middle provides the details of a packet selected in the

packet list pane and is called the packet details pane; and the bottom, called the packet bytes pane, shows the hexadecimal details of the selected packet and will highlight its (selected) fields. Figure 2 illustrates this and shows some captured packets.

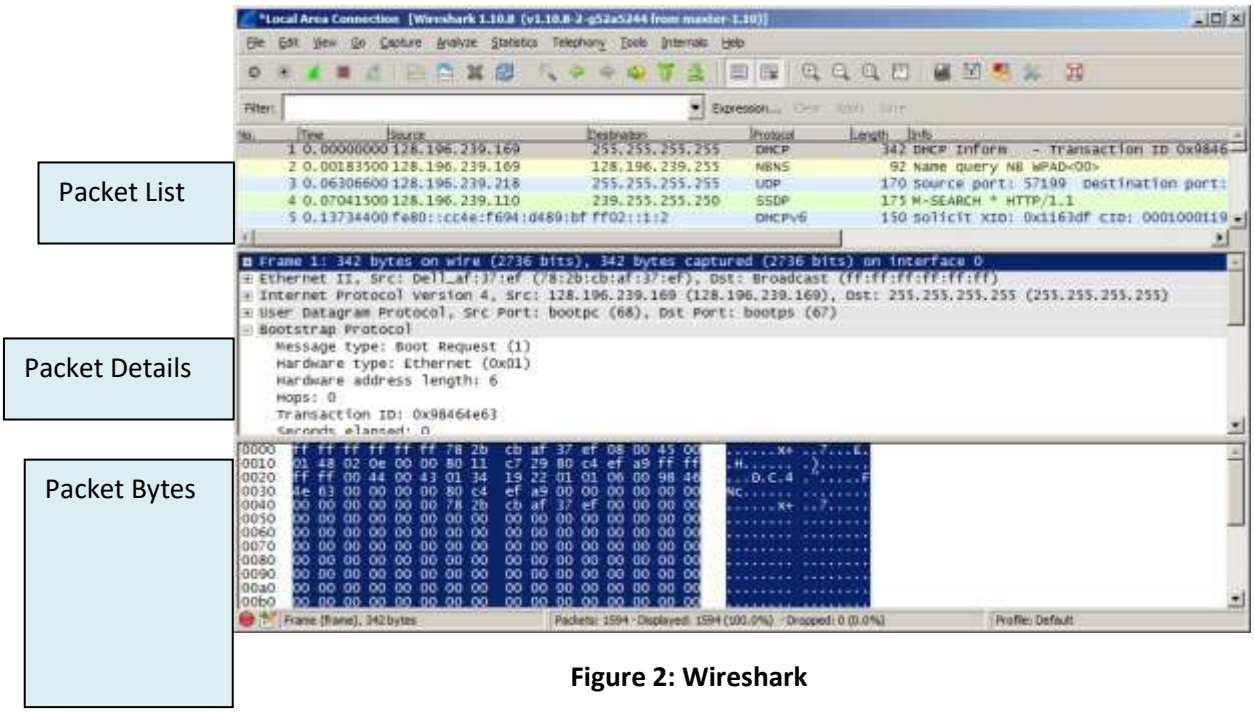


Figure 2: Wireshark

You can see in Figure 2 that multiple packets were captured, and the first packet is selected in the packet list pane. In the packet details pane, you can see the Ethernet frame header, the IP header, the UDP header and finally the data payload, which indicates that this is a Bootstrap Protocol packet. The packet byte pane shows the hexadecimal and ASCII equivalent of each packet at the bottom of the window. Selecting a field in the packet details pane will highlight the hex and ASCII portions of the packet in the packet byte pane.

Here's a look at what the buttons on the toolbar do.

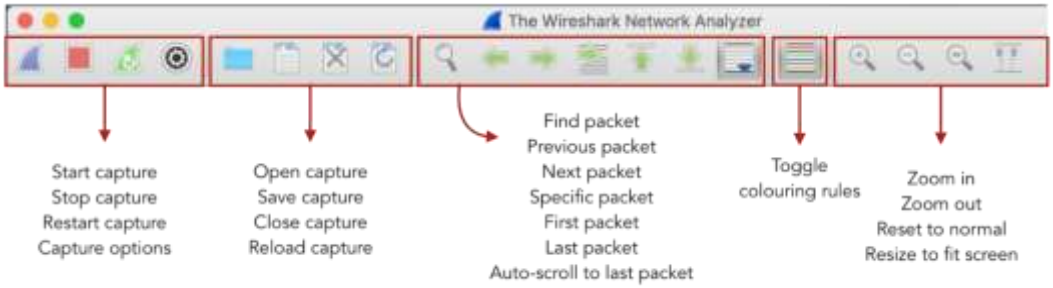


Image ref: http://www.cs.toronto.edu/~arnold/427/19s/427_19S/tool/Wireshark/index.html

Figure 3: Wireshark Toolbar

Wireshark helps you identify packet types by applying common-sense color coding. The packets in the main view are color-coded to helps users easily understand what they mean. To change this color, navigate to View > Coloring Rules. You can also add your own coloring rules and filter incoming packets for a specific IP address. The table below shows the default colors for different packet types:

Color in Wireshark	Packet Type
Light purple	TCP
Light blue	UDP
Black	Packets with errors
Light green	HTTP traffic
Light yellow	Windows-specific traffic, including Server Message Blocks (SMB) and NetBIOS
Dark yellow	Routing
Light purple	TCP

- Wireshark offers a way to see the communication between a client and server. You can navigate to Flow Graph (under Statistics tab) to check the communication flow between a server and client.

Submission 1- part1: Screenshot showing the flow graph of captured traffic.

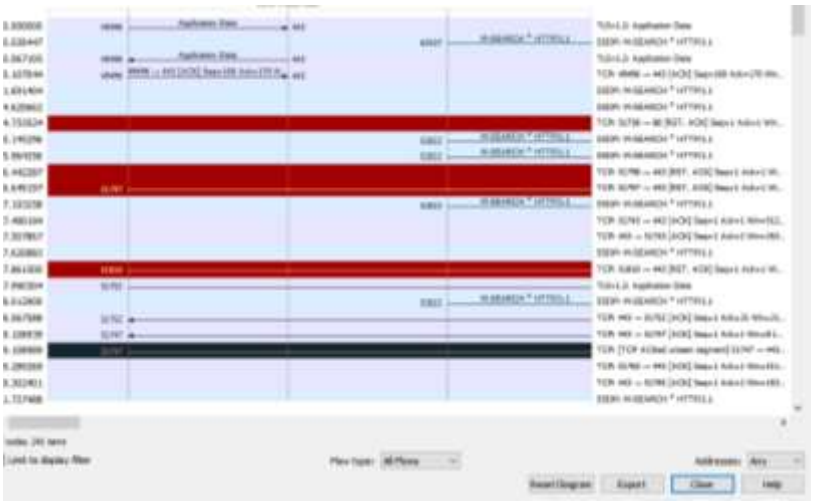
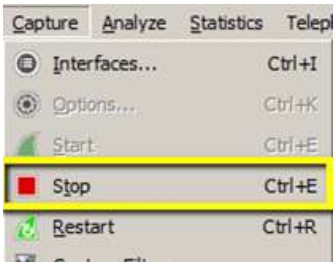


Figure 4: Flow Graph

Go ahead and start a capture session and after receiving a few packets, stop the packet capture (from the Wireshark menu, select the "Capture" menu item, and choose the "Stop" command from the drop-down menu).



Find a TCP packet in the packet list pane and select it. In the packet details pane, click on the "+" next to the word "Frame." When this part of the packet opens, you will see some summary information that Wireshark logs about every packet that it captures. Now open each subsequent section of the packet beginning with "Ethernet II." You should be able to find the portions of each packet corresponding to figures 5a through 5c within the packet details section (though the sizes of each section may not always be apparent without closer examination).

Preamble	Start of Frame	Destination Address	Source Address	Type	Data	FCS	Flag
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	46 - 1500 bytes	4 bytes	8 bytes

Figure 5a: An Ethernet II Frame Layout

Version Number	Header Length	Service Field	Total Length	ID	Flags	Fragment Offset	Time to Live	Next Protocol	Header Checksum	Source IP Address	Destination IP Address	Data
4 bits	4 bits	8 bits	16 bits	16 bits	3 bits	13 bits	8 bits	8 bits	16 bits	32 bits	32 bits	Variable

Figure 5b: The IP Header Layout

Source Port	Destination Port	Sequence Number	ACK Number	Header Length	Unused	Flags	Window Size	Header Checksum	Urgent Pointer	Options	Data
16 bits	16 bits	32 bits	32 bits	4 bits	3 bits	9 bits	16 bits	16 bits	16 bits	32 bits	Variable

Figure 5c: The TCP Header Layout

Figure 5a includes 20 bytes that are processed in the hardware and will not be seen in the packet details pane. These are the preamble (7 bytes), the Start of Frame (1 byte), the Frame Check Sequence (FCS, 4 bytes), and the final Flag (8 bytes).

Step 3: Finding Specific Packet Sequences

For this part you need a workstation that is connected to the Internet and one that receives its IP address from a DHCP server. You should have Wireshark installed on your workstation from part 1. In the step below you will observe the packets required to make and break a connection.

Step 3-1: Observing a TCP connection

1) Ensure that your capture options are set as before and begin another capture session.

2) After the capture session has begun, open a web browser on your workstation, allow the web page to finish loading, and then stop the packet capture session.

3) Look for the first three TCP packets in the packet list pane. TCP packets have a green background color (depending on your settings) and are easily recognized.

These three packets should be listed as [SYN], [SYN, ACK] and [ACK]. This 3-packet interchange builds a connection between two computers. You should notice that the destination port for the [SYN] packet is 80, indicating a web request. The second two packets should provide you with a sequence/acknowledgement analysis.

Submission2 -part1: Screenshot showing Ethernet, IP, and TCP headers of selected packet.

Part 2: Capturing HTTP Traffic

Open Wireshark and start a live capture on the network interface that carries your traffic (e.g., `eth0` or `wlan0`). While the capture is running, open a browser and visit `http://example.com` (make sure you explicitly use `http://` — if the site forces a redirect to HTTPS you'll see that in the capture). Stop the capture after the page loads, then apply a display filter such as `http` or `tcp.port == 80` to narrow results. Look through the packets for the HTTP **GET** request that fetched the page (it shows the requested path and host), inspect the HTTP headers for the **User-Agent** string (which reveals browser and OS details), and check any **Cookie** headers sent to or from the server (these show session or tracking values). Use **Follow → HTTP Stream** or expand the packet's HTTP section to read full request/response headers and bodies. **Caution:** only capture and inspect traffic on networks and devices you own or have permission to analyze—capturing others' private traffic may be illegal or unethical.

3. Start Wireshark capture.
4. Visit `http://example.com`.
5. Stop capture and apply filter `http` or `tcp.port == 80`.
6. Inspect HTTP GET request, User Agent, and Cookie headers.

Submission-part2: Screenshot showing the HTTP Get reuest.

Part 3: Encryption for Privacy

Encryption is a key practice for protecting privacy, and on Kali Linux, **GnuPG (GPG)** provides a powerful tool for securing data. GPG uses strong cryptographic algorithms to encrypt files, emails, or messages so only the intended recipient with the correct private key can decrypt them. In this section we will use gpg for encrypting a simple file.

Step 1: Install gpg

First, check the installed version of GPG with gpg --version, then update your package list using sudo apt update, and if GPG is not already installed, you can install it on Kali Linux with sudo apt install gnupg.

```
gpg --version
sudo apt update
sudo apt install gnupg
```

Step 2: Create a Text File

```
echo "This is my secret message." > secret.txt
```

Step 3: Encrypt the File

```
gpg -c secret.txt
```

- Enter a strong passphrase.
- Creates `secret.txt.gpg`.

Step 4: Decrypt the File

```
gpg secret.txt.gpg
```

- Enter passphrase to restore `secret.txt`.

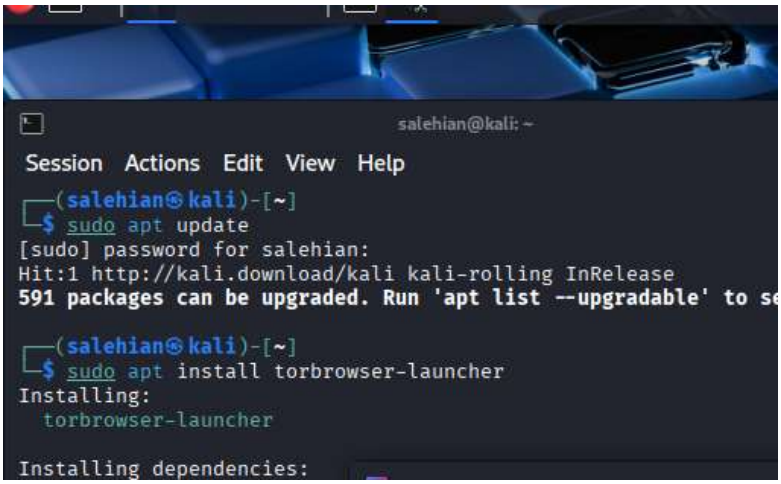
Submission-part3: submit a screenshot from two files (before encryption and after) and explain what encryption is and why it is essential?

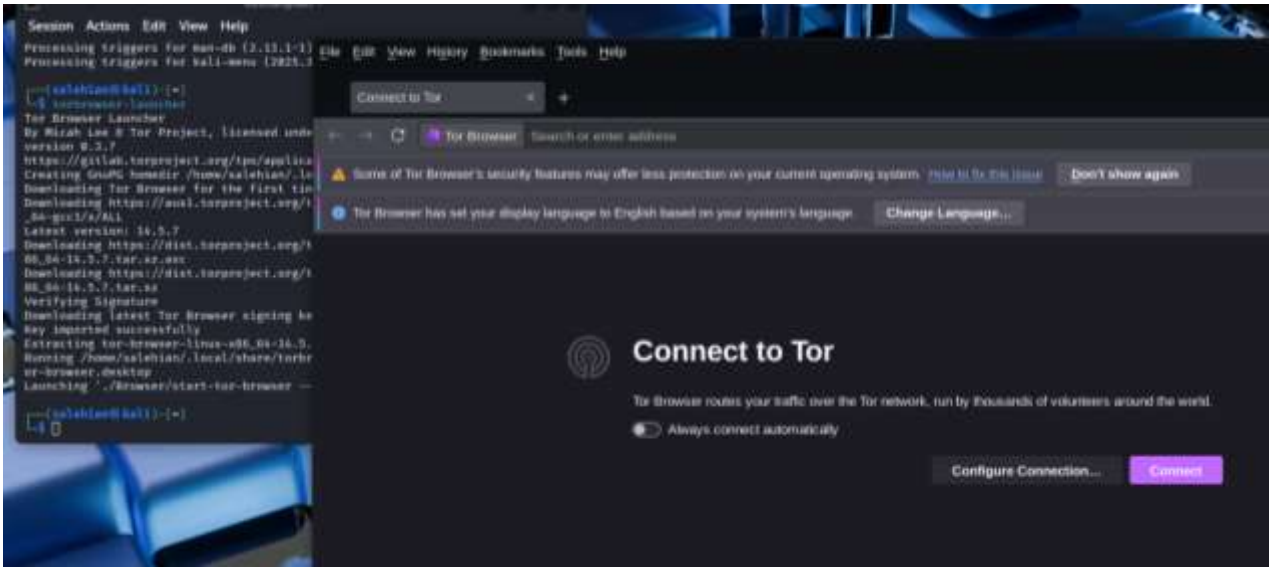
Part 4: Anonymity with Tor

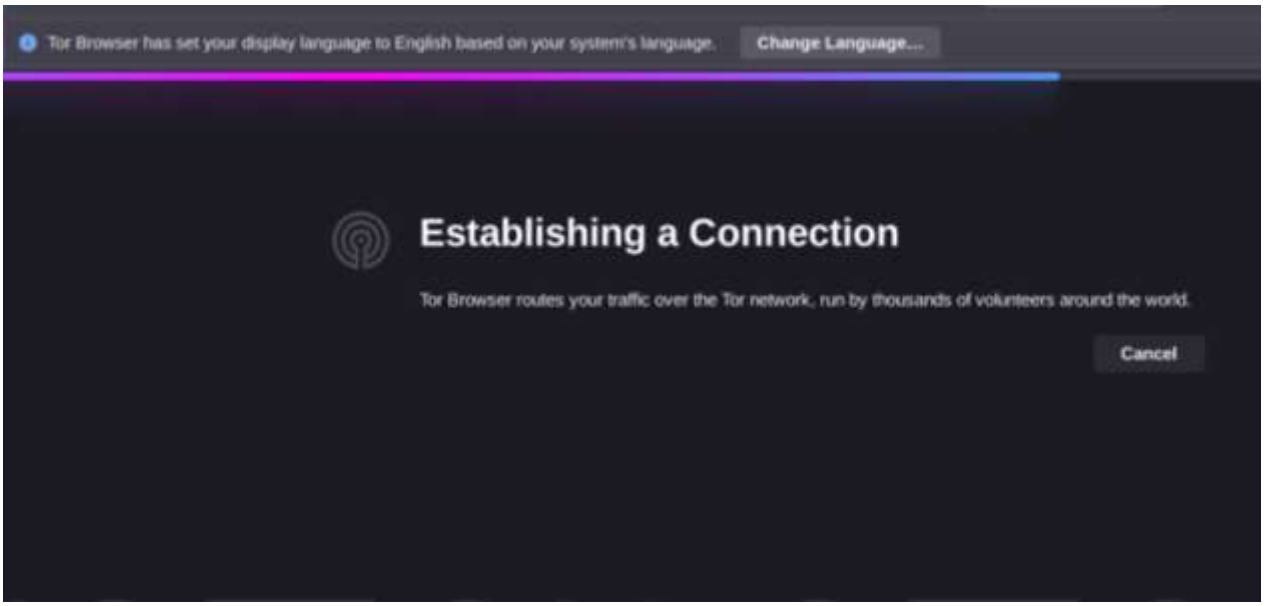
The Tor Browser is a tool that protects your anonymity online by hiding your IP address and encrypting your internet traffic. It routes your connection through a network of volunteer-operated servers, making it difficult for anyone to trace your activity or location. On Kali Linux, it is widely used by security researchers, journalists, and privacy-focused users to browse safely and avoid monitoring. In this section, we will use the Tor Browser to enhance privacy and strengthen user protection.

Step 1: Launch Tor Browser

1. Open a terminal in Kali Linux.
2. Update repositories: sudo apt update
3. Install Tor Browser if not installed: sudo apt install torbrowser-launcher
4. Launch Tor Browser: torbrowser-launcher.
5. On the startup screen, click 'Connect' to access the Tor network.


salehian@kali: ~
Session Actions Edit View Help
(salehian㉿kali)-[~]
$ sudo apt update
[sudo] password for salehian:
Hit:1 http://kali.download/kali kali-rolling InRelease
591 packages can be upgraded. Run 'apt list --upgradable' to se
(salehian㉿kali)-[~]
$ sudo apt install torbrowser-launcher
Installing:
torbrowser-launcher
Installing dependencies:


Connect to Tor
Tor Browser routes your traffic over the Tor network, run by thousands of volunteers around the world.
Always connect automatically
Configure Connection...
Connect


Tor Browser has set your display language to English based on your system's language.
Change Language...
Establishing a Connection
Tor Browser routes your traffic over the Tor network, run by thousands of volunteers around the world.
Cancel

Step 2: Compare IP Addresses

1. In a regular browser (e.g., Firefox/Chromium), go to https://check.torproject.org and note your IP address.
2. In Tor Browser, visit the same site and note the IP address displayed.
3. Compare the two results to see how Tor masks your real IP.

Step 3: Capture and Analyze Traffic

1. Open Wireshark with root privileges: sudo wireshark
2. Select your active network interface (e.g., wlan0 or eth0).

3. In Tor Browser, visit a website and visit `http://example.com` and Observe traffic in Wireshark, Traffic should appear encrypted.

Submission-part4:

1)Screenshots to show:

- Differences in IP addresses before and after Tor.
- Wireshark confirms encrypted Tor traffic.

2) Discuss:

- What are the strengths of Tor in providing anonymity?
- What are the limitations of Tor?
- Can Tor guarantee complete anonymity?
- How Wireshark can be used to enhance security and privacy?

References

- FitzGerald, J., Dennis, A., & Durcikova, A. (2017). *Business Data Communications & Networking, 13th Ed.* Wiley. ISBN 978-11119368830
- CompTIA - What is Wireshark