INDEX

Practical 1.

Encrypting and Decrypting Data Using a Hacker Tool

Background / Scenario

What if you work for a large corporation that had a corporate policy regarding removable media? Specifically, it states that only encrypted zipped documents can be copied to portable USB flash drives.

In this scenario, the Chief Financial Officer (CFO) is out-of-town on business and has contacted you in a panic with an emergency request for help. While out-of-town on business, he attempted to unzip important documents from an encrypted zip file on a USB drive. However, the password provided to open the zip file is invalid. The CFO contacted you to see if there was anything you could to do.

There may some tools available to recover lost passwords. This is especially true in situations such as this where the cybersecurity analyst could acquire pertinent information from the CFO, such as the length of the password, and an idea of what it could be. Knowing pertinent information dramatically helps when attempting to recover passwords.

Examples of password recovery utilities and programs include hashcat, John the Ripper, Lophtcrack, and others. In our scenario, we will use **fcrackzip** which is a simple Linux utility to recover the passwords of encrypted zip files.
Consider that these same tools can be used by cybercriminals to discover unknown passwords. Although they would not have access to some pertinent information, with time, it is possible to discover passwords to open encrypted zip files. The amount of time required depends on the password strength and the password length. Longer and more complex passwords (mix of different types of characters) are more secure.

In this lab, you will:

- Create and encrypt sample text files.
- Decrypt the encrypted zip file.

Note: This lab should be used for instructional purposes only. The methods presented here should NOT be used to secure truly sensitive data.

Required Resources

- Kali Linux Virtual Machine
- Internet access

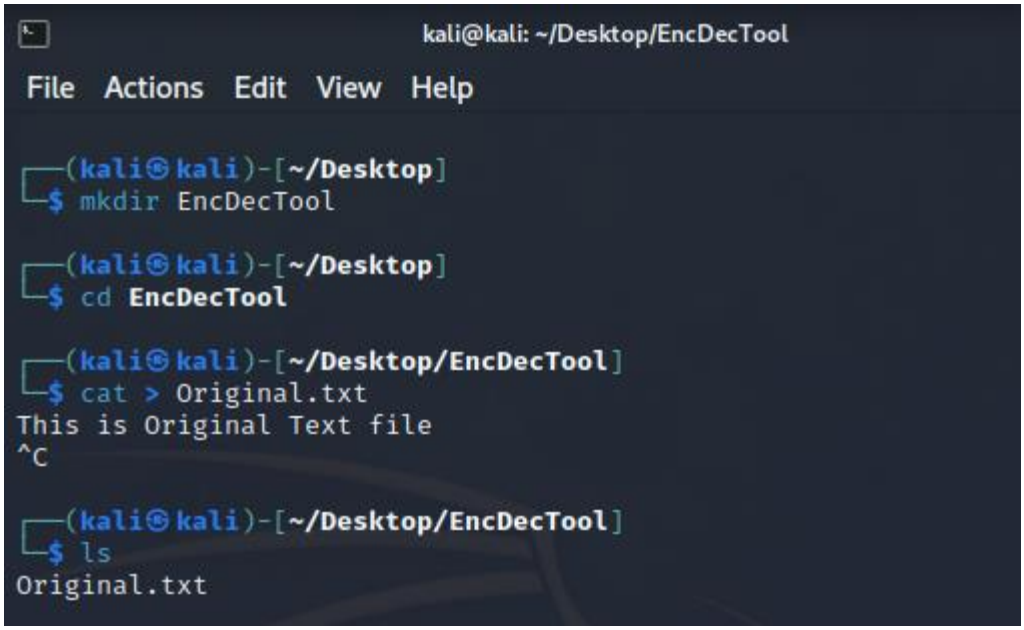
Part 1: Create and Encrypt Files

In this part, you will create a few text files that will be used to created encrypted zip files in the next step.

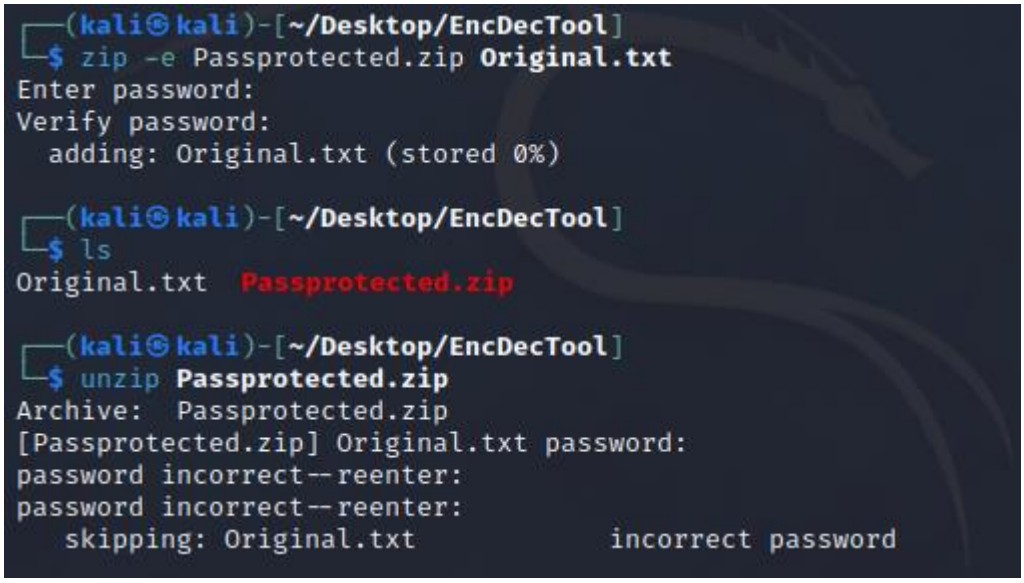
Step 1: Create text files.

a. Start the Kali Linux VM.

b. Open a terminal window. Verify that you are in the analyst home directory. Otherwise, enter **cd ~** at the terminal prompt.



Step 2: Zip and encrypt the text files.



Attempt to open a zip using an incorrect password as shown.

Part 2: Recover Encrypted Zip File Passwords

In this part, you will use the **fcrackzip** utility to recover lost passwords from encrypted zipped files. Fcrackzip searches each zip file given for encrypted files and tries to guess the password using brute-force methods.

The reason we created zip files with varying password lengths was to see if password length influences the time it takes to discover a password.

Step 1: Introduction to fcrackzip

fcrackzip is a fast password cracker partly written in assembler. It is able to crack password protected zip files with brute force or dictionary based attacks, optionally testing with unzip its results.

Insatall fcrackzip

$sudo apt update

sudo apt install fcrackzip

```
┌──(kali㉿kali)-[~/Desktop/enctool]
└─$ sudo apt update
```

```
┌──(kali㉿kali)-[~/Desktop/enctool]
└─$ sudo apt install fcrackzip
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Suggested packages:
```

Step 2: Recovering Passwords using fcrackzip

Now attempt to recover the password of the **Passprotected.zip** file.

```
┌──(kali㉿kali)-[~/Desktop/EncDecTool]
└─$ unzip Passprotected.zip
Archive:  Passprotected.zip
[Passprotected.zip] Original.txt password:
password incorrect--reenter:
password incorrect--reenter:
   skipping: Original.txt            incorrect password

┌──(kali㉿kali)-[~/Desktop/EncDecTool]
└─$ fcrackzip -vul 1-8 Passprotected.zip
found file 'Original.txt', (size cp/uc     39/    27, flags 9, chk 8bca)


PASSWORD FOUND!!!!: pw == 123
```

Practical 2
Encrypting and Decrypting Data Using OpenSSL

OpenSSL is an open source project that provides a robust, commercial-grade, and full-featured toolkit for the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols. It is also a general-purpose cryptography library. In this lab, you will use OpenSSL to encrypt and decrypt text messages.

Required Resources

- Kali Linux Virtual Machine
- Internet access

```
┌──(kali㉿kali)-[~/Desktop]
└─$ mkdir EncDec

┌──(kali㉿kali)-[~/Desktop]
└─$ cd EncDec

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ cat > OrgData.txt
This is Original data file
^C

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ ls
OrgData.txt
```

From the same terminal window, issue the command below to encrypt the text file. The command will use AES-256 to encrypt the text file and save the encrypted version as **BinaryEnc.enc.** OpenSSL will ask for a password and for password confirmation. Provide the password as requested and be sure to remember the password.

```
┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ openssl aes-256-cbc -in OrgData.txt -out BinaryEnc.enc
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ ls
BinaryEnc.enc  OrgData.txt
```

When the process is finished, use the **cat** command again to display the contents of the **BinaryEnc.enc** file.

```
┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ cat BinaryEnc.enc
◆◆Q-◆*O◆◆◆b;a%CT◆◆◆E◆O[◆jн◆◆
```

Did the contents of the **BinaryEnc.enc** file display correctly? What does it look like? Explain.
No. The file seems broken as just symbols are displayed. The symbols are shown because OpenSSL has generated a binary file.

Decrypting Binary Messages with OpenSSL

With a similar OpenSSL command, it is possible to decrypt **BinaryEnc.enc.**
Use the command below to decrypt **BinaryEnc.enc** and get Original data in new File (**Binary2org.txt**). OpenSSL will ask for the password used to encrypt the file. Enter the same password again.

When OpenSSL finishes decrypting the **BinaryEnc.enc** file, it saves the decrypted message in a text file called **Binary2org.txt.** Use the **cat** display the contents of **Binary2org.txt:**

```
┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ openssl aes-256-cbc -d -in BinaryEnc.enc -out Binary2Org.txt
enter AES-256-CBC decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ cat Binary2Org.txt
This is Original data file
```

Was the letter decrypted correctly?
Yes, the letter was decrypted correctly.

To make the Encrypted file readable (Alphabetically), run the OpenSSL command again, but this time add the **-a** option. The **-a** option tells OpenSSL to encode the encrypted message using a different encoding method of Base64 before storing the results in a file.

```
┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ openssl aes-256-cbc -a -in OrgData.txt -out Base64Enc.enc
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ cat Base64Enc.enc
U2FsdGVkX180RAYdXmYheVX57kShq7w0ibob3TqFJlGPdWjpKgsIMQZYC2Zwp7nO

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ openssl aes-256-cbc -a -d -in Base64Enc.enc -out Base64_2Org.txt
enter AES-256-CBC decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

┌──(kali㉿kali)-[~/Desktop/EncDec]
└─$ cat Base64_2Org.txt
This is Original data file
```

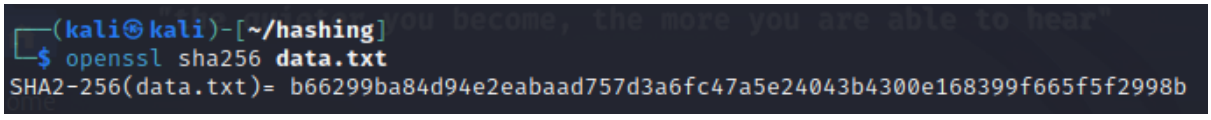
Practical 3

Hashing a Text File with OpenSSL and Verifying Hashes

Cryptographic hash functions can be used for verifying file integrity. To check that the file has not been altered during transfer over the network. Let's say we have the following file named data.txt with the following content:

The openssl command can be used to perform various digest operations. To generate a hash of the file data.txt using SHA-256, run the following command:

Step 1: generate the checksum



Step 2: write the result to a file

To write result to a file called data.sha256, use the -out option:

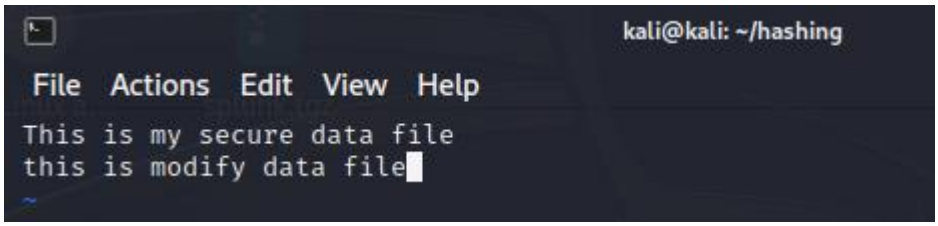
The output should be same as output 1 above.

To verify the file name, the algorithm used and the hash code

Not that if you in any case change the filename, i.e. change case(upper/lower) or even add space the check sum would be different.

Step 3: Modify the file

```
┌──(kali㉿kali)-[~/hashing]
└─$ vi data.txt
```



Step 4: verify the integrity

Your recipient will 1) open the checksum i.e. cat data.sha256 and then also 2) regenerate the checksum again by running

```
┌──(kali㉿kali)-[~/hashing]
└─$ openssl sha256 data.txt
SHA2-256(data.txt)= 49c16c8b5ce69361c1d1353a934859f4a2f5edf29bf10607d1e16b76aff0ab93

┌──(kali㉿kali)-[~/hashing]
└─$ cat data.sha256
SHA2-256(data.txt)= b66299ba84d94e2eabaad757d3a6fc47a5e24043b4300e168399f665f5f2998b
```

If the hash codes for 1) and 2) match then integrity is verified. If not the file was altered during transmission. Don't open or executed it.

Practical 4

Examining Telnet and SSH in Wireshark

In this lab, you will configure a router to accept SSH connectivity and use Wireshark to capture and view Telnet and SSH sessions. This will demonstrate the importance of encryption with SSH.

You will use Wireshark to capture and view the transmitted data of a Telnet session.

Step 1: Capture data.

a. Start the CyberOps Workstation VM and log in with username analyst and password cyberops.

b. Open a terminal window and start Wireshark.

[root@secOps ~]# wireshark &

c. Start a Wireshark capture on the Loopback: lo interface.

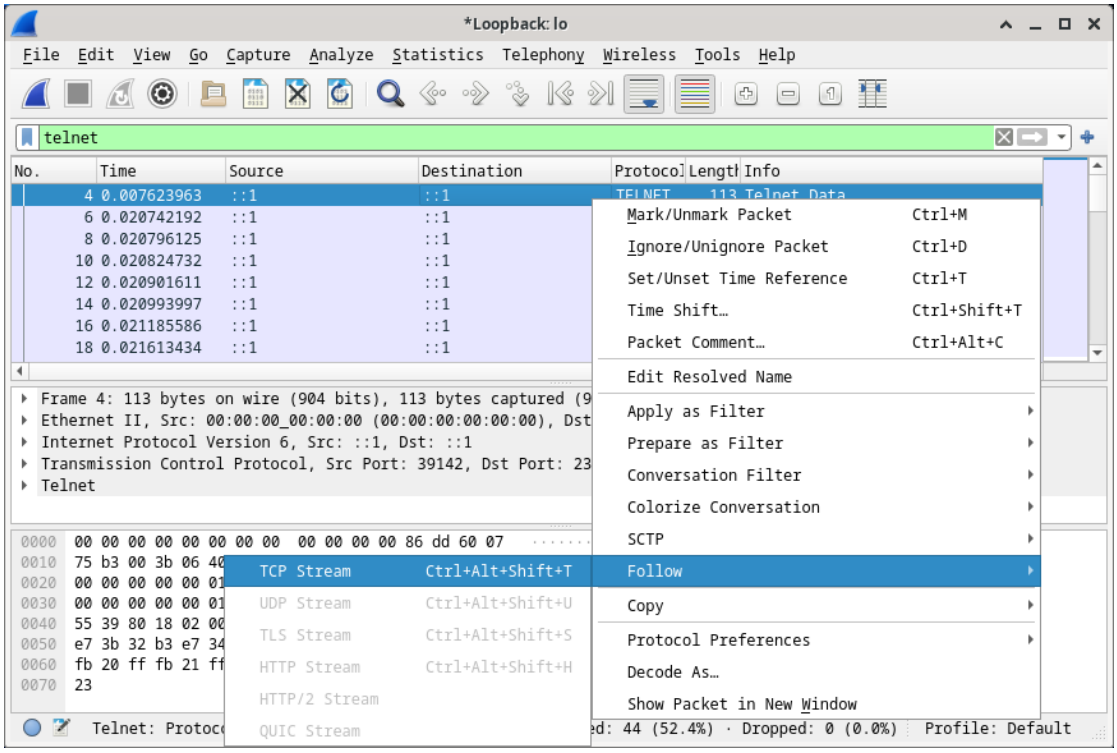
d. Open another terminal window. Start a Telnet session to the localhost. Enter username analyst and password cyberops when prompted. Note that it may take several minutes for the "connected to localhost" and login prompt to appear.

```
[root@secOps ~]# telnet localhost
Trying ::1...
Connected to localhost.
Escape character is '^]'.

Linux 4.10.10-1-ARCH (unallocated.barefruit.co.uk) (pts/12)

secOps login: analyst
Password:
Last login: Fri May 28 10:50:52 from localhost.localdomain
[root@secOps ~]#
```

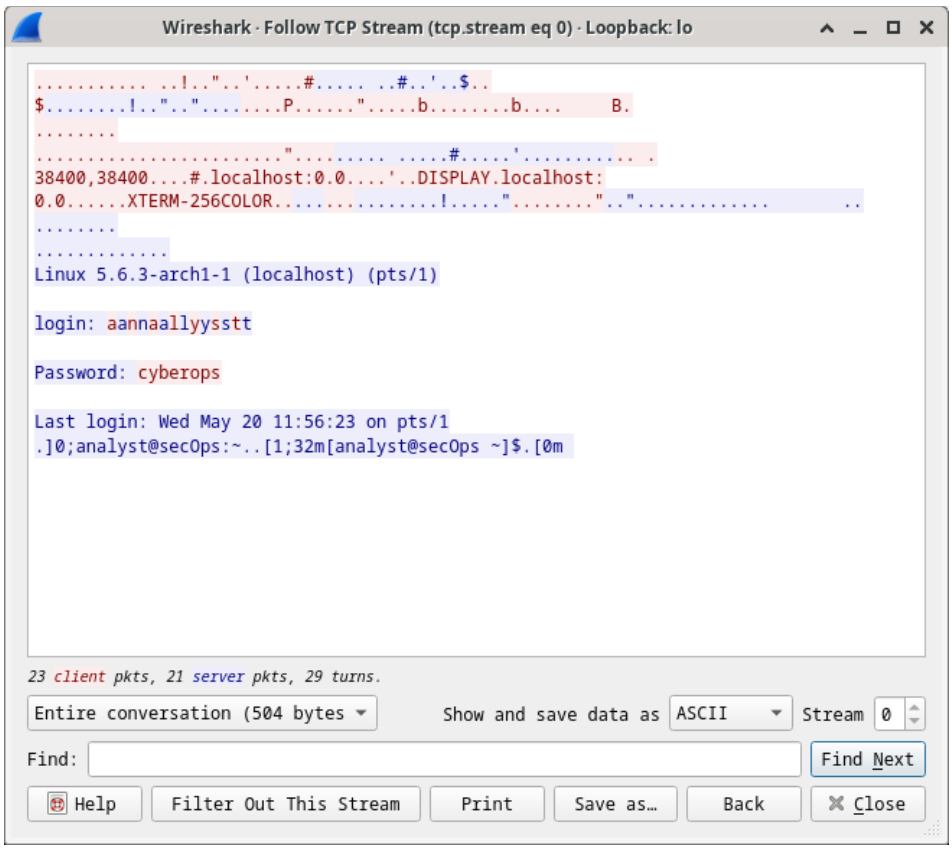
e. Stop the Wireshark capture after you have provided the user credentials.

Step 2: Examine the Telnet session.

a. Apply a filter that only displays Telnet-related traffic. Enter **telnet** in the filter field and click **Apply**.
b. Right-click one of the **Telnet** lines in the **Packet list** section of Wireshark, and from the drop-down list, select **Follow > TCP Stream**.



c. The Follow TCP Stream window displays the data for your Telnet session with the CyberOps Workstation VM. The entire session is displayed in plaintext, including your password. Notice that the username that you entered is displayed with duplicate characters. This is caused by the echo setting in Telnet to allow you to view the characters that you type on the screen.



d. After you have finished reviewing your Telnet session in the **Follow TCP Stream** window, click **Close**.
e. Type exit at the terminal to exit the **Telnet** session.

```
[root@secOps ~]# exit
```

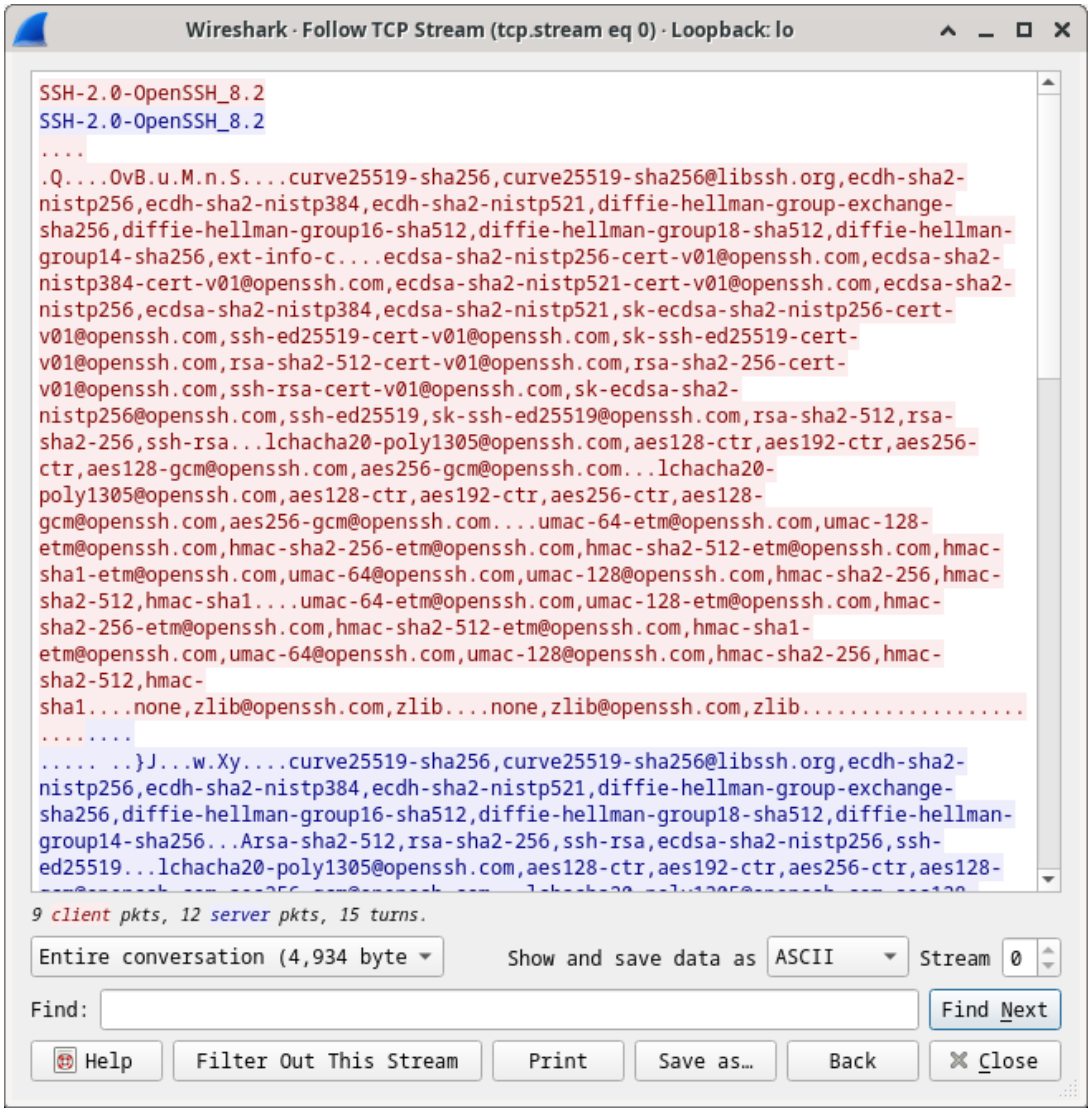
Part 2: Examine an SSH Session with Wireshark

In Part 2, you will establish an SSH session with the localhost. Wireshark will be used to capture and view the data of this SSH session.

a. Start another Wireshark capture using the **Loopback: lo** interface.
b. You will establish an SSH session with the localhost. At the terminal prompt, enter ssh localhost. Enter yes to continue connecting. Enter the **cyberops** when prompted.

```
[analyst@secOps ~]$ ssh localhost

The authenticity of host 'localhost (::1)' can't be established.

ECDSA key fingerprint is
SHA256:1xZuV8NMeVsNQPRrzVf9nXHzdUP+EtgVouZVbWH80XA.

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.

analyst@localhost's password:

Last login: Sat May 23 10:18:47 2020Stop the Wireshark capture.
```

c. Apply an SSH filter on the Wireshark capture data. Enter ssh in the filter field and click **Apply.**
d. Right-click one of the **SSHv2** lines in the **Packet list** section of Wireshark, and in the drop-down list, select the **Follow > TCP Stream**.
e. Examine the **Follow TCP Stream** window of your SSH session. The data has been encrypted and is unreadable. Compare the data in your SSH session to the data of your Telnet session.



f. After examining your SSH session, click Close.

g. Close Wireshark.

Practical 5
Investigating an Attack on a Windows Host

This lab is based on an exercise from the website malware-traffic-analysis.net which is an excellent resource for learning how to analyze network and host attacks.
You will use Sguil to check the IDS alerts and gather more information about the series of events related to an attack

In Sguil, click the first of the alerts on 3-19-2019 (Alert ID 5.439). Make sure to check the **Show Packet Data** and **Show Rule** checkboxes to examine the packet header information and the IDS signature rule related to the alert. Right on the **Alert ID** and pivot to Wireshark. Based on the information derived from this initial alert answer the following questions:
What was the source IP address and port number and destination IP address and port number?

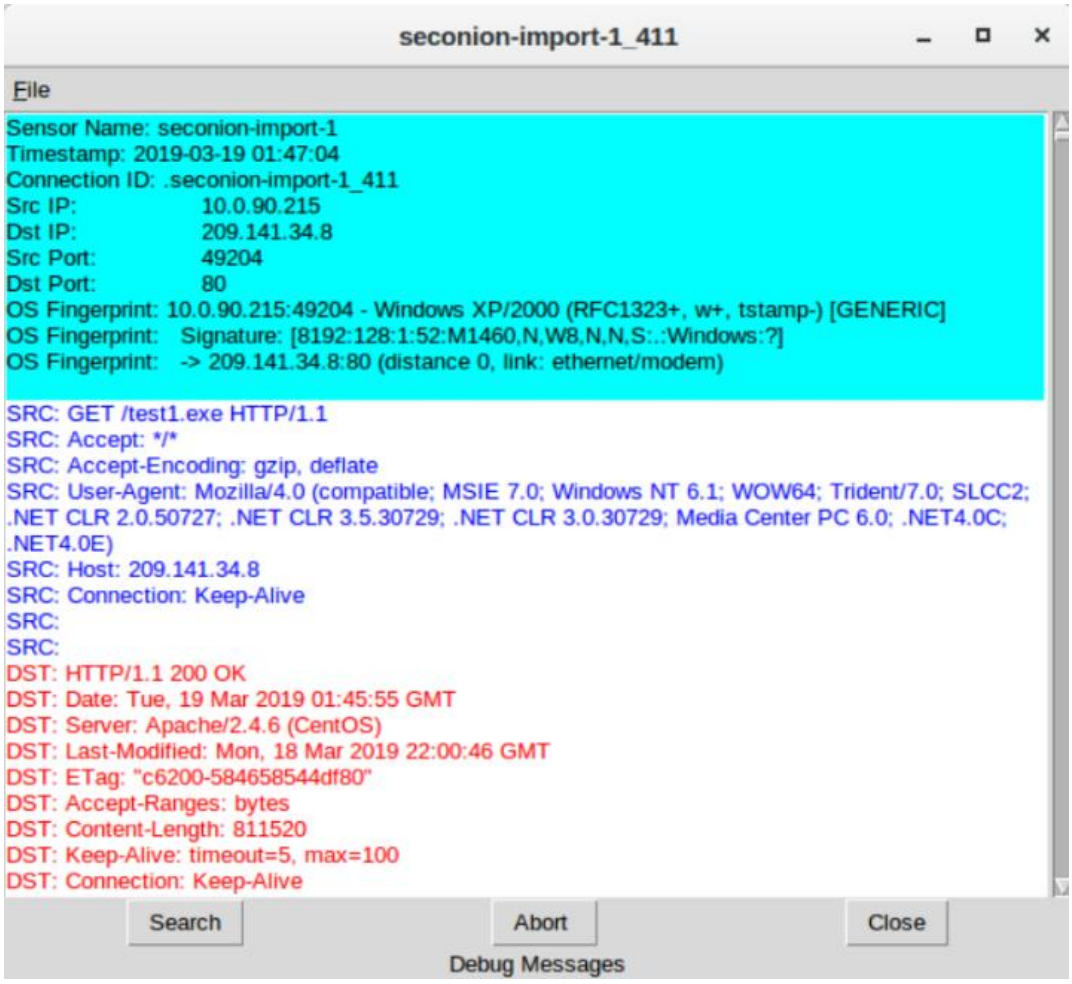
Source: 10.0.90.215:52609, Destination: 10.0.90.9:53
What type of protocol and request or response was involved?

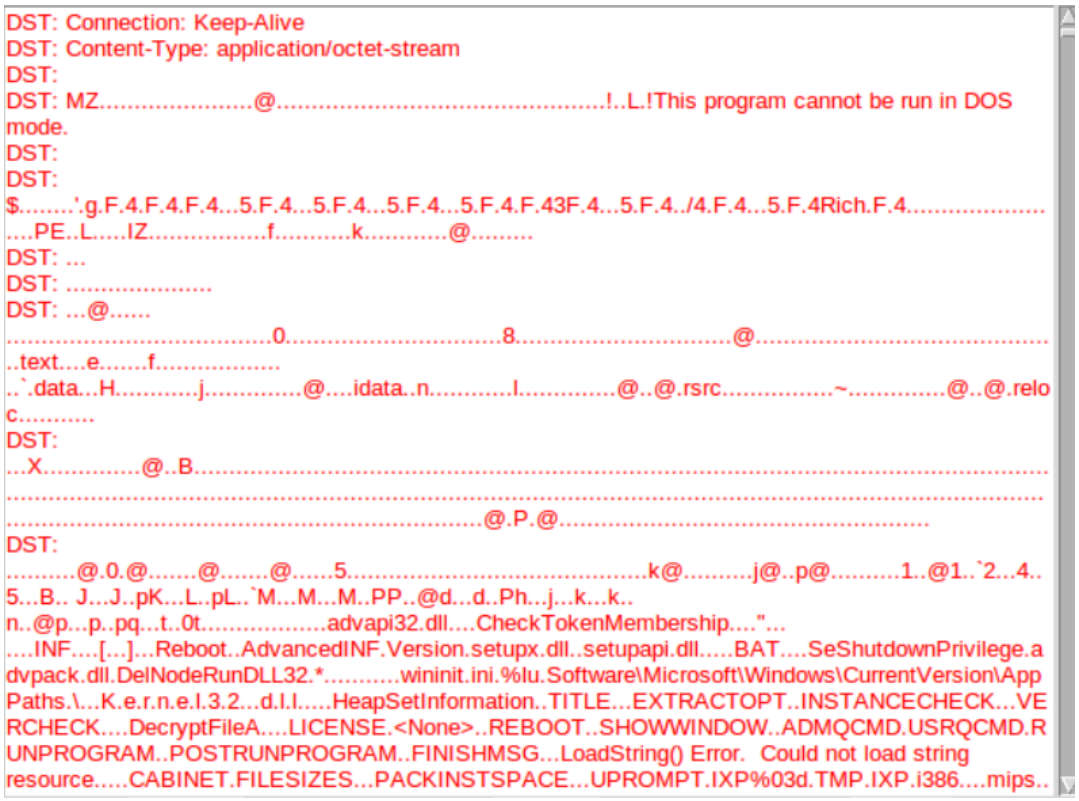
UDP, Dynamic DNS, update and response
What is the IDS alert and message?

Alert udp $EXTERNAL_NET any -> $HOME_NET 53, msg: “ET POLICY DNS Update from External net
Do you think this alert was the result of an IDS misconfiguration or a legitimate suspicious communication?

This alert may be the result of a misconfiguration in the IDS because the DNS request was a Dynamic DNS update from an internal host to a DNS server on the internal network and not from an external network to the internal network.
What is the hostname, domain name, and IP address of the source host in the DNS update?

Bobby-Tiger-PC, littletigers.info, 10.0.90.215
b. In Sguil, select the second of the alerts on 3-19-2019. Right click the Alert ID 5.440 and select **Transcript**.


seconion-import-1_411
File
Sensor Name: seconion-import-1
Timestamp: 2019-03-19 01:47:04
Connection ID: .seconion-import-1_411
Src IP: 10.0.90.215
Dst IP: 209.141.34.8
Src Port: 49204
Dst Port: 80
OS Fingerprint: 10.0.90.215:49204 - Windows XP/2000 (RFC1323+, w+, tstamp-) [GENERIC]
OS Fingerprint: Signature: [8192:128:1:52:M1460,N,W8,N,N,S:.:Windows:?]
OS Fingerprint: -> 209.141.34.8:80 (distance 0, link: ethernet/modem)
SRC: GET /test1.exe HTTP/1.1
SRC: Accept: */*
SRC: Accept-Encoding: gzip, deflate
SRC: User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; WOW64; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E)
SRC: Host: 209.141.34.8
SRC: Connection: Keep-Alive
SRC:
SRC:
DST: HTTP/1.1 200 OK
DST: Date: Tue, 19 Mar 2019 01:45:55 GMT
DST: Server: Apache/2.4.6 (CentOS)
DST: Last-Modified: Mon, 18 Mar 2019 22:00:46 GMT
DST: ETag: "c6200-584658544df80"
DST: Accept-Ranges: bytes
DST: Content-Length: 811520
DST: Keep-Alive: timeout=5, max=100
DST: Connection: Keep-Alive
Search
Abort
Close
Debug Messages


DST: Connection: Keep-Alive
DST: Content-Type: application/octet-stream
DST:
DST: MZ.....@.....!..L.!This program cannot be run in DOS mode.
DST:
DST:
$.....'.g.F.4.F.4.F.4...5.F.4...5.F.4...5.F.4...5.F.4.F.43F.4...5.F.4../4.F.4...5.F.4Rich.F.4.....
....PE..L.....IZ.....f.....k.....@.....
DST: ...
DST: .....
DST: ...@.....
.....0.....8.....@.....
..text....e.....f.....
..`.data...H.....j.....@....idata..n.....I.....@..@.rsrc.....~.....@..@.reloc.....
DST:
...X.....@..B.....
.....
.....@.P.@.....
DST:
.....@.0.@.....@.....@.....5.....k@.....j@..p@.....1..@1..`2...4..5...B.. J...J..pK...L..pL..`M...M...M..PP..@d...d..Ph...j...k...k..
n..@p...p..pq...t..0t.....advapi32.dll....CheckTokenMembership....".....
....INF....[...]...Reboot..AdvancedINF.Version.setupx.dll..setupapi.dll.....BAT....SeShutdownPrivilege.advpack.dll.DelNodeRunDLL32.*.....wininit.ini.%lu.Software\Microsoft\Windows\CurrentVersion\App Paths.\...K.e.r.n.e.l.3.2...d.l.l.....HeapSetInformation..TITLE...EXTRACTOPT..INSTANCECHECK...VERCHECK....DecryptFileA....LICENSE.<None>..REBOOT..SHOWWINDOW..ADMQCMD.USRQCMD.RUNPROGRAM..POSTRUNPROGRAM..FINISHMSG...LoadString() Error. Could not load string resource.....CABINET.FILESIZES...PACKINSTSPACE...UPROMPT.IXP%03d.TMP.IXP.i386....mips..

From the transcript answer the following questions:

- **What is the source and destination IP address and port numbers?**

Source 10.0.90.215:49204 and Destination 209.141.34.8:80

- **Looking at the request (blue) what was the request for?**

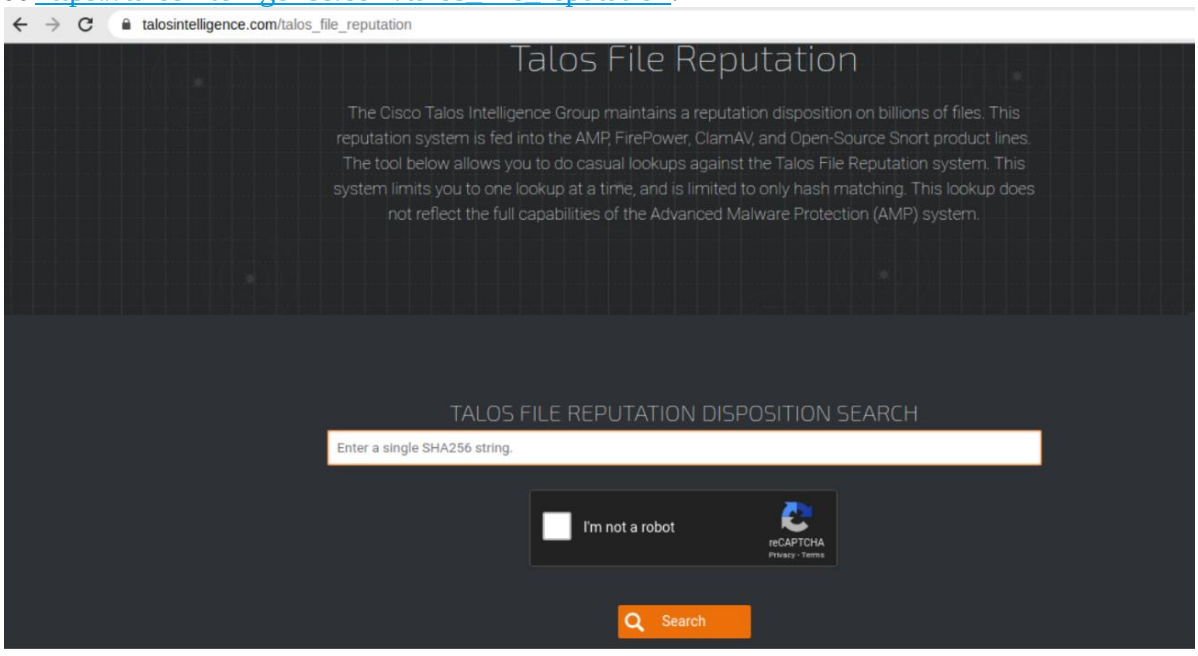
GET /test1.exe
Looking at the reply (red) many files will reveal their file signature in the initial few characters of the file when viewed as text. File signatures help identify the type of file that is represented. Use a web browser to search for a list of common file signatures.

- **What is the initial few characters of the file file. Search for this file signature to find out what type of file was downloaded in the data?**

The initial characters of this fiel is MZ, a Windows executable .exe or .dll file
c. Close the transcript. Use Wireshark to export the executable file for malware analysis (**File > Export Objects > HTTP…**). Save the file to the analyst's home folder.
d. Open a terminal in Security Onion VM and create a SHA256 hash from the exported file. Use the following command:

```
analyst@SecOnion:~$ sha256sum test1.exe

2a9b0ed40f1f0bc0c13ff35d304689e9cadd633781cbcad1c2d2b92ced3f1c85 test1.exe
```

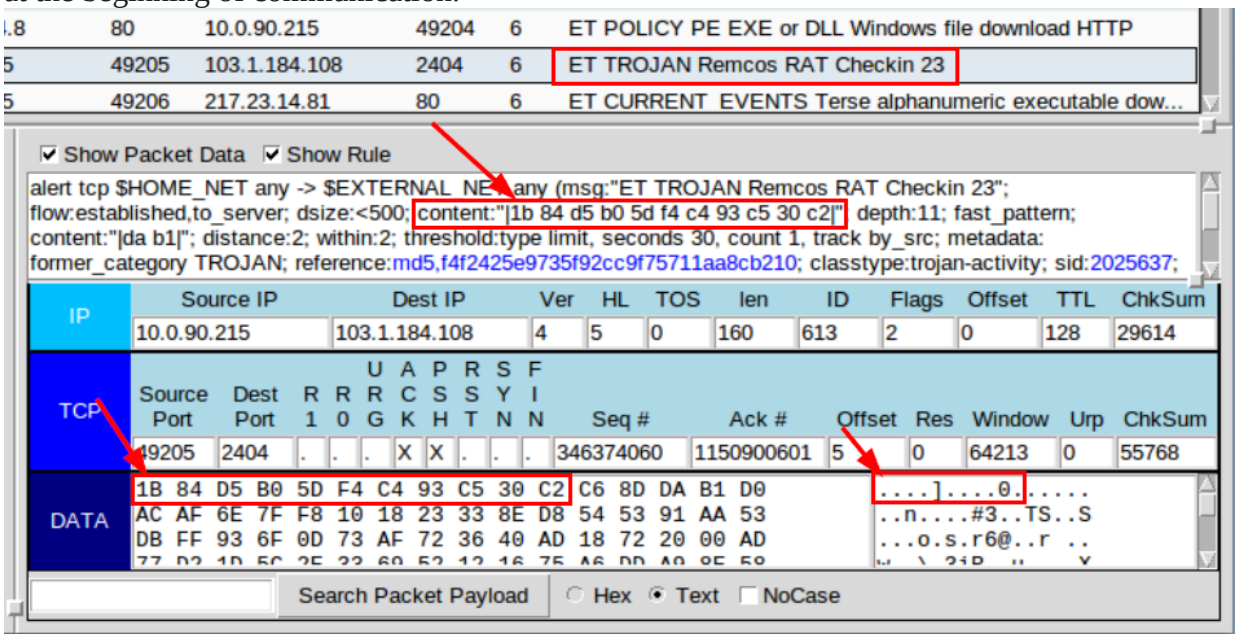
e. Copy the file hash and submit it to the Cisco Talos file reputation center at https://talosintelligence.com/talos_file_reputation.



- **Did Talos recognize the file hash and identify it as malware? If so, what kind of malware?**

Yes, win32 trojan-spy-agent

f. In Sguil select the alert with **Alert ID 5.480** and the **Event Message** Remcos RAT Checkin 23. Notice that the IDS signature has detected the Remcos RAT based on the binary hex codes at the beginning of communication.



g. Right click the Alert ID and select Transcript. Scroll through the transcript and answer the following questions:

- **What is the destination port of the communication? Is it a well-known port?**

The destination port is 2404 and it is not a well-known port.
Is the communication readable or is it encrypted?

It is encrypted

- **Do some online research on Remcos RAT Checkin 23. What does Remcos stand for?**

Remote control and surveillance software

- **What type of communication do you think was being transmitted?**

A keylogger possibly sending keystroke information to a C2C server

- **What type of encryption and obfuscation was used to bypass detection?**

Remcos RAT uses multiple packers, base64 encoding and RC4 encryption to bypass detection and throw off security analysts
h. Using Sguil and the remaining alerts from 3-19-2019, locate the second executable file that was downloaded and check to see if it is known malware.

- **What Alert IDs alert to a second executable file being downloaded?**

Answers may vary. In this example, 5.483, 5.485, 5.497, 5.509, 5.521, 5.533
From which server IP address and port number was the file downloaded from?

217.23.14.81:80

- **What is the name of the file that was downloaded?**

F4.exe

Create a SHA256 hash of the file and submit the hash online at Cisco Talos File Reputation Center to see if it matches known malware. Is the executable file known malware and if so, what type? What is the AMP DETECTION NAME?

Yes, PE32 executable, trojan downloader Win.Dropper.Cridex::1201
i. Examine the remaining three alerts from 3-19-2019 by looking at the header information in Show Packet Data, the IDS signature in Show Rule, and the Alert ID Transcripts.

- **How are all three alerts related?**

All three alerts are encrypted and all three alerts were triggered by a blacklisted malicious SSL certificate – Dridex
j. Even though you have examined all the alerts in Sguil related to an attack on a Windows host on 3-19-2019, there may be additional related information available in Kibana. Close Sguil and launch Kibana from the desktop.

Practical 6

Using Wireshark to Examine HTTP and HTTPS Traffic

HyperText Transfer Protocol (HTTP) is an application layer protocol that presents data via a web browser. With HTTP, there is no safeguard for the exchanged data between two communicating devices.

With HTTPS, encryption is used via a mathematical algorithm. This algorithm hides the true meaning of the data that is being exchanged. This is done through the use of certificates that can be viewed later in this lab.

Regardless of HTTP or HTTPS, it is only recommended to exchange data with websites that you trust. Just because a site uses HTTPS does not mean it is a trustworthy site. Threat actors commonly use HTTPS to hide their activities.

In this lab, you will explore and capture HTTP and HTTPS traffic using Wireshark.

- **Part 1: Capture and view HTTP traffic**

In this part, you will use tcpdump to capture the content of HTTP traffic. You will use command options to save the traffic to a packet capture (pcap) file. These records can then be analyzed using different applications that read pcap files, including Wireshark.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default ql
en 1000
    link/ether 00:0c:29:c2:e8:a6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.137.138/24 brd 192.168.137.255 scope global dynamic noprefixroute eth0
       valid_lft 1713sec preferred_lft 1713sec
    inet6 fe80::9db9:3510:1150:a1d6/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~/Desktop]
└─$ sudo tcpdump -i eth0 -s 0 -w httpdump.pcap
[sudo] password for kali:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C2509 packets captured
2510 packets received by filter
0 packets dropped by kernel

┌──(kali㉿kali)-[~/Desktop]
└─$ ls
EncDec  httpdump.pcap
```

While in the terminal application, enter the command

$sudo tcpdump –i eth0 –s 0 –w httpsdump.pcap

Enter the password cyberops for the user analyst when prompted.

This command will start tcpdump and record network traffic on the **eth0** interface of the Linux workstation.

- The -i command option allows you to specify the interface.
- The -s command option specifies the length of the snapshot for each packet.
- The -w command option is used to write the result of the tcpdump command to a file.

Open a web browser from the launch bar within the Workstation VM. Navigate to http://www.altoromutual.com/login.jsp

Enter a username of **Admin** with a password of **Admin** and click **Login**.

Logout and close the web browser.

Return to the terminal window where tcpdump is running. Enter **CTRL+C** to stop the packet capture.

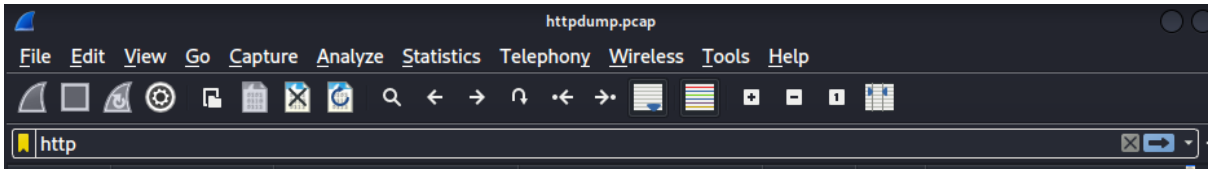
All recorded traffic will be printed to the file httpsdump.pcap in the home directory of the user analyst.

View the HTTP capture.

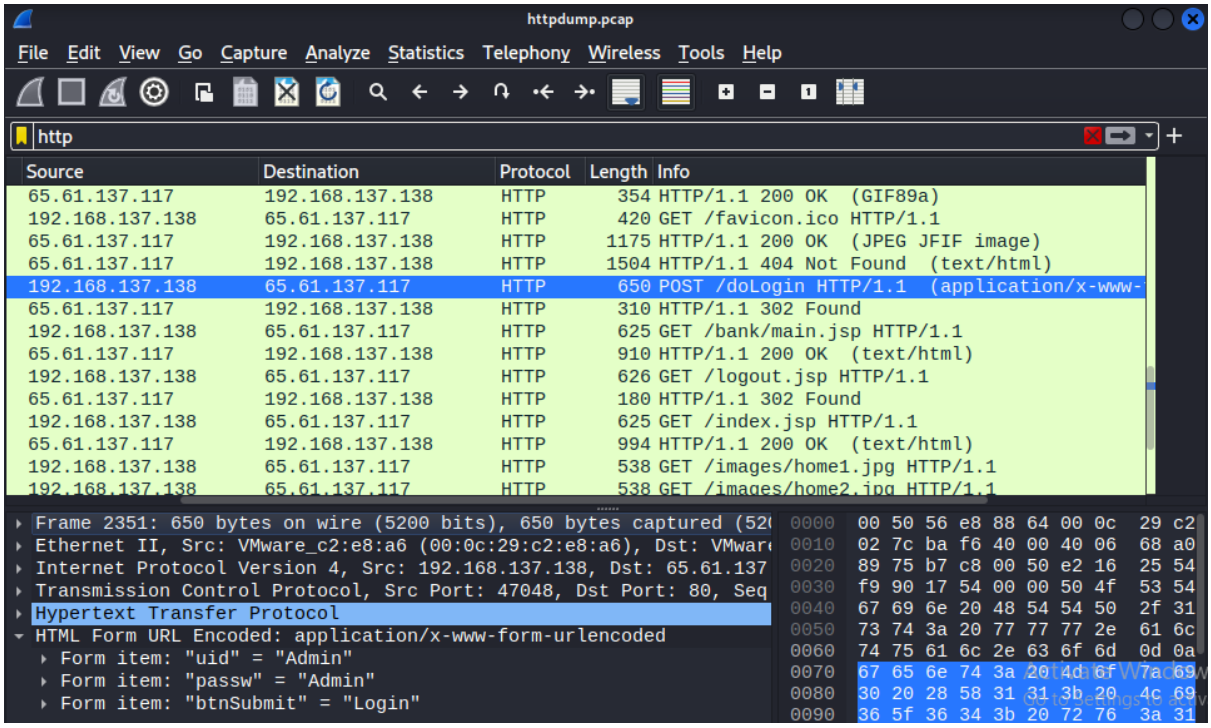
Double-click the httpdump.pcap file, in the Open With dialog box scroll down to Wireshark and then click Open.

In the Wireshark application, filter for http and click Apply.

Browse through the different HTTP messages and select the POST message.



In the lower window, the message is displayed. Expand the HTML Form URL Encoded: **application/x-www-form-urlencoded** section.



What two pieces of information are displayed?

The uid of Admin and passw of Admin

Close the Wireshark application.

- **Part 2: Capture and view HTTPS traffic**

You will now use tcpdump from the command line of a Linux workstation to capture HTTPS traffic. After starting tcpdump, you will generate HTTPS traffic while tcpdump records the contents of the network traffic. These records will again be analyzed using Wireshark.

While in the terminal application, enter the command

$sudo tcpdump –i eth0 –s 0 –w httpsdump.pcap.

Enter the password for the user when prompted.

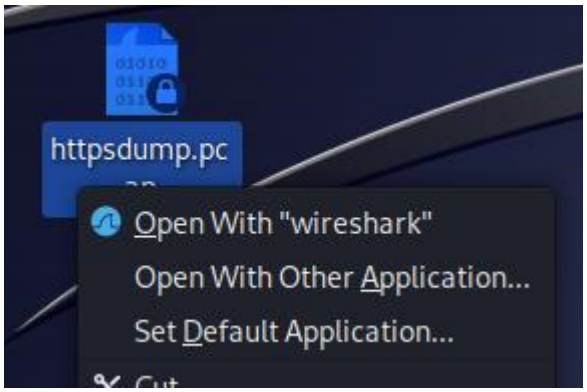
```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo tcpdump -i eth0 -s 0 -w httpsdump.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

This command will start tcpdump and record network traffic on the **eth0** interface of the Linux workstation.

All recorded traffic will be printed to the file **httpsdump.pcap** in the home directory of the user.

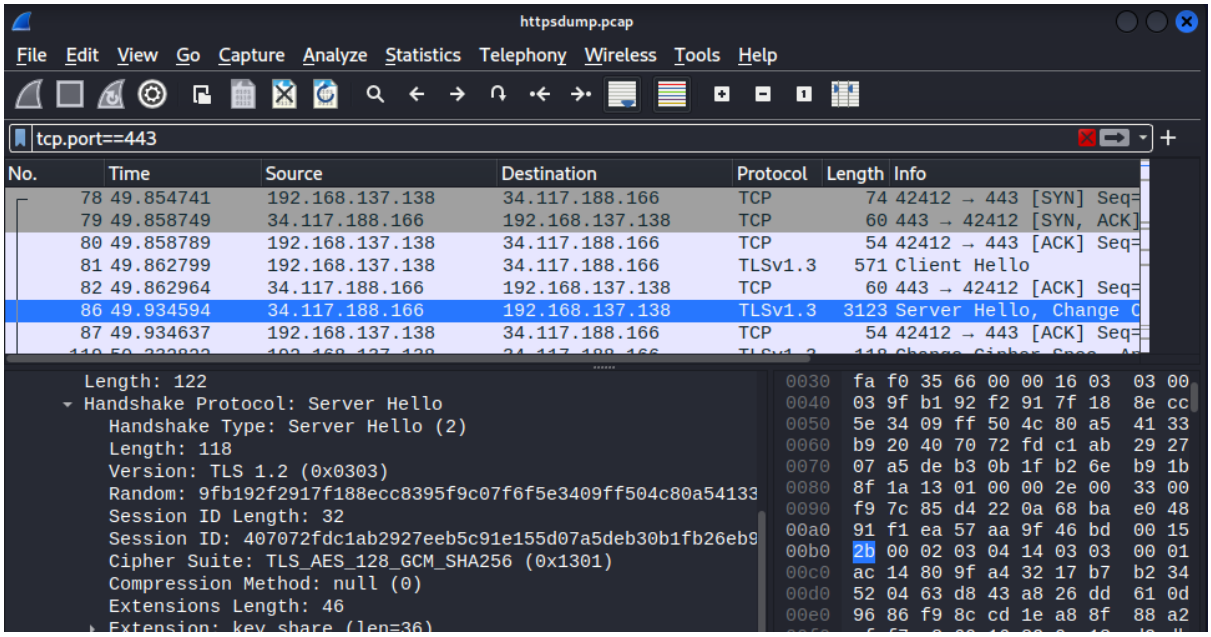
Open a web browser from the launch bar within the Workstation VM. Navigate to www.securewebpage.com.

Login with your username and password



In the Wireshark application, expand the capture window vertically and then filter by HTTPS traffic via port 443.

Enter tcp.port==443 as a filter, and click Apply.



Browse through the different HTTPS messages and select an Application Data message.

In the lower window, the message is displayed.

What has replaced the HTTP section that was in the previous capture file?

After the TCP section, there is now a Secure Sockets Layer (SSL/TLS 1.2) section instead of HTTP.

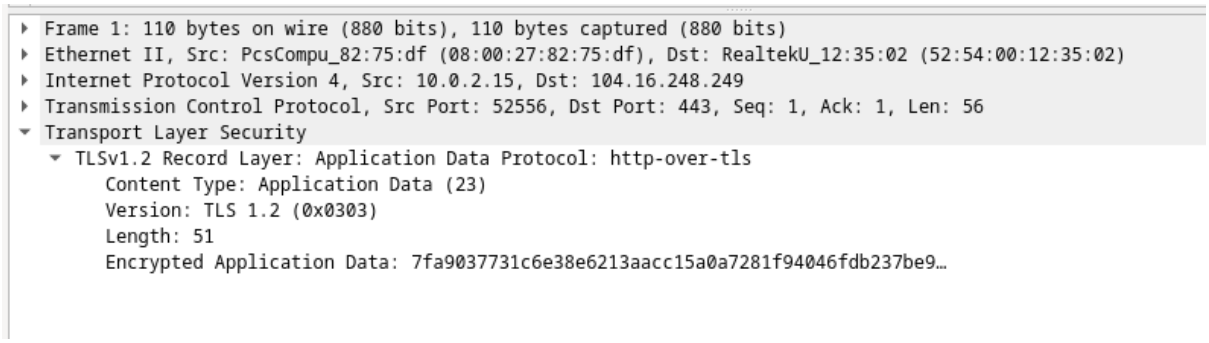
Completely expand the **Secure Sockets Layer** section.

Click the **Encrypted Application Data**.

Is the application data in a plaintext or readable format?

The data payload is encrypted using TLSv1.2 and cannot be viewed.

Close all windows and shut down the virtual machine.



Practical 7
Exploring Processes, Threads, Handles, and Windows Registry.

In this lab, you will explore the processes, threads, and handles using Process Explorer in the SysInternals Suite. You will also use the Windows Registry to change a setting. Part 1: Exploring Processes Part 2: Exploring Threads and Handles Part 3: Exploring Windows Registry

Part 1: Exploring Processes

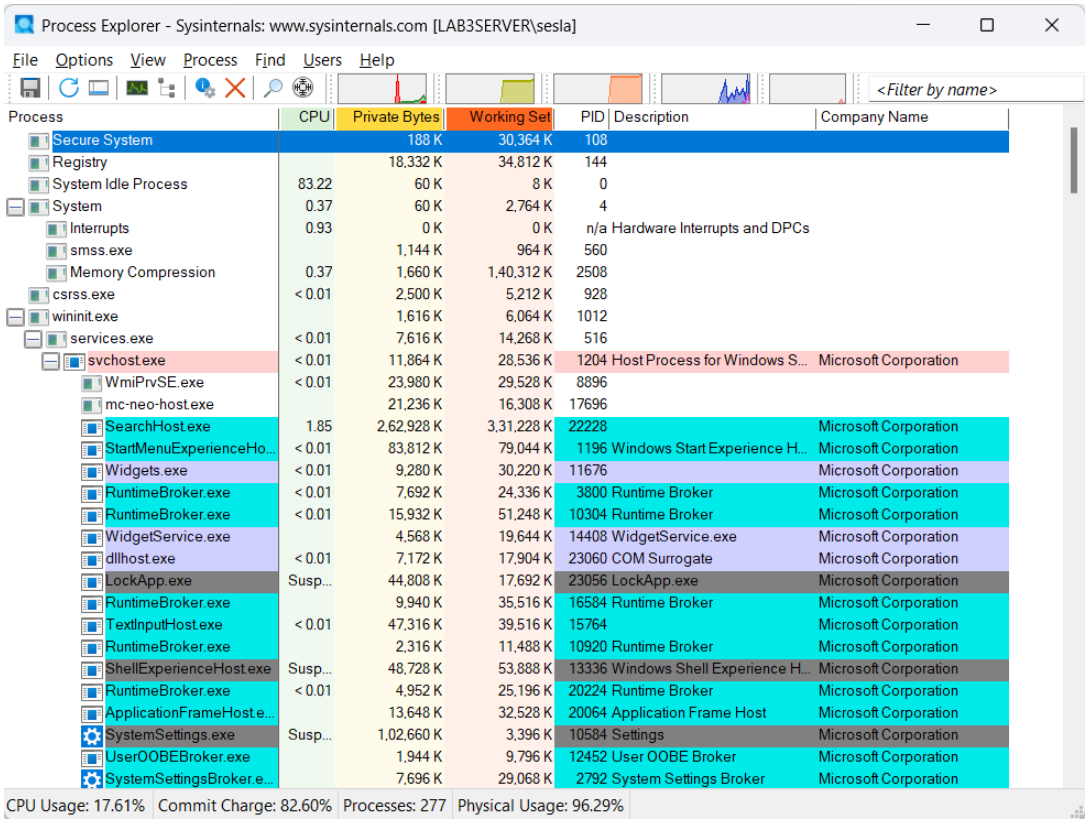
In this part, you will explore processes. Processes are programs or applications in execution. You will explore the processes using Process Explorer in the Windows SysInternals Suite. You will also start and observe a new process.

Step 1: Download Windows SysInternals Suite.

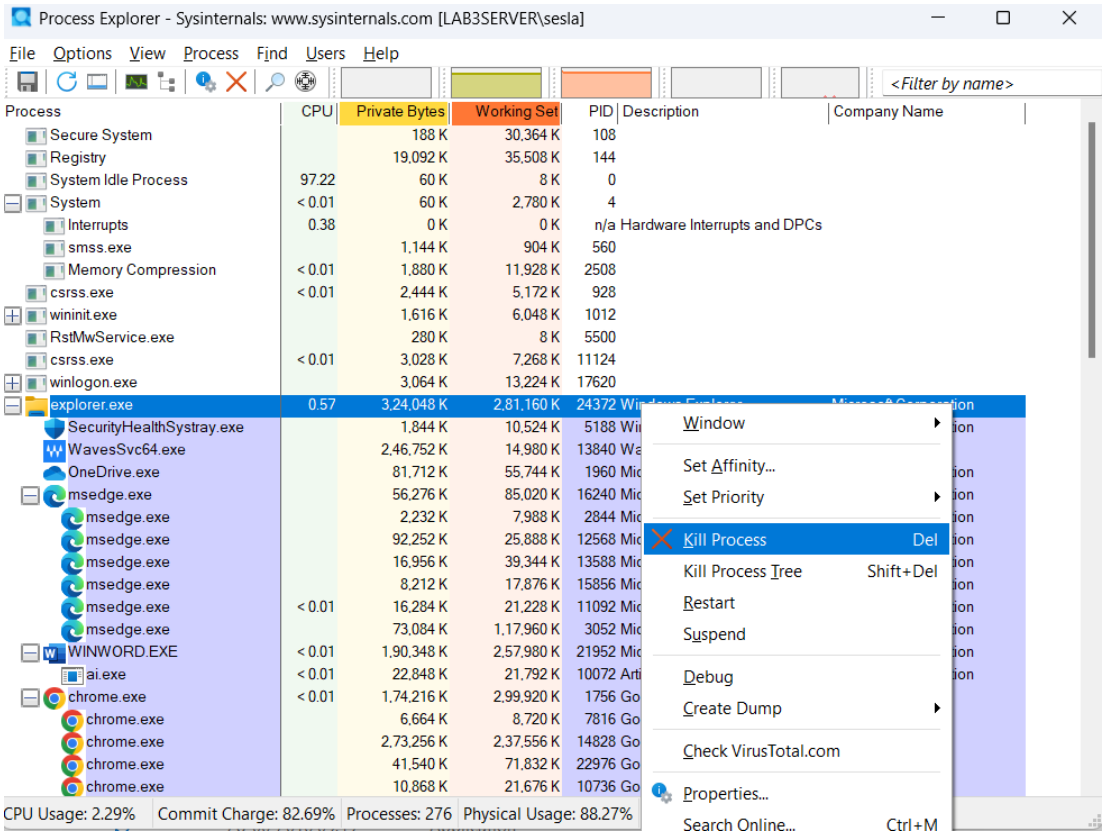
a) Navigate to the following link to download Windows SysInternals Suite: https://technet.microsoft.com/en-us/sysinternals/bb842062.aspx
b) After the download is completed, extract the files from the folder.
c) Leave the web browser open for the following steps.

Step 2: Explore an active process.

a) Navigate to the SysinternalsSuite folder with all the extracted files.
b) Open procexp.exe. Accept the Process Explorer License Agreement when prompted.
c) The Process Explorer displays a list of currently active processes.
d) To locate the web browser process, drag the Find Window's Processicon into the opened web browser window. Microsoft Edge was used in this example.



e) The Microsoft Edge process can be terminated in the Process Explorer. Right-click the selected process and select Kill Process. Click OKto continue.



What happened to the web browser window when the process is killed? Stopped and closed the program

Step 3: Start another process.

a) Open a Command Prompt. (Start > search Command Prompt> select Command Prompt)
b) Drag the Find Window's Processicon into the Command Prompt window and locate the highlighted Command Prompt process in Process Explorer.
c) The process for the Command Prompt is cmd.exe. Its parent process is explorer.exe process. The cmd.exe has a child process, conhost.exe.
d) Navigate to the Command Prompt window. Start a ping at the prompt and observe the changes under the cmd.exe process.
Question: What happened during the ping process?
PING.EXE open up under cmd.exe Type your answers here.
e) As you review the list of active processes, you find that the child process conhost.exe may be suspicious. To check for malicious content, right-click conhost.exe and select Check VirusTotal. When prompted, click Yes to agree to VirusTotal Terms of Service. Then click OKfor the next prompt.
f) Expand the Process Explorer window or scroll to the right until you see the VirusTotal column. Click the link under the VirusTotal column. The default web browser opens with the results regarding the malicious content of conhost.exe.
g) Right-click the cmd.exe process and select Kill Process.

Question: What happened to the child process conhost.exe? It also closed.

Part 2: Exploring Threads and Handles

In this part, you will explore threads and handles. Processes have one or more threads. A thread is a unit of execution in a process. A handle is an abstract reference to memory blocks or objects managed by an operating system. You will use Process Explorer (procexp.exe) in Windows SysInternals Suite to explore the threads and handles.

Step 1: Explore threads.

a) Open a command prompt.
b) In Process Explorer window, right-click conhost.exe and Select Properties..... Click the Threads tab to view the active threads for the conhost.exe process. Click OK to continue if prompted by a warning dialog box.
c) Examine the details of the thread.
Question: What type of information is available in the Properties window?
Thread ID, stat time, stat. kernel time, user time cyclesType your answers here.
d) Click OKto continue.

Step 2: Explore handles.

a) In the Process Explorer, click View> select Lower Pane View > Handlesto view the handles associated with the conhost.exe process.
Examine the handles. What are the handles pointing to? Files, keys, processesType your answers here.
b) Close the Process Explorer when finished.

Part 3: Exploring Windows Registry

The Windows Registry is a hierarchical database that stores most of the operating systems and desktop environment configuration settings.

a) To access the Windows Registry, click Start> Search for regeditand select Registry Editor. Click Yes when asked to allow this app to make changes. The Registry Editor has five hives. These hives are at the top level of the registry.

- HKEY_CLASSES_ROOT is actually the Classes subkey of HKEY_LOCAL_MACHINE\Software\. It stores information used by registered applications like file extension association, as well as a programmatic identifier (ProgID), Class ID (CLSID), and Interface ID (IID) data.
- HKEY_CURRENT_USER contains the settings and configurations for the users who are currently logged in.
- HKEY_LOCAL_MACHINE stores configuration information specific to the local computer.
- HKEY_USERS contains the settings and configurations for all the users on the local computer.
- HKEY_CURRENT_USER is a subkey of HKEY_USERS.
- HKEY_CURRENT_CONFIG stores the hardware information that is used at bootup by the local computer.

b) b.In a previous step, you had accepted the EULA for Process Explorer. Navigate to the EulaAccepted registry key for Process Explorer. Click to select Process Explorer in HKEY_CURRENT_USER> Software> Sysinternals> Process Explorer.

Scroll down to locate the key EulaAccepted. Currently, the value for the registry key EulaAccepted is 0x00000001(1).

c) Double-click EulaAcceptedregistry key. Currently the value data is set to 1. The value of 1 indicates that the EULA has been accepted by the user.

d) Change the 1to 0for Value data. The value of 0 indicates that the EULA was not accepted. Click OK to continue.

Question: What is value for this registry key in the Data column?
Type your answers here 0

e) Open the Process Explorer. Navigate to the folder where you have downloaded SysInternals. Open the folder SysInternalsSuite> Open procexp.exe.

Question: When you open the Process Explorer, what did you see? Nothing

Practical 8

Perform a practical to Attack on a MySQL Database by using PCAP file.

SQL injection attacks allow malicious hackers to type SQL statements in a web site and receive a response from the database. This allows attackers to tamper with current data in the database, spoof identities, and miscellaneous mischief.

A PCAP file has been created for you to view a previous attack against a SQL database. In this lab, you will view the SQL database attacks and answer the questions.

Part 1: Open the PCAP file and follow the SQL database attacker

You will use Wireshark, a common network packet analyzer, to analyze network traffic. After starting Wireshark, you will open a previously saved network capture and view a step by step SQL injection attack against a SQL database.

Step 1: Open Wireshark and load the PCAP file.

The Wireshark application can be opened using a variety of methods on a Linux workstation.

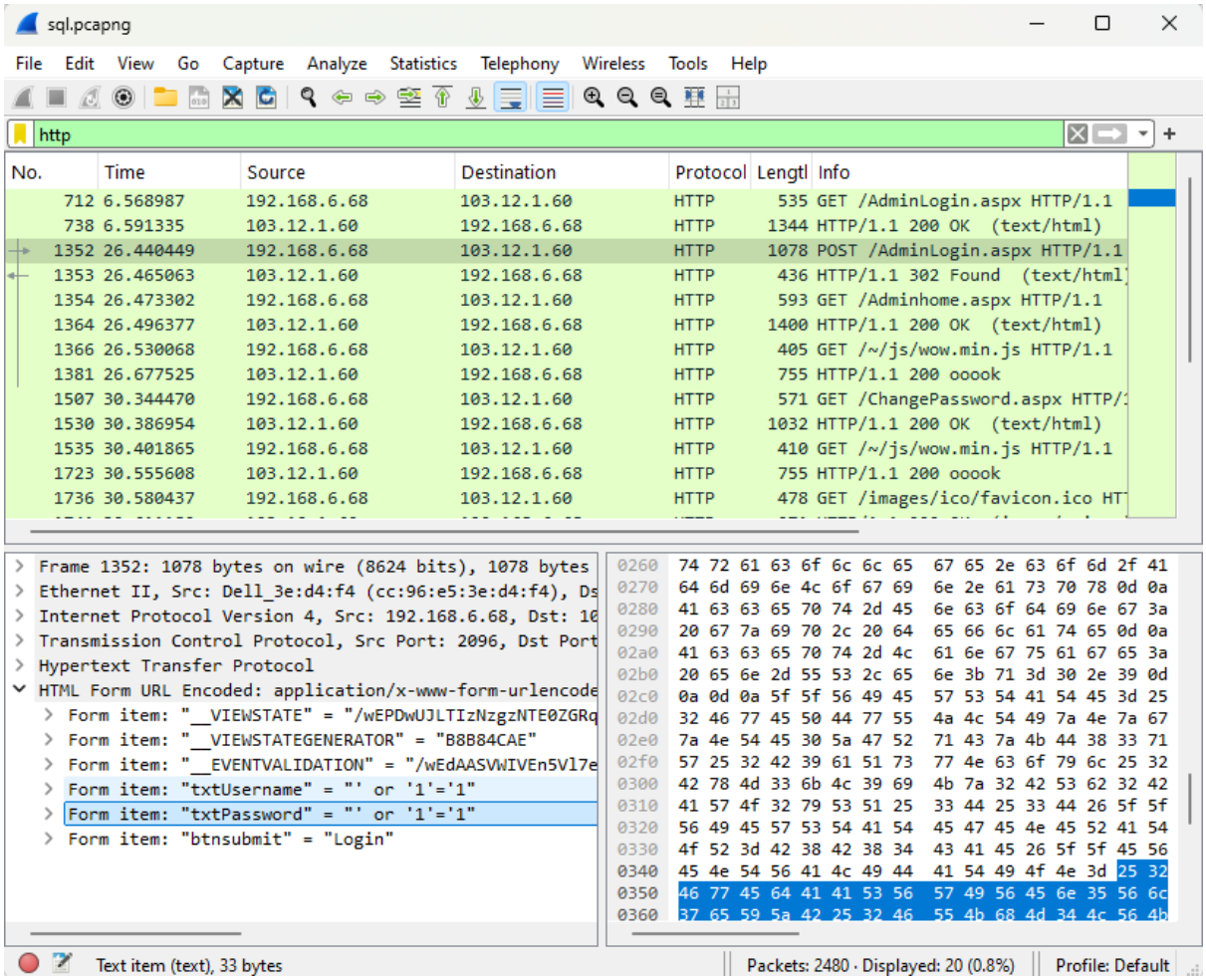
a) Start the Workstation VM.
b) Browse to the Wireshark application.
c) In the Wireshark application, click **Open** in the middle of the application under Files.
d) Open the **SQL_Lab.pcap** file.

What are the two IP addresses involved in this SQL injection attack based on the information displayed?
103.12.1.60

Step 2: View the SQL Injection Attack.

a) In this step, you will be viewing the beginning of an attack.

b) Within the Wireshark capture, right-click line 4 and select **Follow HTTP Stream**. Line 4 was chosen because it is a GET HTTP request. This will be very helpful in following the data stream as the application layers sees it and leads up to the query testing for the SQL injection.
c) Click **Find** and enter 1=1. Search for this entry. When the text is located, click **Cancel** in the Find text search box. The string 1=1
d) The attacker has entered a query (1=1) into a UserID search box on the target 103.12.1.60 to see if the application is vulnerable to SQL injection. Instead of the application responding with a login failure message, it responded with a record from a database. The attacker has verified they can input an SQL command and the database will respond. The search string 1=1 creates an SQL statement that will be always true. In the example, it does not matter what is entered into the field, it will always be true.
e) Close the Follow HTTP Stream window.
f) Click **Clear** to display the entire Wireshark conversation.


sql.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
No. Time Source Destination Protocol Lengtl Info
712 6.568987 192.168.6.68 103.12.1.60 HTTP 535 GET /AdminLogin.aspx HTTP/1.1
738 6.591335 103.12.1.60 192.168.6.68 HTTP 1344 HTTP/1.1 200 OK (text/html)
1352 26.440449 192.168.6.68 103.12.1.60 HTTP 1078 POST /AdminLogin.aspx HTTP/1.1
1353 26.465063 103.12.1.60 192.168.6.68 HTTP 436 HTTP/1.1 302 Found (text/html)
1354 26.473302 192.168.6.68 103.12.1.60 HTTP 593 GET /Adminhome.aspx HTTP/1.1
1364 26.496377 103.12.1.60 192.168.6.68 HTTP 1400 HTTP/1.1 200 OK (text/html)
1366 26.530068 192.168.6.68 103.12.1.60 HTTP 405 GET /~/js/wow.min.js HTTP/1.1
1381 26.677525 103.12.1.60 192.168.6.68 HTTP 755 HTTP/1.1 200 ooook
1507 30.344470 192.168.6.68 103.12.1.60 HTTP 571 GET /ChangePassword.aspx HTTP/1
1530 30.386954 103.12.1.60 192.168.6.68 HTTP 1032 HTTP/1.1 200 OK (text/html)
1535 30.401865 192.168.6.68 103.12.1.60 HTTP 410 GET /~/js/wow.min.js HTTP/1.1
1723 30.555608 103.12.1.60 192.168.6.68 HTTP 755 HTTP/1.1 200 ooook
1736 30.580437 192.168.6.68 103.12.1.60 HTTP 478 GET /images/ico/favicon.ico HT
Frame 1352: 1078 bytes on wire (8624 bits), 1078 bytes
Ethernet II, Src: Dell_3e:d4:f4 (cc:96:e5:3e:d4:f4), Ds
Internet Protocol Version 4, Src: 192.168.6.68, Dst: 10
Transmission Control Protocol, Src Port: 2096, Dst Port
Hypertext Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urlencode
Form item: "__VIEWSTATE" = "/wEPDwUJLTIzNzgzNTE0ZGRq
Form item: "__VIEWSTATEGENERATOR" = "B8B84CAE"
Form item: "__EVENTVALIDATION" = "/wEdAASVWIVEn5Vl7e
Form item: "txtUsername" = "' or '1'='1"
Form item: "txtPassword" = "' or '1'='1"
Form item: "btnsubmit" = "Login"
0260 74 72 61 63 6f 6c 6c 65 67 65 2e 63 6f 6d 2f 41
0270 64 6d 69 6e 4c 6f 67 69 6e 2e 61 73 70 78 0d 0a
0280 41 63 63 65 70 74 2d 45 6e 63 6f 64 69 6e 67 3a
0290 20 67 7a 69 70 2c 20 64 65 66 6c 61 74 65 0d 0a
02a0 41 63 63 65 70 74 2d 4c 61 6e 67 75 61 67 65 3a
02b0 20 65 6e 2d 55 53 2c 65 6e 3b 71 3d 30 2e 39 0d
02c0 0a 0d 0a 5f 5f 56 49 45 57 53 54 41 54 45 3d 25
02d0 32 46 77 45 50 44 77 55 4a 4c 54 49 7a 4e 7a 67
02e0 7a 4e 54 45 30 5a 47 52 71 43 7a 4b 44 38 33 71
02f0 57 25 32 42 39 61 51 73 77 4e 63 6f 79 6c 25 32
0300 42 78 4d 33 6b 4c 39 69 4b 7a 32 42 53 62 32 42
0310 41 57 4f 32 79 53 51 25 33 44 25 33 44 26 5f 5f
0320 56 49 45 57 53 54 41 54 45 47 45 4e 45 52 41 54
0330 4f 52 3d 42 38 42 38 34 43 41 45 26 5f 5f 45 56
0340 45 4e 54 56 41 4c 49 44 41 54 49 4f 4e 3d 25 32
0350 46 77 45 64 41 41 53 56 57 49 56 45 6e 35 56 6c
0360 37 65 59 5a 42 25 32 46 55 4b 68 4d 34 4c 56 4b
Text item (text), 33 bytes
Packets: 2480 · Displayed: 20 (0.8%)
Profile: Default


Wireshark · Follow HTTP Stream (tcp.stream eq 23) · sql.pcapng
POST /AdminLogin.aspx HTTP/1.1
Host: www.sourashtracollege.com
Connection: keep-alive
Content-Length: 371
Cache-Control: max-age=0
Origin: [redacted]
Content-Type: application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/132.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://www.sourashtracollege.com/AdminLogin.aspx
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
__VIEWSTATE=%2FwEPDwUJLTIzNzgzNTE0ZGRqCzKD83qW%2B9aQswNcoy1%2BxM3kL9iKz2BSb2BAWO2ySQ%3D%3D&__VIEWSTATEGENERATOR=B8B84CAE&__EVENTVALIDATION=%2FwEdAASVWIVEn5Vl7eYZB%2FUKhM4LVK7B
rRAtEiqu9nGFEI%2BjB3Y2%2BMc6SrnAqio3oCKbxYYfeetr3c4H3XmikgJc2ZlIho21wWmHJtBv0kyOQEHMG%2B7PvFJ0eVsZ8etRVrUfrNY%3D&txtUsername=%27+or+%271%27%3D%271&txtPassword=%27+or+%271%27%3
D%271&btnsubmit=Login
HTTP/1.1 302 Found
Cache-Control: private
Content-Type: text/html; charset=utf-8
Location: /Adminhome.aspx
Server: Microsoft-IIS/8.5
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
Date: Sat, 01 Feb 2025 02:39:02 GMT
Content-Length: 132
<html><head><title>Object moved</title></head><body>
<h2>Object moved to here.</h2>
</body></html>

Practical 9

Create your own syslog Server, Configure your windows system to send syslog messages to a syslog server and Read them

First Steps

1. Open Syslog Watcher Manager

The Syslog Watcher Manager enables the administration and configuration of the Syslog Watcher server. It also allows viewing, analyzing, and exporting collected syslog messages.

2. (Optional) Activate Enterprise License

3. Add Network Interface

To start receiving syslog messages you need at least one network interface in the server configuration.

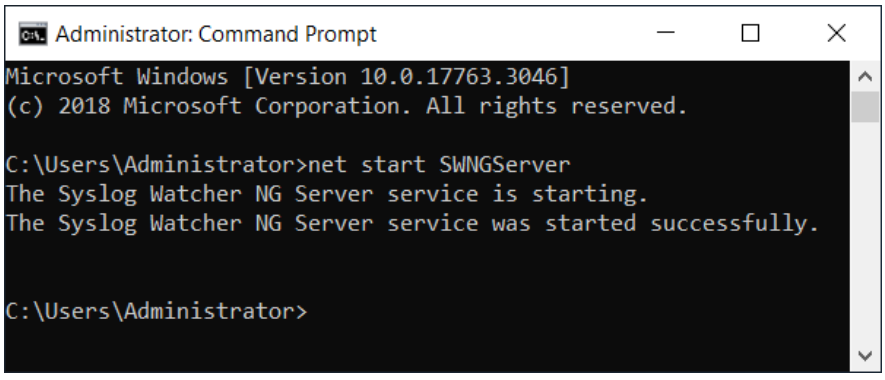
If you just installed Syslog Watcher, the most basic (Syslog over UDP port number 514) interface has already been added to the configuration. UDP/514 works well for many default-configured originators, but you may need to add network interfaces that are more appropriate for your specific situation. Later, we recommend switching to Syslog over TLS if your originators support it.

4. Start Syslog Watcher Server

Using Command Prompt

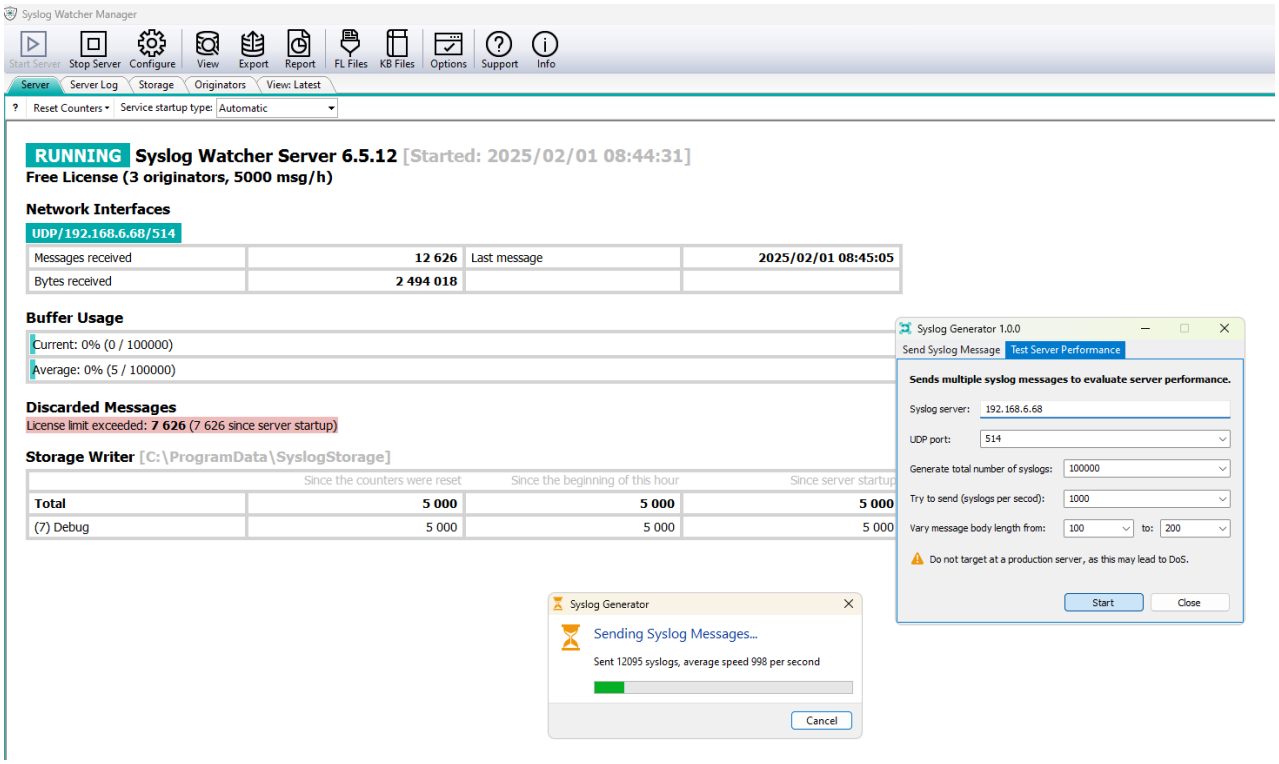
From the **Start** menu, right-click *Command Prompt*, select **More**, and select **Run as Administrator**.

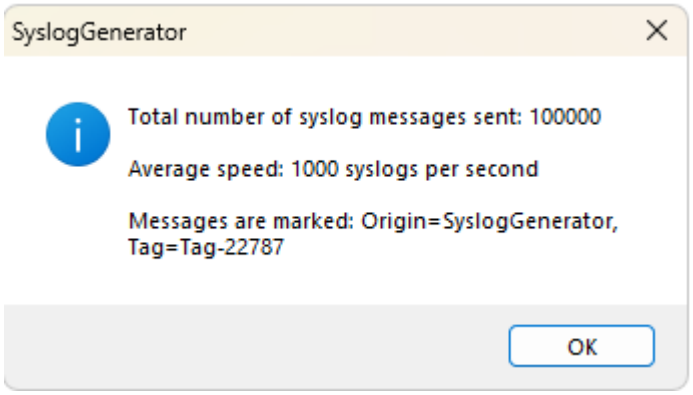
- Type net start SWNGServer to start the server.
- Type net stop SWNGServer to stop the server.

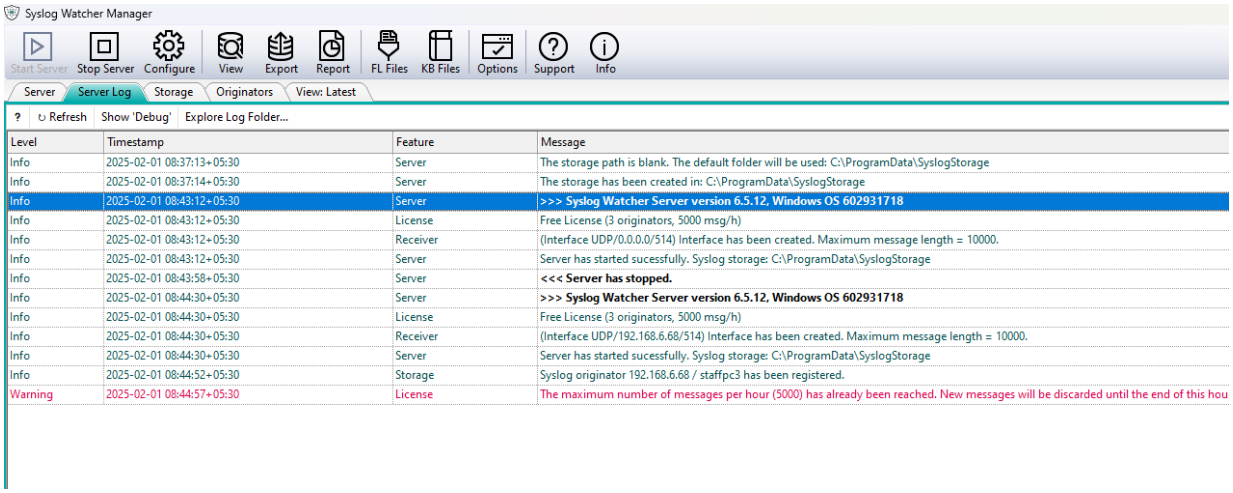


5. Get a Test Message

To make sure that the network interfaces are configured correctly, and the server can receive messages, you can first get a test syslog messages from SyslogGenerator.


Syslog Watcher Manager
Start Server
Stop Server
Configure
View
Export
Report
FL Files
KB Files
Options
Support
Info
Server
Server Log
Storage
Originators
View: Latest
Reset Counters
Service startup type: Automatic
RUNNING Syslog Watcher Server 6.5.12 [Started: 2025/02/01 08:44:31]
Free License (3 originators, 5000 msg/h)
Network Interfaces
UDP/192.168.6.68/514
Messages received 12 626 Last message 2025/02/01 08:45:05
Bytes received 2 494 018
Buffer Usage
Current: 0% (0 / 100000)
Average: 0% (5 / 100000)
Discarded Messages
License limit exceeded: 7 626 (7 626 since server startup)
Storage Writer [C:\ProgramData\SyslogStorage]
Since the counters were reset
Since the beginning of this hour
Since server startup
Total 5 000 5 000 5 000
(7) Debug 5 000 5 000 5 000
Syslog Generator 1.0.0
Send Syslog Message
Test Server Performance
Sends multiple syslog messages to evaluate server performance.
Syslog server: 192.168.6.68
UDP port: 514
Generate total number of syslogs: 100000
Try to send (syslogs per secod): 1000
Vary message body length from: 100 to: 200
Do not target at a production server, as this may lead to DoS.
Start
Close
Syslog Generator
Sending Syslog Messages...
Sent 12095 syslogs, average speed 998 per second
Cancel


SyslogGenerator
Total number of syslog messages sent: 100000
Average speed: 1000 syslogs per second
Messages are marked: Origin=SyslogGenerator, Tag=Tag-22787
OK


Syslog Watcher Manager
Start Server
Stop Server
Configure
View
Export
Report
FL Files
KB Files
Options
Support
Info
Server
Server Log
Storage
Originators
View: Latest
Refresh
Show 'Debug'
Explore Log Folder...
Level Timestamp Feature Message
Info 2025-02-01 08:37:13+05:30 Server The storage path is blank. The default folder will be used: C:\ProgramData\SyslogStorage
Info 2025-02-01 08:37:14+05:30 Server The storage has been created in: C:\ProgramData\SyslogStorage
Info 2025-02-01 08:43:12+05:30 Server >>> Syslog Watcher Server version 6.5.12, Windows OS 602931718
Info 2025-02-01 08:43:12+05:30 License Free License (3 originators, 5000 msg/h)
Info 2025-02-01 08:43:12+05:30 Receiver (Interface UDP/0.0.0.0/514) Interface has been created. Maximum message length = 10000.
Info 2025-02-01 08:43:12+05:30 Server Server has started sucessfully. Syslog storage: C:\ProgramData\SyslogStorage
Info 2025-02-01 08:43:58+05:30 Server <<< Server has stopped.
Info 2025-02-01 08:44:30+05:30 Server >>> Syslog Watcher Server version 6.5.12, Windows OS 602931718
Info 2025-02-01 08:44:30+05:30 License Free License (3 originators, 5000 msg/h)
Info 2025-02-01 08:44:30+05:30 Receiver (Interface UDP/192.168.6.68/514) Interface has been created. Maximum message length = 10000.
Info 2025-02-01 08:44:30+05:30 Server Server has started sucessfully. Syslog storage: C:\ProgramData\SyslogStorage
Info 2025-02-01 08:44:52+05:30 Storage Syslog originator 192.168.6.68 / staffpc3 has been registered.
Warning 2025-02-01 08:44:57+05:30 License The maximum number of messages per hour (5000) has already been reached. New messages will be discarded until the end of this hou

Practical 10

Install and Run Splunk on Linux

A **Splunk server** in Linux is the instance of Splunk Enterprise or Splunk Free installed and running on a Linux system. Splunk is a powerful platform for **searching, monitoring, and analyzing machine-generated big data** through a web-based interface. It is often used for IT operations, security, application monitoring, and business analytics.

Prerequisites

Before you begin, ensure the following:

- A Linux distribution is installed (e.g., Ubuntu, CentOS, or Red Hat).
- At least **2 GB of RAM** and **2 CPU cores** are recommended for basic Splunk setups.
- You have root or sudo privileges on the system.
- Ensure ports **8000 (web interface)** and **8089 (management)** are open.

Download Splunk

- Go to the Splunk Downloads page. **https://www.splunk.com/**
- Select the **Free Splunk Enterprise** version (or another edition you prefer).
- Choose the correct package for your Linux system:

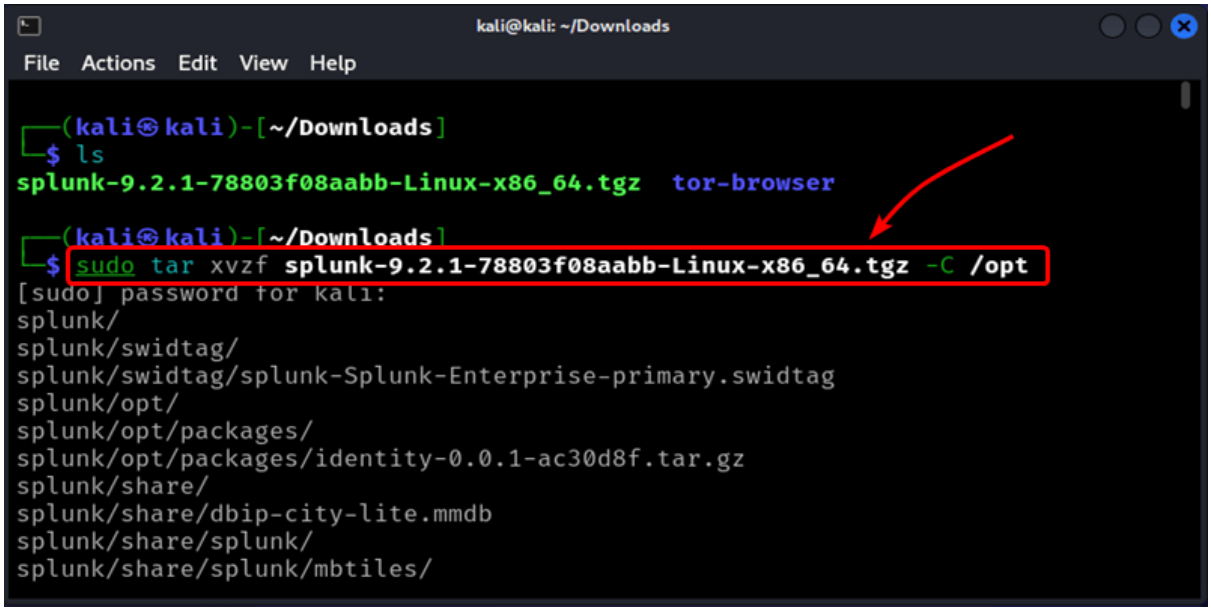
1. .rpm for Red Hat, CentOS, Fedora
2. .deb for Ubuntu, Debian
3. .tgz for manual installations on any Linux distribution.

#wget -O splunk.tgz
"https://download.splunk.com/products/splunk/releases/9.0.3/linux/splunk-9.0.3-dd0128b1f8cd-Linux-x86_64.tgz"

To install Splunk, go to the folder where you downloaded the file. Usually, the file is in the Downloads folder. Open the terminal and change to the Downloads folder. Use the below command.

#cd Downloads

Now we've to extract the file in order to install it to do that use the below command. The '/opt' directory is a standard location on Linux systems for installing optional software packages

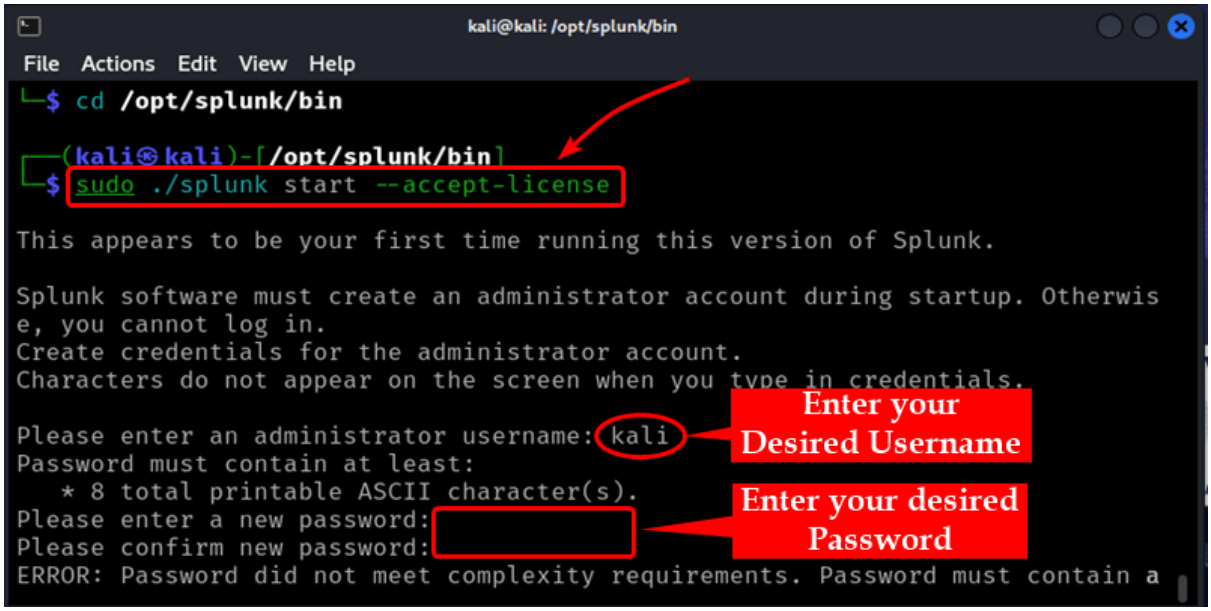


#tar xvzf splunk-9.2.1-78803f08aabb-Linux-x86_64.tgz -C /opt

After extracting the Splunk installation files, we need to change to the specific directory where the program files are located. We can do this by using the following command.

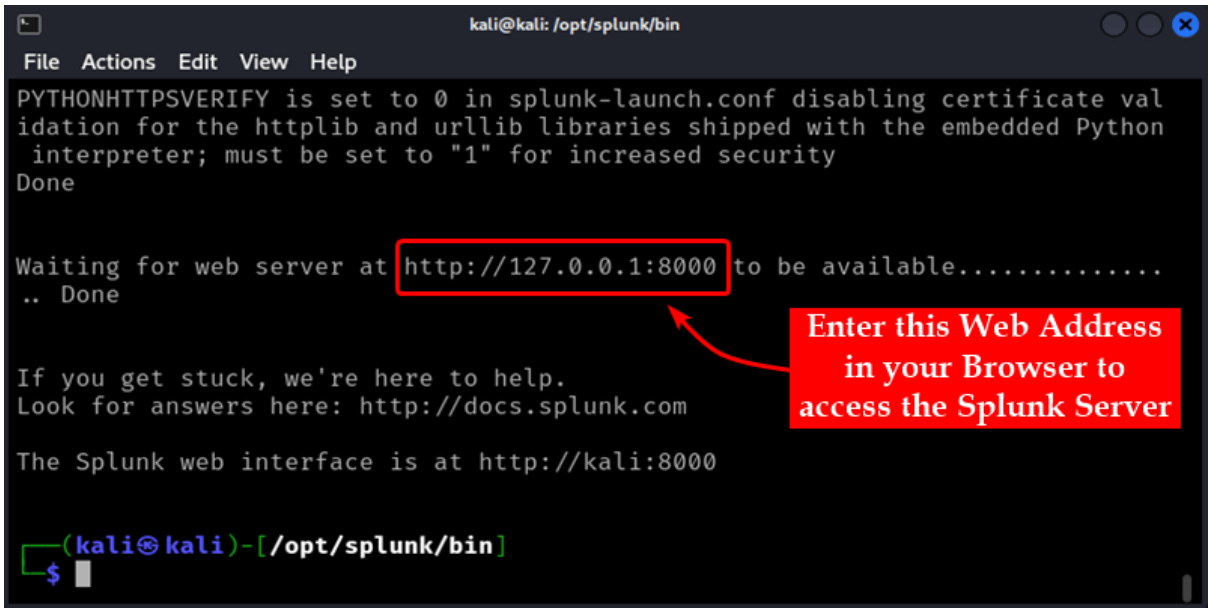
cd /opt/splunk/bin

After changing to the Splunk program directory, we need to accept the license agreement to start the Splunk server. We can do this by running the following command. After running this command, Splunk will prompt you to enter a username and password. You can choose any username and password you prefer. This username and password will be used to log into the Splunk web interface later.




```
# ./splunk start --accept-license
```

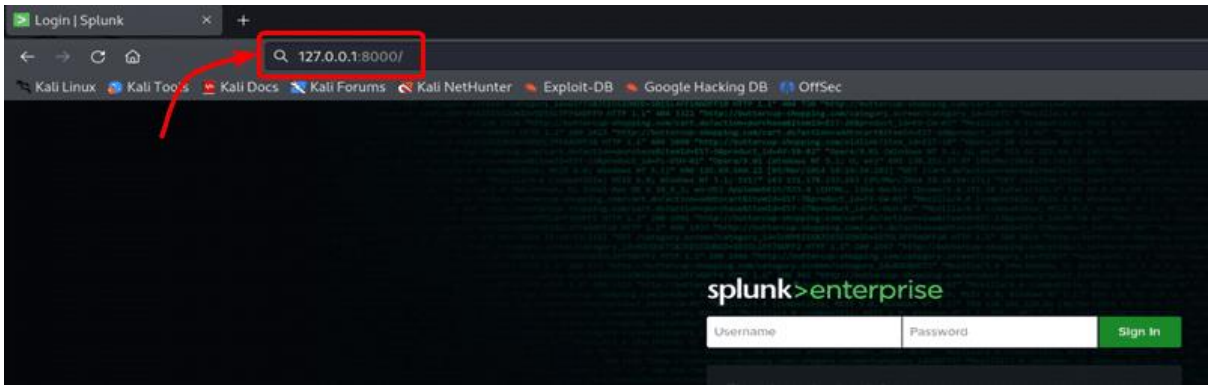
After setting up the username and password, you will see a web address displayed in the terminal output. This is the address you need to access the Splunk server through a web browser.

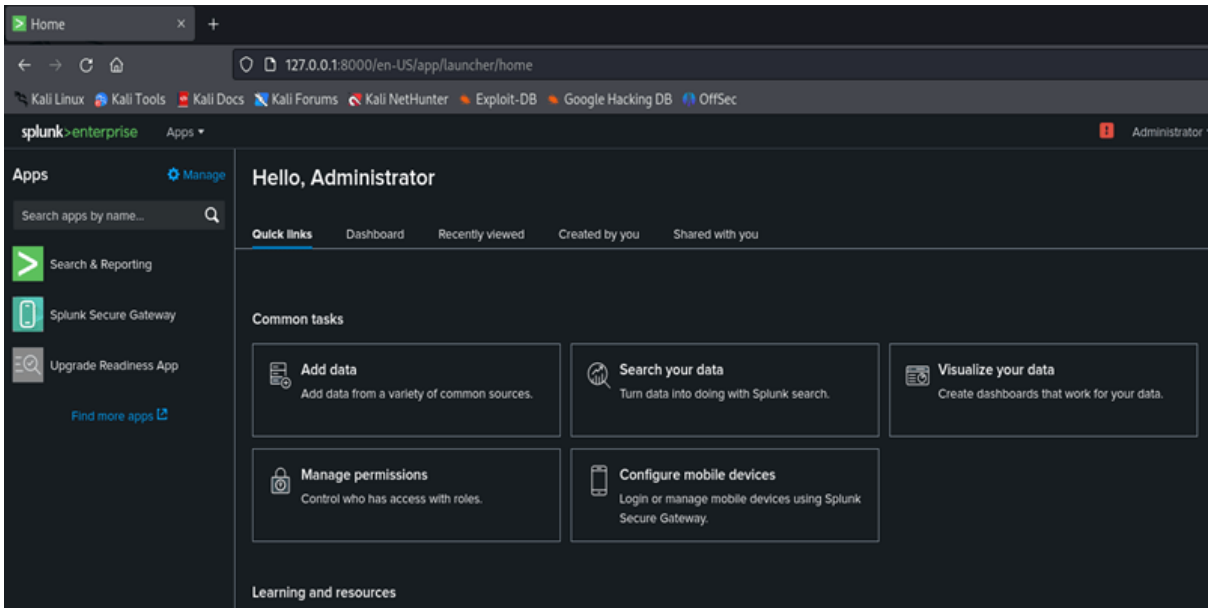


On the Splunk login page, you will see fields to enter your username and password. Enter the same username and password that you set in Step 4 when you ran the command to start the Splunk server.

Enable Splunk as a Service
To ensure Splunk starts automatically on system boot:
/opt/splunk/bin/splunk enable boot-start

Check the Splunk status to ensure it's running:
/opt/splunk/bin/splunk status
Open a browser and go to http://<your-server-ip>:8000





That's it! You have now successfully signed in to the Splunk server using the web interface. After signing in, you will be able to access the Splunk dashboard and start exploring its various features for searching, analyzing, and visualizing your data.

Practical 11

Install and Configure ELK on Linux

An **ELK server** refers to a server configured to run the **ELK Stack**, which is a popular suite of open-source tools for **log management, monitoring, and analytics**. The ELK Stack is composed of three primary tools:

1. **Elasticsearch**: A distributed search and analytics engine.
2. **Logstash**: A data processing pipeline that ingests, transforms, and sends data.
3. **Kibana**: A visualization tool for creating dashboards and graphs based on Elasticsearch data.

When combined, these tools create a robust system for managing and analyzing large volumes of data. It is ideal for organizations looking for a scalable, customizable, and open-source alternative to commercial solutions like Splunk.

Installing and configuring the ELK Stack (Elasticsearch, Logstash, and Kibana) on Linux involves several steps. Here's a comprehensive guide:

1. Prerequisites

Before installing ELK, ensure the following:

- **Linux distribution**: Ubuntu, Debian, CentOS, or RHEL.
- **Java installed**: Elasticsearch and Logstash require Java. Use OpenJDK or Oracle JDK.
- **Sudo or root access**.
- **Ports**:
  - Elasticsearch: 9200 (HTTP API) and 9300 (Node communication).
  - Kibana: 5601.
  - Logstash: Configurable input ports (default is 5044 for Beats).

Java Installation

1. Install Java

Verify Java installation: # java –version

2. Add Elastic's GPG Key and Repository

rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch

3. Add the repository:

```
cat <<EOF | sudo tee /etc/yum.repos.d/elastic.repo
[elastic-8.x]
name=Elastic repository for 8.x packages
baseurl=https://artifacts.elastic.co/packages/8.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
```

type=rpm-md
EOF

Elasticsearch Installation

1. **Install Elasticsearch:**
   # yum install elasticsearch –y

2. **Configure Elasticsearch:**
Edit /etc/elasticsearch/elasticsearch.yml

network.host: 0.0.0.0

discovery.type: single-node

3. **Start and enable Elasticsearch:**
systemctl enable elasticsearch

systemctl start elasticsearch

4. **Verify Elasticsearch:**
curl -X GET "http://localhost:9200"

Logstash Installation

1. **Install Logstash**
yum install logstash –y

2. **Configure Logstash:**
Create a configuration file, e.g., /etc/logstash/conf.d/logstash.conf

```
input {
  beats {
    port => 5044
  }
}
output {
  elasticsearch {
    hosts => ["http://localhost:9200"]
    index => "logstash-%{+YYYY.MM.dd}"
  }
}
```

3. Start and enable Logstash

systemctl enable logstash

systemctl start logstash

Kibana Installation

1. Install Kibana

sudo yum install kibana -y # For CentOS/RHEL

2. Configure Kibana:

Edit /etc/kibana/kibana.yml

server.host: "0.0.0.0"

elasticsearch.hosts: ["http://localhost:9200"]

3. Start and enable Kibana

sudo systemctl enable kibana

sudo systemctl start kibana

4. Access Kibana:

Open a browser and go to http://<server-ip>:5601

Configure a Data Pipeline

1. Install a Beats agent

(e.g., Filebeat) on a client/server to send data to Logstash.

sudo apt install filebeat -y # Ubuntu/Debian

sudo yum install filebeat -y # CentOS/RHEL

2. Configure Filebeat to send data to Logstash:

Edit /etc/filebeat/filebeat.yml

output.logstash:

hosts: ["<ELK-server-IP>:5044"]

3. Start Filebeat:

sudo systemctl enable filebeat

sudo systemctl start filebeat

4. Confirm data flow

Logs sent by Filebeat are processed by Logstash, stored in Elasticsearch, and visualized in Kibana.

5. Test the ELK Stack

- In Kibana, navigate to **Discover** and check if logs from Logstash/Filebeat appear.
- Use **Dashboards** to visualize data.

Practical 12

Install and Configure GrayLog on Linux

A **Graylog server** is a centralized log management system designed to collect, store, and analyze machine data, including logs from applications, servers, and network devices. It provides powerful tools for monitoring, searching, and visualizing log data, making it a popular choice for IT operations, security, and compliance use cases.

Graylog vs. ELK Stack

Feature	Graylog	ELK Stack
Ease of Use	Easier to set up and manage.	More complex configuration.
Backend	Requires Elasticsearch and MongoDB.	Relies solely on Elasticsearch.
Alerting	Built-in alerting.	Requires plugins (e.g., Watcher).
Scalability	Suitable for mid-sized deployments.	Better suited for large-scale.
Community	Smaller, but active.	Large and mature community.

Installing and configuring **Graylog** on Linux involves several steps, as it requires setting up dependencies such as MongoDB and Elasticsearch. Here's a step-by-step guide:

1. Prerequisites

- A Linux system (Ubuntu or CentOS/RHEL is recommended).
- sudo or root privileges.
- A stable internet connection.
- Ensure required ports are open:
  - **9000**: Graylog web interface.
  - **9200**: Elasticsearch.
  - **27017**: MongoDB.
  - **1514**: Default for syslog (optional).

Install Java

 # Install Java on CentOS/RHEL

sudo yum install java-11-openjdk-devel –y

Verify Java installation:

java –version

Install MongoDB

Graylog uses MongoDB to store configuration and metadata.

On CentOS/RHEL:

Add the MongoDB repository:

```
cat <<EOF | sudo tee /etc/yum.repos.d/mongodb-org-6.0.repo
[mongodb-org-6.0]
name=MongoDB Repository
baseurl=https://repo.mongodb.org/yum/redhat/\$releasever/mongodb-org/6.0/x86_64/
gpgcheck=1
enabled=1
gpgkey=https://www.mongodb.org/static/pgp/server-6.0.asc
EOF
```

Install MongoDB:

```
sudo yum install mongodb-org –y
```

Start and enable MongoDB:

```
sudo systemctl start mongod
sudo systemctl enable mongod
```

Install Elasticsearch

Graylog requires Elasticsearch for storing and indexing log data.

Add the Elasticsearch repository:

```
cat <<EOF | sudo tee /etc/yum.repos.d/elastic.repo
[elastic-8.x]
name=Elastic repository for 8.x packages
baseurl=https://artifacts.elastic.co/packages/8.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
```

enabled=1

autorefresh=1

type=rpm-md

EOF

sudo yum install elasticsearch –y

Configure Elasticsearch: Edit /etc/elasticsearch/elasticsearch.yml

cluster.name: graylog

network.host: 127.0.0.1

Start and enable Elasticsearch:

sudo systemctl start elasticsearch

sudo systemctl enable elasticsearch

Install Graylog

Add the Graylog repository:

sudo rpm -Uvh https://packages.graylog2.org/repo/packages/graylog-5.x-repository_latest.rpm

sudo yum install graylog-server –y

Configure Graylog: Edit /etc/graylog/server/server.conf

Set the **password secret**

password_secret=$(openssl rand -base64 32)

echo "password_secret = $password_secret" | sudo tee -a /etc/graylog/server/server.conf

Hash the admin password

admin_password=$(echo -n 'YourPassword' | sha256sum | awk '{print $1}')

echo "root_password_sha2 = $admin_password" | sudo tee -a /etc/graylog/server/server.conf

Set the **rest_listen_uri**

http_bind_address = 0.0.0.0:9000

Start and enable Graylog

sudo systemctl start graylog-server

sudo systemctl enable graylog-server

Access Graylog Web Interface

Open a browser and navigate to

http://<server-ip>:9000

Log in with:

- Username: admin
- Password: The password you set in root_password_sha2

Configure Log Inputs

1. In the Graylog web interface, go to **System > Inputs**.
2. Select an input type (e.g., Syslog UDP) and configure it.
3. Start the input to collect logs.

Secure the Graylog Stack

- Enable TLS for secure communication between components.
- Use firewalls to restrict access to ports.
- Regularly update Graylog and its dependencies.