

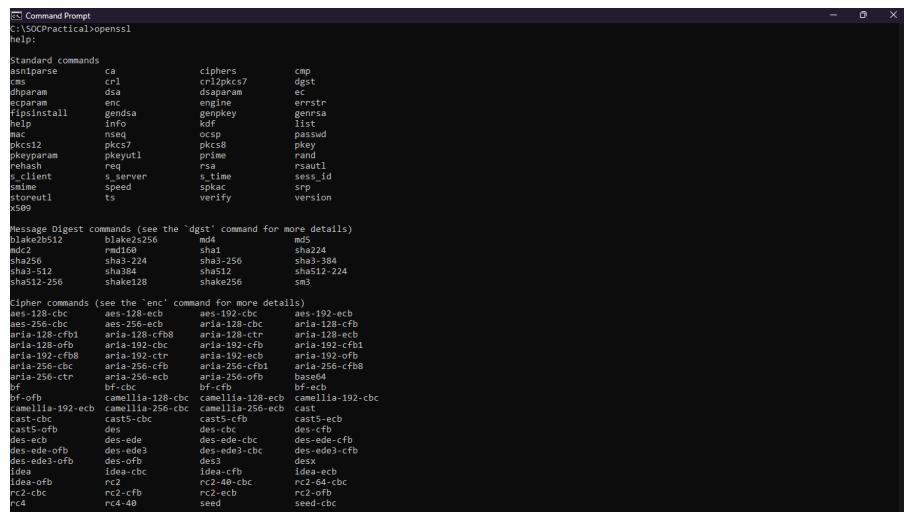
Index

List of Practical		Date	Signature
1.	Encrypting and Decrypting Data Using OpenSSL	20/2/23	
2.	Demonstrate the use of Snort and Firewall Rules.	1/3/23	
3.	Demonstrate Extract an Executable from a PCAP.	9/3/23	
4.	Demonstrate Analysis of DNS Traffic	18/3/23	
5.	Create your own syslog server	21/3/23	
6.	Install and Run Splunk on Linux	3/4/23	
7.	Install and Configure ELK on Linux	10/4/23	
8.	Install and Configure GrayLog on Linux	17/4/23	

Table of Contents

Practical 1: Encrypting and Decrypting Data Using OpenSSL.....	3
Practical 2: Demonstrate the use of Snort and Firewall Rules.....	4
Practical 3: Demonstrate Extract an Executable from a PCAP	11
Practical 4: Demonstrate Analysis of DNS Traffic.....	14
Practical 5: Create your own syslog Server.....	17
Practical 6: Install and Run Splunk on Linux.....	20
Practical 7: Install and configure ELK on linux.....	30
Practical 8: Install and Configure GrayLog on Linux.....	44

PRACTICAL 1: ENCRYPTING AND DECRYPTING DATA USING OPENSSL



```
C:\ Command Prompt
C:\SOCPractical>openssl
help:

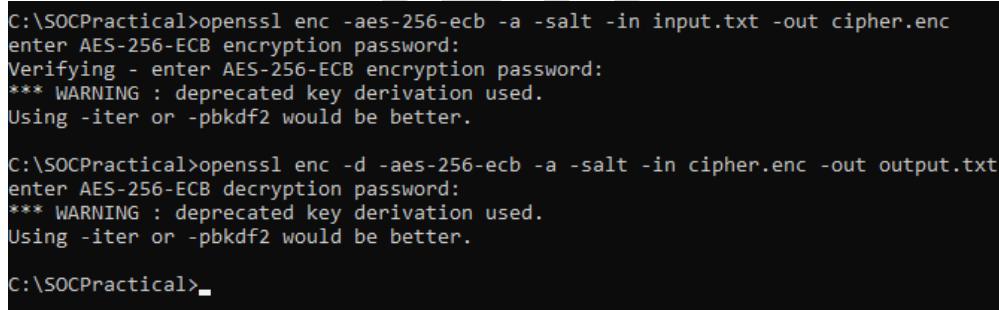
Standard commands
asn1parse      ca          ciphers       cmp
cms            crl        crl2pkcs7   dgst
dparam         dsa          dssparam     ec
ecparam        enc          engine       erestr
geninstall    gendsa     genkey      genrsa
help           info        kdf          list
mac            nseq        ocsp         pkey
pkcs12        pkcs7      pkcs8       passwd
program       pkcs12utl   pkcs12utl  rand
rehash        req          rsa          rsautl
s_client       s_server    s_time      sess_id
simech        speed       spkac      srp
smeventl      ts          verify      version
x509

Message Digest commands (see the 'dgst' command for more details)
blake2b512    blake2s256   md4          md5
md2           md5160     sha1        sha224
sha256        sha3-224   sha3-256   sha3-384
sha512        sha384     sha512    sha512-224
sha512-256   sha512-256  shaKE256   sm3

Cipher commands (see the 'enc' command for more details)
aes-128-cbc   aes-128-ecb  aes-192-cfb  aes-192-ecb
aes-128-cfb   aes-128-ecb  aes-192-cbc  aes-192-cfb
aria-128-cfb1 aria-128-cfb8  aria-128-ctr  aria-128-cbc
aria-128-ofb   aria-128-cbc  aria-192-cfb  aria-192-cfb1
aria-192-cfb8  aria-192-ctr  aria-192-cbc  aria-192-ofb
aria-192-ofb8  aria-192-cfb  aria-256-cfb  aria-256-cfb
aria-256-cfb1 aria-256-cbc  aria-256-cfb  aria-256-cfb
bf             bf-cbc      bf-cfb      bf-cfb
bf-cfb        camellia-128-cbc  camellia-128-ecb  camellia-192-cbc
camellia-192-cbc camellia-192-ecb  camellia-192-cfb  camellia-192-cfb
cast-cbc       casts-cbc   casts-cfb   casts-ecb
casts-ofb      casts-cbc   des-cbc     des-cfb
des-ecb        des-ede    des-ede3   des-ede-cfb
des-ede3      des-ede3   des-ede3-cbc  des-ede3-cfb
des-ede3-ofb  des-ofb    des3       des3
idea           idea-cbc   idea-cfb   idea-ecb
idea-ofb       rc4-40     rc4-40-cbc  rc4-64-cbc
rc2-cbc        rc2-cfb   rc2-40     rc2-ofb
rc4-40         seed       seed-cbc   seed-cfb
```

FIGURE 1 - OPENSSL COMMANDS

Create a text file named input.txt and add some text or data in it. Encrypt the file with file command. After encrypting the file, a .enc file will be generated. And decrypt it into another file and name it output.txt.



```
C:\SOCPractical>openssl enc -aes-256-ecb -a -salt -in input.txt -out cipher.enc
enter AES-256-ECB encryption password:
Verifying - enter AES-256-ECB encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

C:\SOCPractical>openssl enc -d -aes-256-ecb -a -salt -in cipher.enc -out output.txt
enter AES-256-ECB decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

C:\SOCPractical>
```

FIGURE 2 - ENCRYPTION AND DECRYPTION USING AES 256

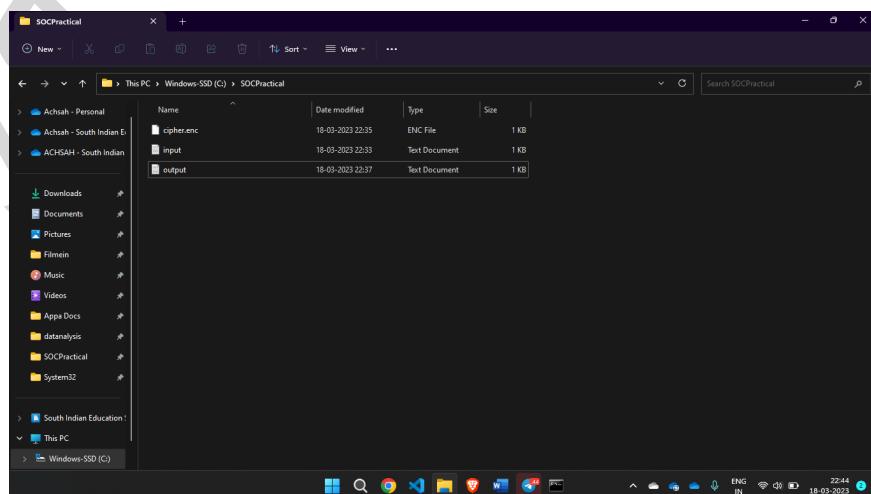


FIGURE 3 - ALL FILES GENERATED BY OPENSSL

PRACTICAL 2: DEMONSTRATE THE USE OF SNORT AND FIREWALL RULES

Step-1-> Download snort application from <https://snort.org/downloads> from binaries section [Snort 2.9.20 Installer.x64.exe](#)

Step-2-> Download rules tar file from the registered section [snortrules-snapshot-29200.tar.gz](#).

Step-3-> Install snort.exe file

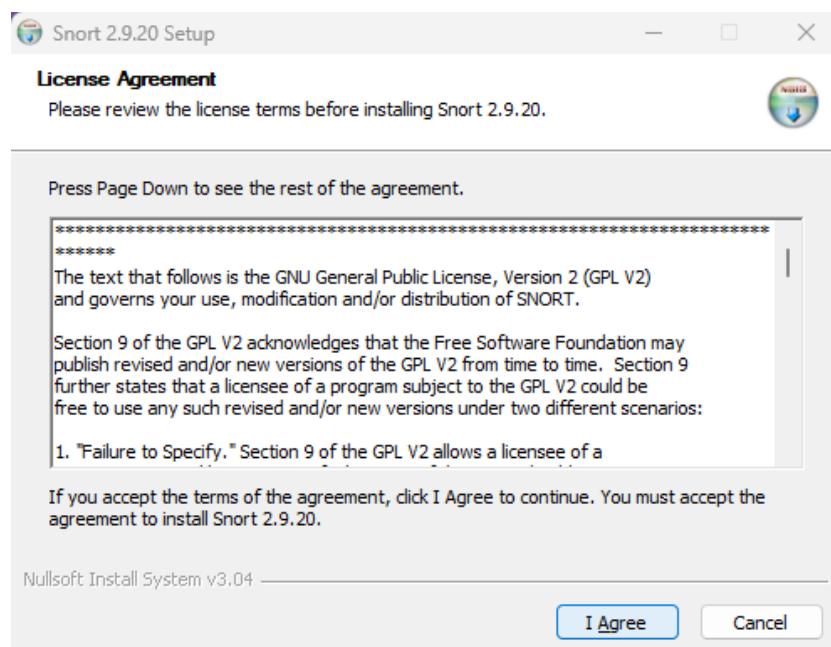


FIGURE 4 - SNORT INSTALLATION

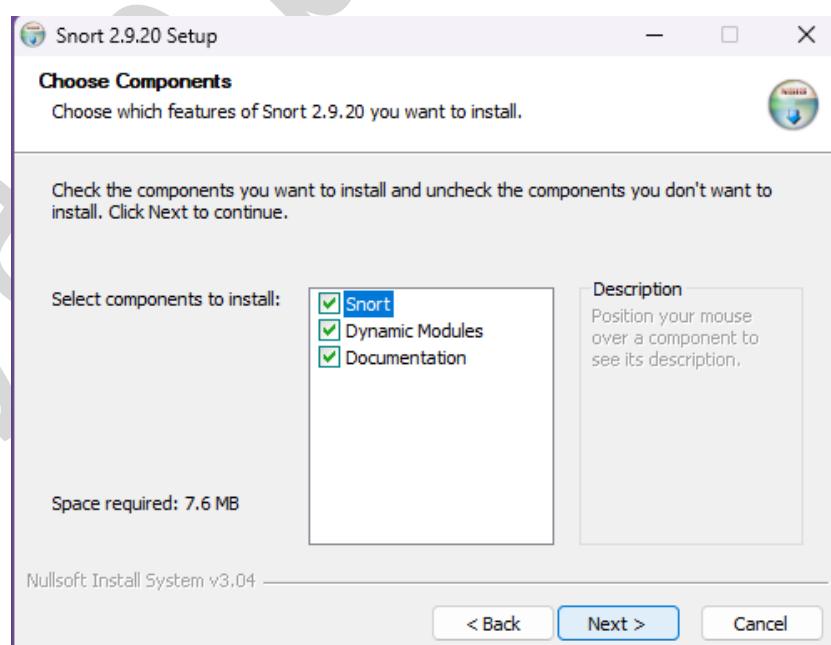


FIGURE 5 - CHOOSE ALL OPTIONS

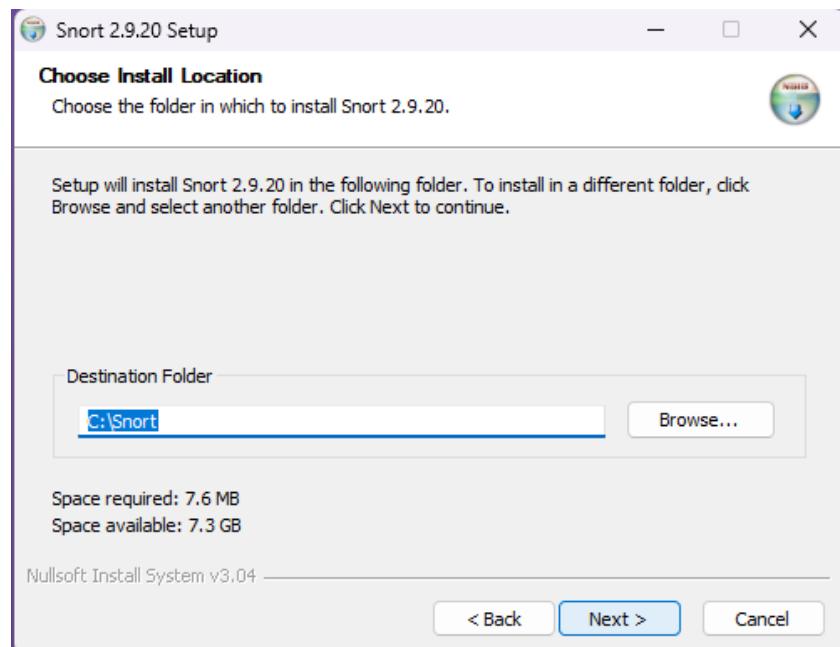


FIGURE 6 - DESTINATION FOLDER

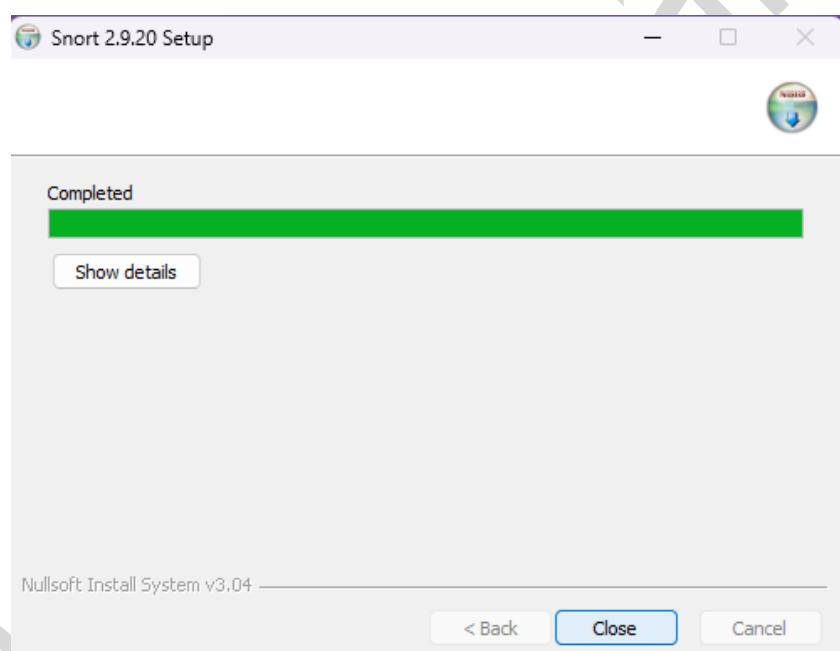


FIGURE 7 - INSTALLATION SUCCESSFUL

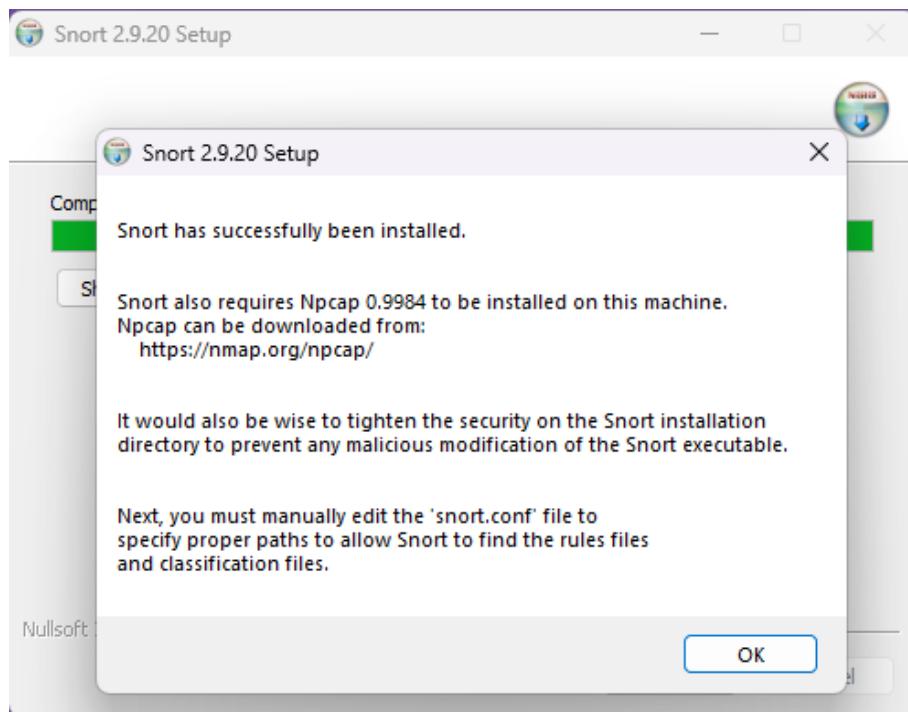


FIGURE 8 - NPCAP REQUIREMENT MESSAGE

Step-4-> Unzip and extract the .tar file

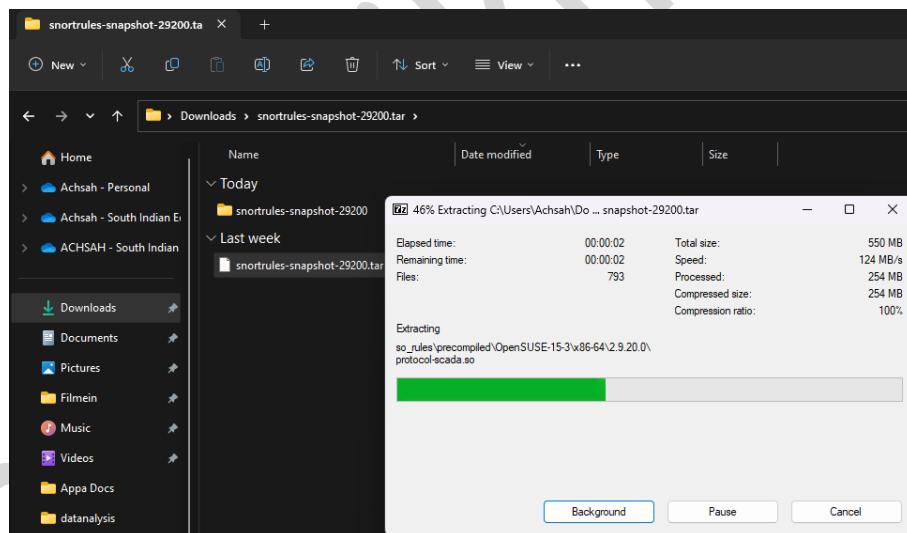


FIGURE 9 - EXTRACTING RULES ARCHIVE

Step-5-> Copy rules and preproc_rules folder from the extracted folder and paste it to the Snort folder in C:\Snort thus replacing the existing folder in the same.

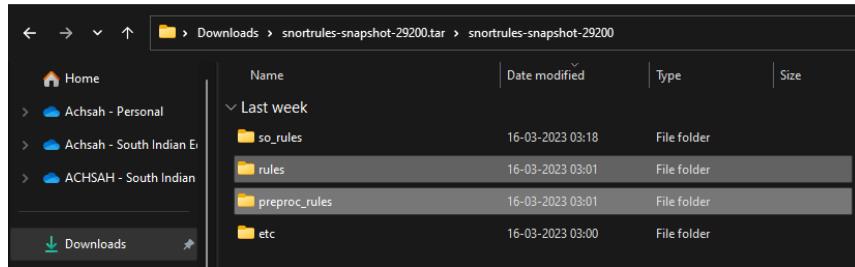


FIGURE 10 - UNZIPPED ARCHIVE

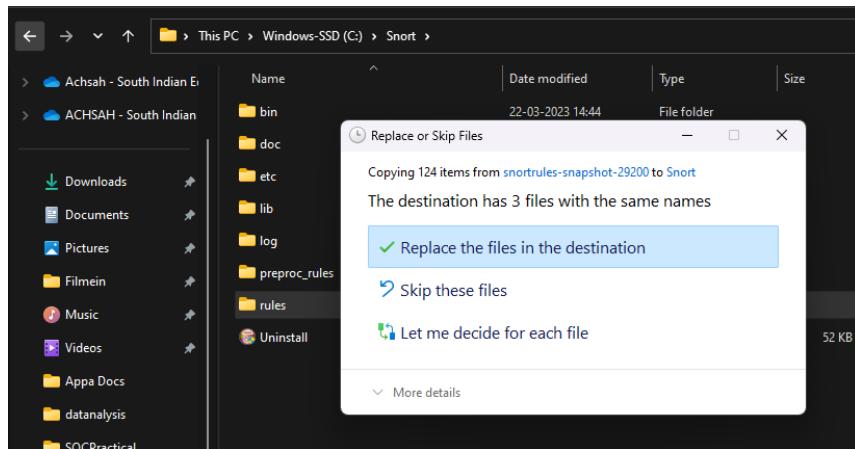


FIGURE 11 - REPLACING PREVIOUS FILES

Step-6-> Download and install notepad++ from <https://notepad-plus-plus.org/downloads/> and open the snort.conf file from C:\Snort\etc in the same.

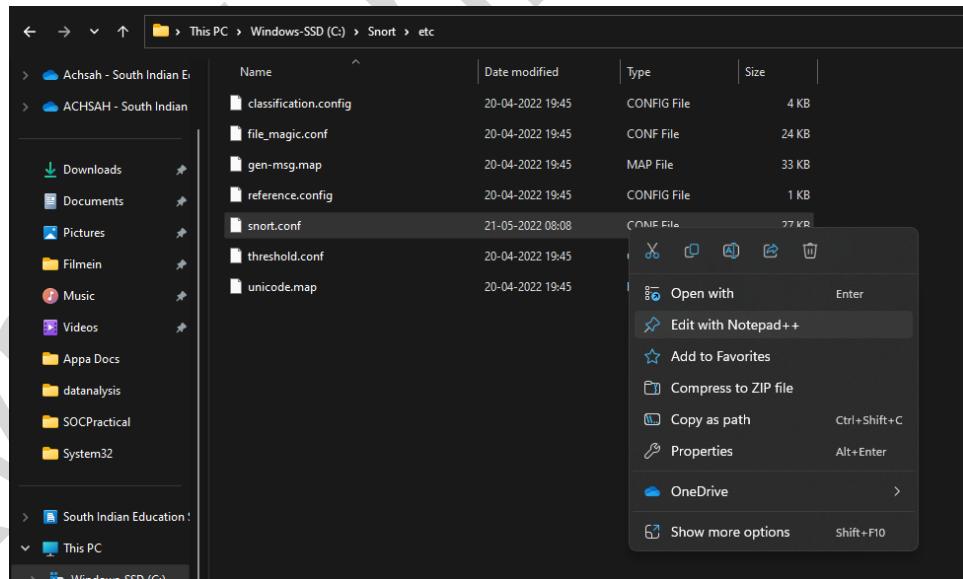


FIGURE 12 - EDIT SNORT.CONF WITH NOTE PAD++

Step-7-> Search for blacklist from the snort folder in C:\Snort, open in notepad++, do save as and name it to whitelist.rules now you will have files with name blacklist.rules and whitelist.rules in the folder.

Step-8-> Edit the config file in the following lines

```
44      # Setup the network addresses you are interested in
45      ipvar HOME_NET 192.168.0.0/24
46
47      # Set up the external network address
48      ipvar EXTERNAL_NET !$HOME_NET
49
104     var RULE_PATH C:\Snort\rules
105     # var SO_RULE_PATH ../so_rules
106     var PREPROC_RULE_PATH C:\Snort\preproc_rules
107
112     # Set the absolute path appropriately
113     var WHITE_LIST_PATH C:\Snort\rules
114     var BLACK_LIST_PATH C:\Snort\rules
115
185     #
186     config logdir: C:\Snort\log
187
188
246     # path to dynamic preprocessor libraries
247     dynamicpreprocessor directory C:\Snort\lib\snort_dynamicpreprocessor
248
249     # path to base preprocessor engine
250     dynamicengine C:\Snort\lib\snort_dynamicengine\sf_engine.dll
251
252     # path to dynamic rules libraries
253     # dynamicdetection directory /usr/local/lib/snort_dynamicrules
254
264     # Does nothing in IDS mode
265     # preprocessor normalize_ip4
266     # preprocessor normalize_tcp: ips ecn stream
267     # preprocessor normalize_icmp4
268     # preprocessor normalize_ip6
269     # preprocessor normalize_icmp6
270
```

Step-9-> From line no. 546 to 651 replace forward slash to backslash of the path given.

```
545 # site specific rules
546 include $RULE_PATH\local.rules
547 ...
548 include $RULE_PATH\app-detect.rules
549 include $RULE_PATH\attack-responses.rules
550 include $RULE_PATH\backdoor.rules
551 include $RULE_PATH\bad-traffic.rules
552 include $RULE_PATH\blacklist.rules
553 include $RULE_PATH\botnet-cnc.rules
554 include $RULE_PATH\browser-chrome.rules
555 include $RULE_PATH\browser-firefox.rules
556 include $RULE_PATH\browser-ie.rules
557 include $RULE_PATH\browser-other.rules
558 include $RULE_PATH\browser-plugins.rules
559 include $RULE_PATH\browser-webkit.rules
560 include $RULE_PATH\chat.rules
561 include $RULE_PATH\content-replace.rules
562 include $RULE_PATH\ddos.rules
563 include $RULE_PATH\dns.rules
564 include $RULE_PATH\dos.rules
565 include $RULE_PATH\experimental.rules
566 include $RULE_PATH\exploit-kit.rules
567 include $RULE_PATH\exploit.rules
568 include $RULE_PATH\file-executable.rules
569 include $RULE_PATH\file-flash.rules
570 include $RULE_PATH\file-identify.rules
571 include $RULE_PATH\file-image.rules
572 include $RULE_PATH\file-multimedia.rules
573 include $RULE_PATH\file-office.rules
574 include $RULE PATH\file-other.rules

657
658 # decoder and preprocessor event rules
659 include $PREPROC_RULE_PATH/preprocessor.rules
660 include $PREPROC_RULE_PATH/decoder.rules
661 include $PREPROC_RULE_PATH/sensitive-data.rules
662
```

Step-10-> Open local.rules from C:\Snort\rules in notepad++ and add the following lines of code in it.

```
21 alert icmp any any -> any any (msg:"Testing ICMP"; sid:1000001;)
22 alert udp any any -> any any (msg:"Testing UDP"; sid:1000002;)
23
24 alert tcp any any -> any any (msg:"Testing TCP"; sid:1000003;)
```

Step-11-> Open command line in the path C:\Snort\bin and run the command snort -W

```
C:\Snort\bin>snort -W
--> Snort! <-
o...)- Version 2.9.20-WIN64 GRE (Build 82)
By Martin Roesch and The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014-2022 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using PCRE version: 8.10 2010-06-25
Using ZLIB version: 1.2.11

Index Physical Address IP Address Device Name Description
1 00:00:00:00:00:00 disabled \Device\NPF_{70B1023C-CB1B-4D42-AACE-6053BA239C62} WAN Miniport (Network Monitor)
2 00:00:00:00:00:00 disabled \Device\NPF_{7041BC9-C459-4414-9246-4557CAF3787} WAN Miniport (IPv6)
3 00:00:00:00:00:00 disabled \Device\NPF_{A02C2DF4-GA26-47AB-841F-EFA013BD4B4D} WAN Miniport (IP)
4 E8:D0:FC:E1:EF:58 169.254.39.155 \Device\NPF_{5F24DE34-CF9E-468C-91A3-7FCE444BE6CF} Bluetooth Device (Personal Area Network)
5 E8:D0:FC:E1:EF:4F 192.168.0.102 \Device\NPF_{8003E5E0-90F0-45E9-B649-C54C390F5069} Qualcomm Atheros QCA9377 Wireless Network Adapter
6 FA:D0:FC:E1:EF:4F 169.254.184.44 \Device\NPF_{0957CDB5-AC63-402C-B689-73EB8461E09B} Microsoft Wi-Fi Direct Virtual Adapter #4
7 EA:D0:FC:E1:EF:4F 169.254.7.234 \Device\NPF_{FFD03A80-0792-4013-A9AF-599631CBFBAA} Microsoft Wi-Fi Direct Virtual Adapter #3
8 00:00:00:00:00:00 0000:0000:0000:0000:0000:0000 \Device\NPF_Loopback Adapter for loopback traffic capture

C:\Snort\bin>
```

FIGURE 13 - WIRELESS ADAPTERS

Step-12-> Run the command: snort -i 5 -c C:\Snort\etc\snort.conf -A console

```
C:\Windows\System32\cmd.exe
C:\Snort\bin>snort -i 5 -c C:\Snort\etc\snort.conf -A console
Running in IDS mode

--> Initializing Snort ---
Initializing Output Plugins!
Initializing Processors!
Initializing Plugins!
Parsing Rules file "C:\Snort\etc\snort.conf"
PortVar 'HTTP_PORTS' defined : [ 80:81 311 383 591 593 901 1220 1414 1741 1830 2301 2381 2899 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 7777 7779 800
8008 8014 8028 8088 8088 8098 8118 8123 8180:8181 8243 8280 8300 8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 34443:34444 41880 50002 55555 ]
PortVar 'TCP_PORTS' defined : [ 1024:65535 ]
PortVar 'SSH_PORTS' defined : [ 22 ]
PortVar 'FTP_PORTS' defined : [ 21 2100 3535 ]
PortVar 'SIP_PORTS' defined : [ 5060:5061 5600 ]
PortVar 'FILE_DATA_PORTS' defined : [ 80:81 110 143 211 383 591 593 901 1220 1414 1741 1830 2301 2381 2899 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510
7777 7779 8000 8008 8014 8028 8088 8085 8098 8118 8123 8180:8181 8243 8280 8300 8800 8888 8899 9000 9060 9080 9090:9091 9443 9999 11371 34443:34444 41880 50002 555
55 ]
PortVar 'GTP_PORTS' defined : [ 2123 2152 3386 ]
Detection:
  Sensors enabled = Full-D
  Split Any/Any group = enabled
  Search-Method-Optimizations = enabled
  Maximum pattern length = 20
Tagged Packet Limit: 256
Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicengine\sf_engine.dll... done
Loading all dynamic preprocessor libs from C:\Snort\lib\snort\dynamicpreprocessor...
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_dce.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_dnp3.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_dns.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_ftpd.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_imap.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_modbus.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_pop.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_rsh.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_sip.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_smtp.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_ssh.dll... done
  Loading dynamic preprocessor library C:\Snort\lib\snort\dynamicpreprocessor\sf_ssl.dll... done
  Filling known full dynamic preprocessor libs from C:\Snort\lib\snort\dynamicpreprocessor
Log directory = C:\Snort\log
Frag3 global config:
Max frags: 65536
Fragment memory cap: 4194304 bytes
```

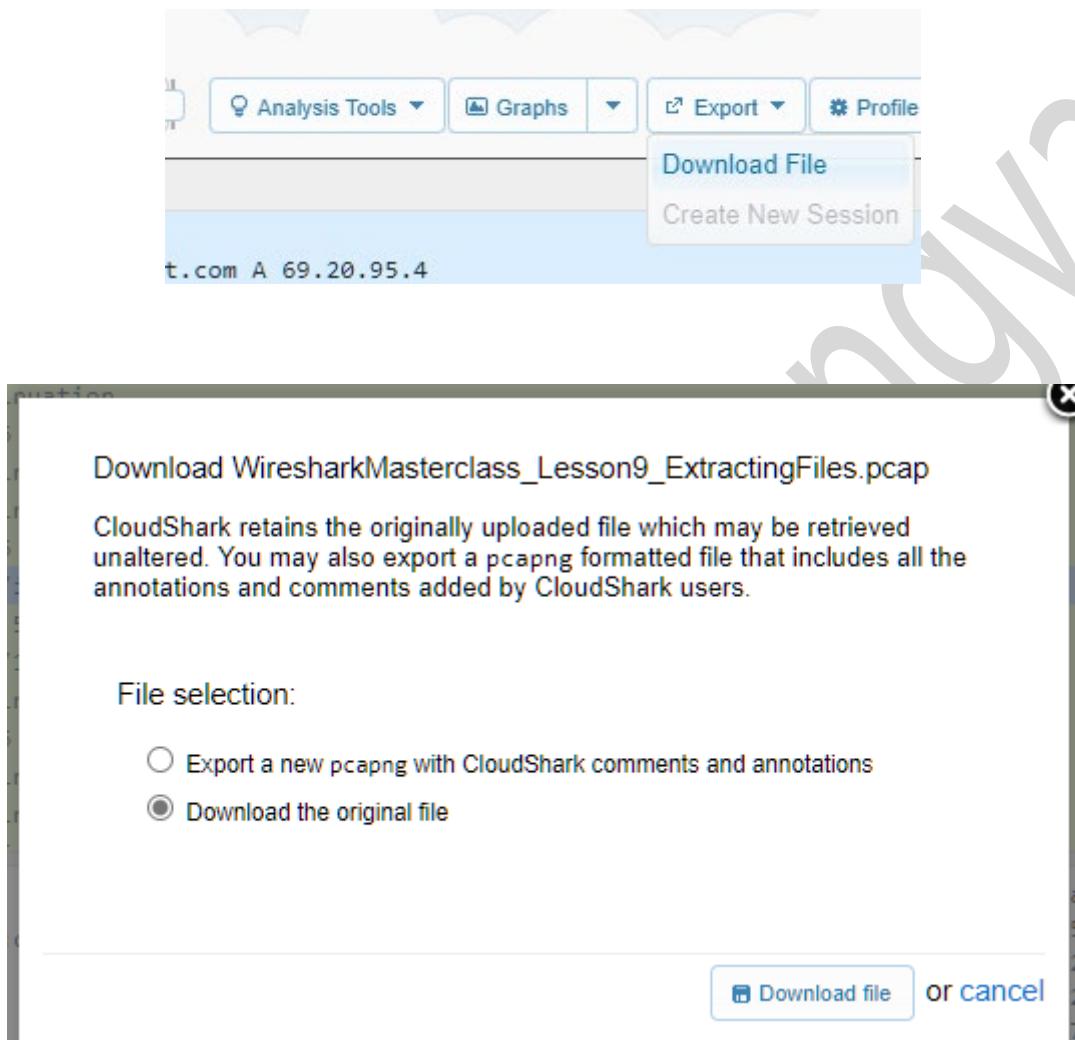
```
03/22-16:38:50.119481 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.121941 [**] [1:1000003:0] Testing TCP [**] [Priority: 0] {TCP} 192.168.0.102:36713 -> 35.223.238.178:443
03/22-16:38:50.136726 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:63398 -> 35.207.209.40:50009
```

```
C:\Windows\System32\cmd.exe
03/22-16:38:50.774351 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774437 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774516 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774564 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774621 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774679 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774729 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774779 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774848 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.774930 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775018 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775065 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775103 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775141 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775179 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775208 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775246 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775284 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775322 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775360 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775398 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 35.267.209.40:50009
03/22-16:38:50.775436 [**] [1:1000001:0] Testing ICMP [*][*] [Priority: 0] {IPV6-ICMP} fe80:0:0:0:40:48ff:fed:8930 -> ff02:0000:0000:0000:0000:0000:0000:0000
1
03/22-16:38:50.775468 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775506 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775544 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775582 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775620 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775658 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775696 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775734 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775772 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775810 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775848 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775886 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775924 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.775962 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776000 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776038 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776076 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776114 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776152 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776190 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776228 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776266 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776304 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776342 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776380 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776418 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776456 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776494 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776532 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776570 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776608 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776646 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776684 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776722 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776760 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776798 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776836 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776874 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776912 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776950 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.776988 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777026 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777064 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777102 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777140 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777178 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777216 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777254 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777292 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777330 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777368 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777406 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777444 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777482 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777520 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777558 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777596 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777634 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777672 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777710 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777748 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777786 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777824 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777862 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777900 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777938 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.777976 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.778014 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.239.4:50007
03/22-16:38:50.778052 [**] [1:1000002:0] Testing UDP [**] [Priority: 0] {UDP} 192.168.0.102:56570 -> 66.22.2
```

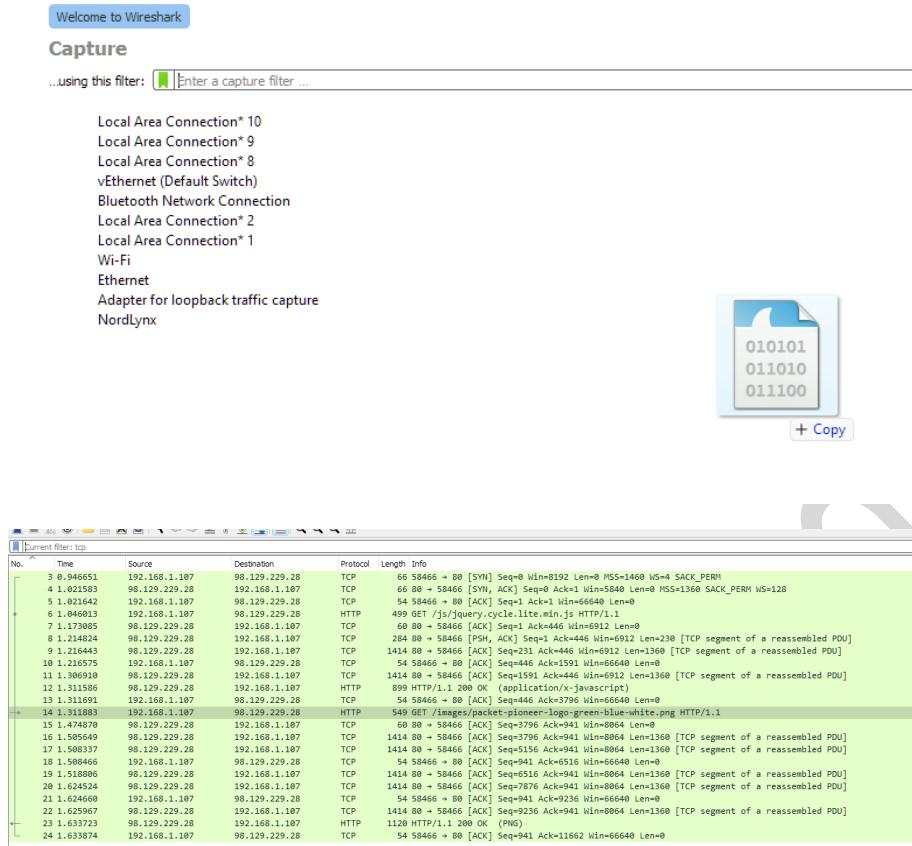
PRACTICAL 3: DEMONSTRATE EXTRACT AN EXECUTABLE FROM A PCAP

Step-1-> Download packet capture file from :

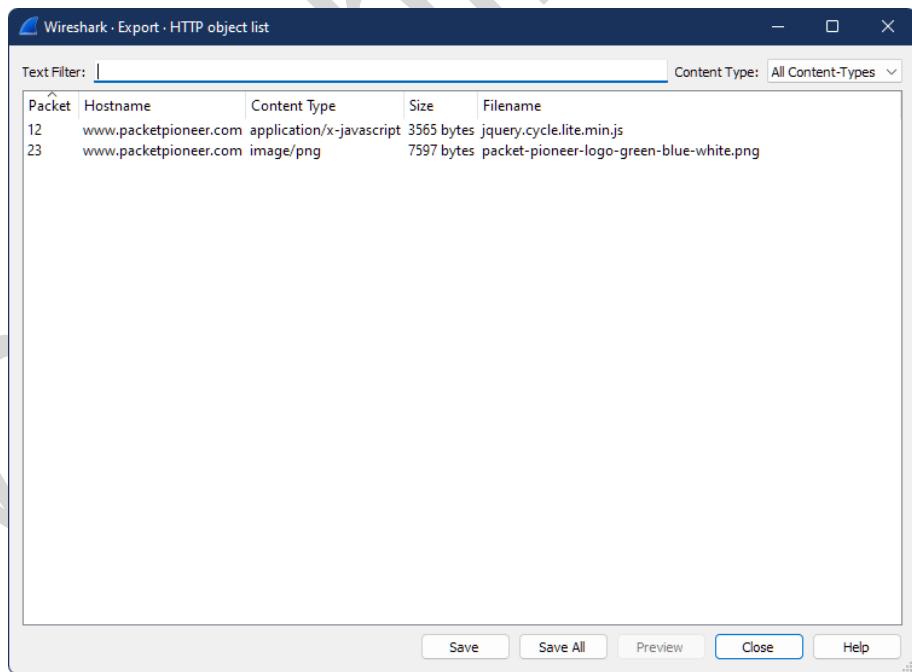
<https://www.cloudshark.org/captures/a9472fbe700a>



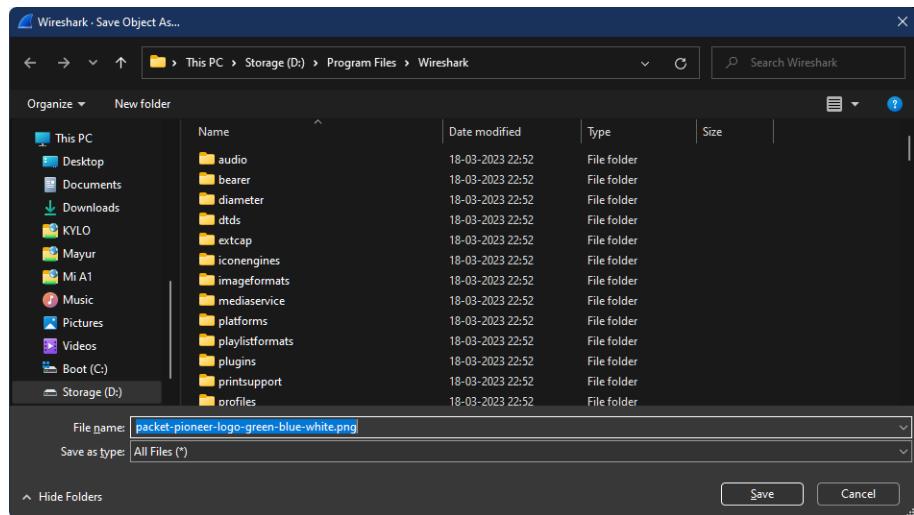
Step-2-> Open the packet capture file in Wireshark by dragging and dropping the capture file into the Wireshark interface.



Step-3-> Click on File->Export Objects-> HTTP



Step-4-> Select image/png and click on Save. Select Save Location.



PRACTICAL 4: DEMONSTRATE ANALYSIS OF DNS TRAFFIC

Step-1-> Open Wireshark, select Wi-Fi and click on Capture to capture the DNS Traffic

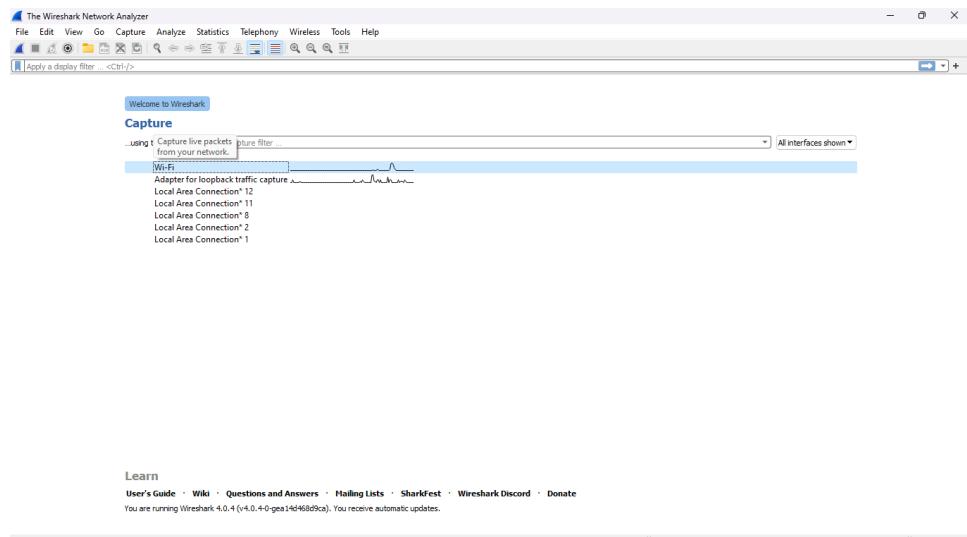


FIGURE 15 - ALL NETWORK ADAPTERS

Step-2-> Click start

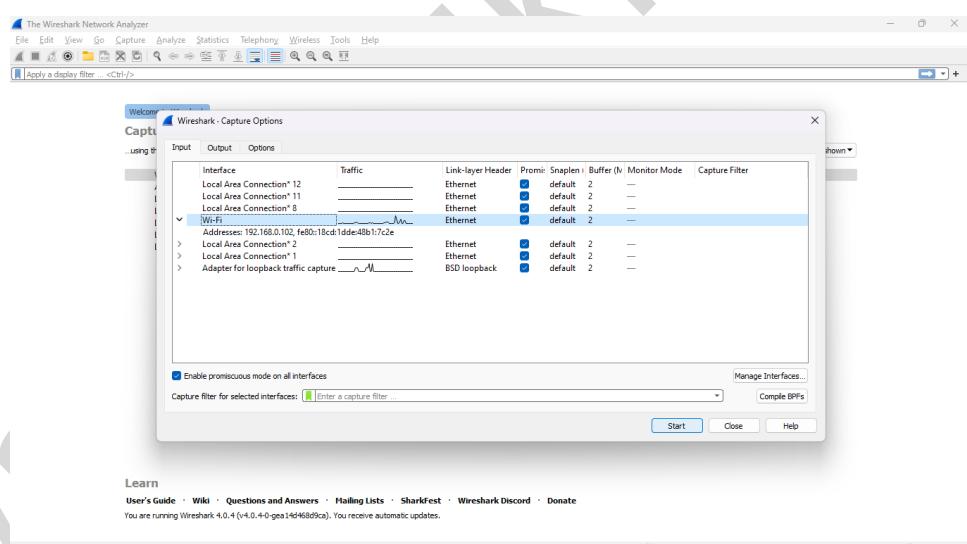


FIGURE 16 - SELECTING ADAPTER TO CAPTURE PACKETS

Step-3-> Following window will open

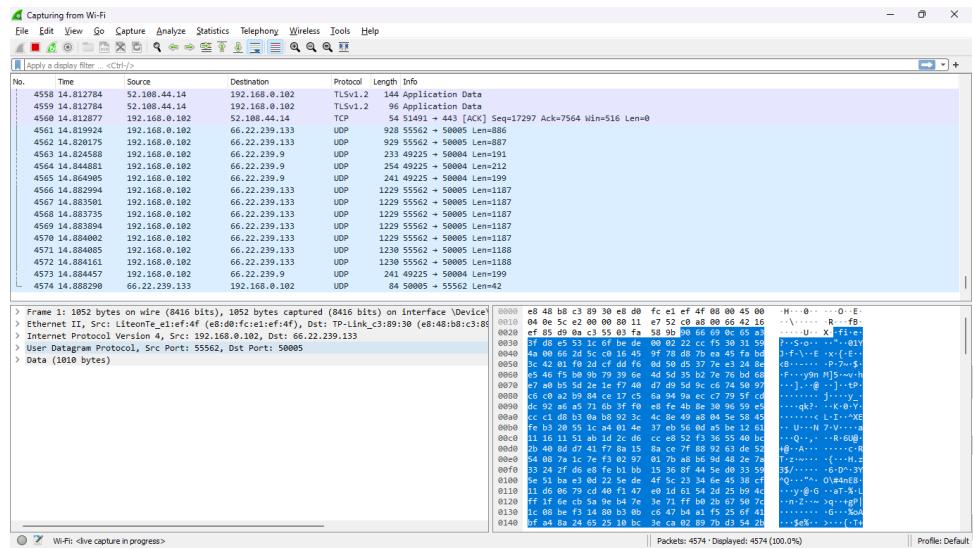


FIGURE 17 - ALL PACKETS CAPTURED BY WIRESHARK

Step-4-> Search for dns in the search bar

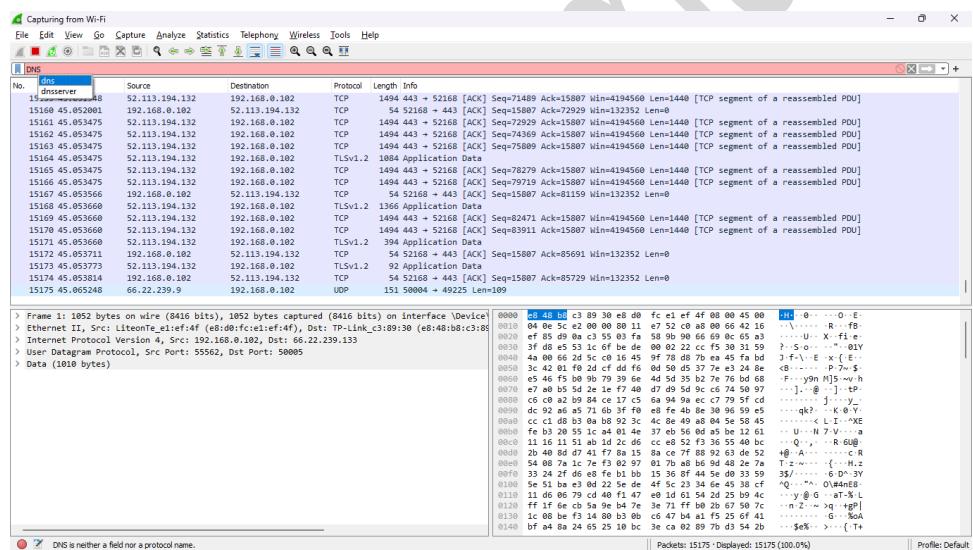


FIGURE 18 DNS FILTER

Step-5-> Search for some websites and check them on Wireshark, it will give you the following output

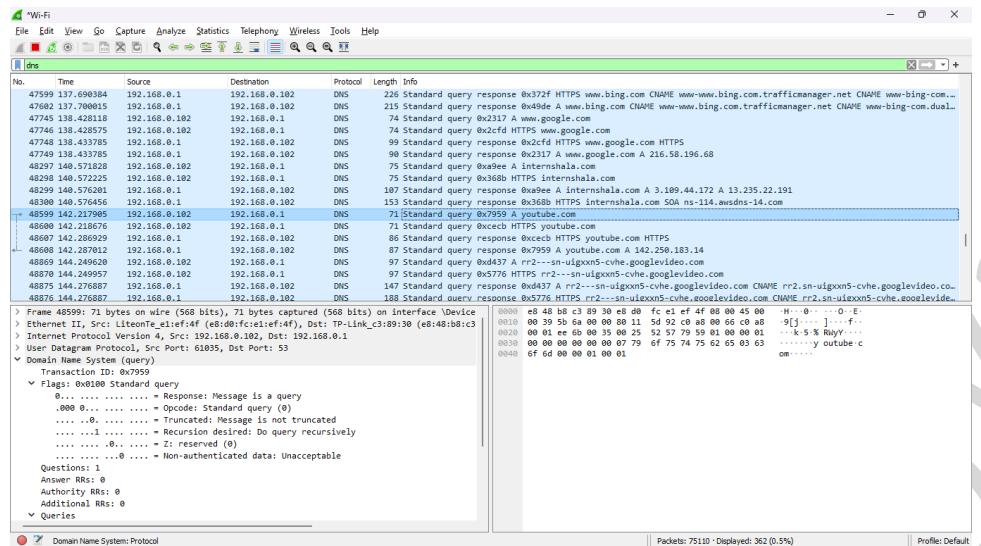


FIGURE 19 - PACKETS INCOMING FROM YOUTUBE.COM

Step-6-> Double click on that and open the DNS arrow and analyse it

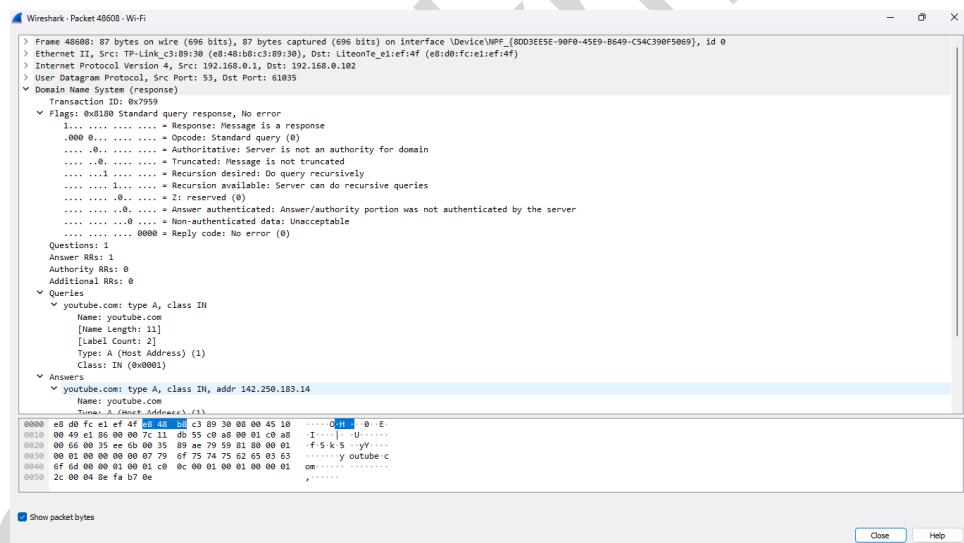


FIGURE 20 - OPENING A PACKET AND ANALYZING

PRACTICAL 5: CREATE YOUR OWN SYSLOG SERVER

Step 1-> Run sudo apt-get install openssh-server command to install ssh

```
ubuntu@ubuntu:~$ sudo apt-get install openssh-server
[sudo] password for ubuntu:
Reading package lists... Done
Building dependency tree
Reading state information... Done
openSSH-server is already the newest version (1:8.2p1-0.5).
The following package was automatically installed and
longer required:
  libfprint-2-tod1
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 32 not
d.
[REDACTED]
```

Step 2-> Run sudo apt-get install net-tools to install net-tools

```
ubuntu@ubuntu:~$ sudo apt-get install net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
net-tools is already the newest version (1.60+gitd88e-1ubuntu1).
The following package was automatically installed and
longer required:
  libfprint-2-tod1
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 32 not
d.
```

Step 3-> Run sudo service rsyslog restart to restart rsyslog service

```
ubuntu@ubuntu:/etc$ sudo service rsyslog restart
ubuntu@ubuntu:/etc$
```

Step 4-> Edit rsyslog config using sudo nano /etc/rsyslog.conf command and uncomment the lines

```
module(load="imudp") input(type="imudp" port="514")
module(load="imtcp") input(type="imtcp" port="514")
```

Press Ctrl+X & Enter to save file

```
ubuntu@ubuntu:/etc$ sudo nano /etc/rsyslog.conf
```

```
GNU nano 4.8      /etc/rsyslog.conf
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index>
#
# Default logging rules can be found in /etc/rsyslog.>

#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local >
#module(load="immark") # provides --MARK-- message c>

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
```

[Read 61 lines]

^G Get Help **^O** Write Out **^W** Where Is **^K** Cut Text
^X Exit **^R** Read File **^** Replace **^U** Paste Text

```
# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")
```

Step 5-> Restart rsyslog service with sudo service rsyslog restart command

```
ubuntu@ubuntu:/etc$ sudo service rsyslog restart
```

Step 6-> Check status of rsyslog service with sudo service rsyslog status command

```
ubuntu@ubuntu:/etc$ sudo service rsyslog status
● rsyslog.service - System Logging Service
  Loaded: loaded (/lib/systemd/system/rsyslog.service; enabled; re>
  Active: active (running) since Fri 2023-04-21 04:16:20 UTC
    Docs: man:rsyslogd(8)
          https://www.rsyslog.com/doc/
  Main PID: 14060 (rsyslogd)
    Tasks: 9 (limit: 3449)
   Memory: 1.2M
      CGrou: /system.slice/rsyslog.service
              └─14060 /usr/sbin/rsyslogd -n -iNONE

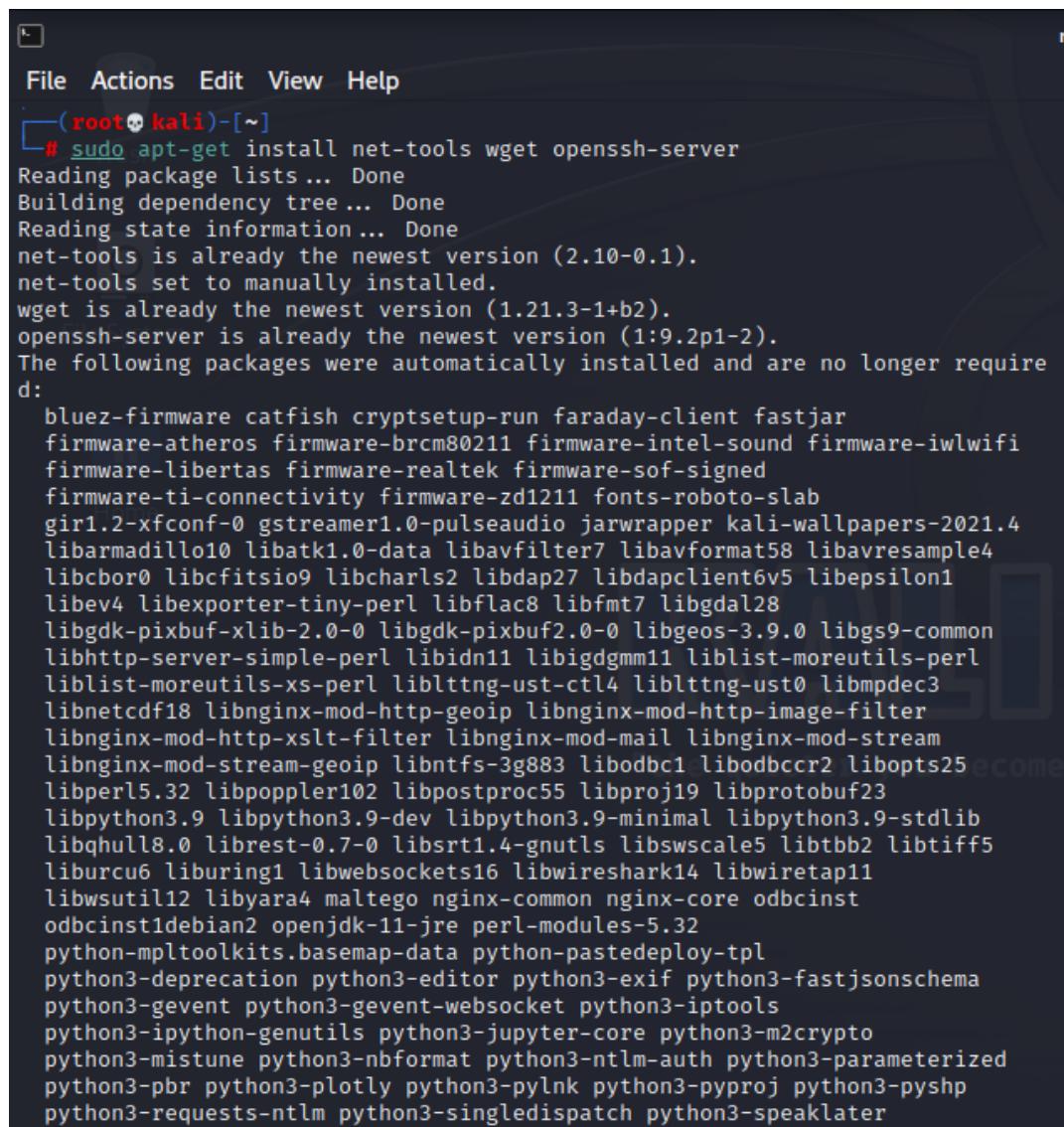
Terminal
```



```
Apr 21 04:16:20 ubuntu systemd[1]: Starting System Log Service...
Apr 21 04:16:20 ubuntu systemd[1]: Started System Log Service.
Apr 21 04:16:20 ubuntu rsyslogd[14060]: imuxsock: Acq...
Apr 21 04:16:20 ubuntu rsyslogd[14060]: rsyslogd's gr...
Apr 21 04:16:20 ubuntu rsyslogd[14060]: rsyslogd's us...
Apr 21 04:16:20 ubuntu rsyslogd[14060]: [origin softw...
lines 1-18/18 (END)
```

PRACTICAL 6: INSTALL AND RUN SPLUNK ON LINUX

Step-1-> Run `sudo apt-get update` , `sudo apt-get upgrade` primarily. Next run `sudo apt-get install net-tools wget openssh-server`. This install necessary tools for this practical.



```
File Actions Edit View Help
└─(root㉿kali)-[~]
# sudo apt-get update
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
net-tools is already the newest version (2.10-0.1).
net-tools set to manually installed.
wget is already the newest version (1.21.3-1+b2).
openssh-server is already the newest version (1:9.2p1-2).
The following packages were automatically installed and are no longer required:
  bluez-firmware catfish cryptsetup-run faraday-client fastjar
  firmware-atheros firmware-brcm80211 firmware-intel-sound firmware-iwlwifi
  firmware-libertas firmware-realtek firmware-sof-signed
  firmware-ti-connectivity firmware-zd1211 fonts-roboto-slab
  gir1.2-xfconf-0 gstreamer1.0-pulseaudio jarwrapper kali-wallpapers-2021.4
  libarmadillo10 libatk1.0-data libavfilter7 libavformat58 libavresample4
  libcbor0 libcfitsio9 libcharls2 libdap27 libdapclient6v5 libepsilon1
  libev4 libexporter-tiny-perl libflac8 libfmtp7 libgdal28
  libgdk-pixbuf-xlib-2.0-0 libgdk-pixbuf2.0-0 libgeos-3.9.0 libgs9-common
  libhttp-server-simple-perl libidn11 libigdgmm11 liblist-moreutils-perl
  liblist-moreutils-xs-perl liblttng-ust-ctl4 liblttng-ust0 libmpdec3
  libnetcdf18 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip libntfs-3g883 libodbc1 libodbccr2 libopts25
  libperl5.32 libpoppler102 libpostproc55 libproj19 libprotobuf23
  libpython3.9 libpython3.9-dev libpython3.9-minimal libpython3.9-stdlib
  libqhull8.0 libreast-0.7-0 libsrt1.4-gnutls libswscale5 libtbb2 libtiff5
  liburcu6 liburing1 libwebsockets16 libwireshark14 libwiretap11
  libwsutil12 libyara4 maltego nginx-common nginx-core odbcinst
  odbcinst1debian2 openjdk-11-jre perl-modules-5.32
  python-mpltoolkits.basemap-data python-pastedeploy-tpl
  python3-deprecation python3-editor python3-exif python3-fastjsonschema
  python3-gevent python3-gevent-websocket python3-iprotocols
  python3-ipython-genutils python3-jupyter-core python3-m2crypto
  python3-mistune python3-nbformat python3-ntlm-auth python3-parameterized
  python3-pbr python3-plotly python3-pynk python3-pyproj python3-pyshp
  python3-requests-ntlm python3-singledispatch python3-speaklater
```

FIGURE 21 - UPDATING AND UPGRADING KALI

Step-2-> Next restart ssh server using `sudo systemctl restart ssh` and run `sudo nano /etc/rsyslog.conf` to edit rsyslog configuration file.

```
(root㉿kali)-[~]
# sudo systemctl restart ssh

(root㉿kali)-[~]
# sudo nano /etc/rsyslog.conf

(root㉿kali)-[~]
#
```

FIGURE 22 - RESTARTING SSH SERVICE AND EDITING RSYSLOG CONFIGURATION FILE

Step-3-> Uncomment lines `module(load="imudp") input(type="imudp" port="514")` & `module(load="imtcp") input(type="imtcp" port="514")` and Press `Ctrl+X`, then `Y`, and `Enter`

```
GNU nano 7.2
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html

#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local system logging
module(load="imklog") # provides kernel logging support
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")
```

FIGURE 23 -UNCOMMENTING LINES

Step-4-> Run `sudo systemctl restart rsyslog` to restart rsyslog service. Run `sudo systemctl rsyslog status` to check status of service. If service is active, you can proceed.

The terminal window shows the following session:

```
File Actions Edit View Help
Unknown command verb rsyslog.

└─(root💀 kali)-[~]
  # sudo systemctl restart rsyslog

└─(root💀 kali)-[~]
  # sudo rsyslogd
rsyslogd: Error while binding tcp socket: Address already in use [v8.2302.0]
rsyslogd: Error while binding tcp socket: Address already in use [v8.2302.0]
rsyslogd: Could not create tcp listener, ignoring port 514 bind-address **UNSPECIFIED

└─(root💀 kali)-[~]
  # sudo systemctl rsyslog status
Unknown command verb rsyslog.

└─(root💀 kali)-[~]
  # sudo systemctl status rsyslog
● rsyslog.service - System Logging Service
  Loaded: loaded (/lib/systemd/system/rsyslog.service; enabled; preset: enabled)
  Active: active (running) since Sat 2023-04-01 00:01:35 IST; 55s ago
    TriggeredBy: ● syslog.socket
    Docs: man:rsyslogd(8)
           man:rsyslog.conf(5)
           https://www.rsyslog.com/doc/
   Main PID: 2202 (rsyslogd)
     Tasks: 10 (limit: 4594)
    Memory: 3.2M
      CPU: 26ms
     CGroup: /system.slice/rsyslog.service
             └─2202 /usr/sbin/rsyslogd -n -iNONE

Apr 01 00:01:35 kali systemd[1]: Starting rsyslog.service - System Logging Service ..
Apr 01 00:01:35 kali rsyslogd[2202]: imuxsock: Acquired UNIX socket '/run/systemd/jo
Apr 01 00:01:35 kali rsyslogd[2202]: [origin software="rsyslogd" swVersion="8.2302.0
Apr 01 00:01:35 kali systemd[1]: Started rsyslog.service - System Logging Service.

└─(root💀 kali)-[~]
```

FIGURE 24 - RESTARTING RSYSLOG SERVICE AND CHECKING STATUS

Step-5-> Add a group using `sudo groupadd <group_name>`. Add a user to that group using `sudo useradd -d /opt/<group_name> -m -g splunk splunkdemo`

The terminal window shows the following session:

```
└─(root💀 kali)-[~]
  # sudo groupadd splunkdemo

└─(root💀 kali)-[~]
  # sudo useradd -d /opt/splunk -m -g splunk splunkdemo
```

FIGURE 25 - ADDING GROUP AND NEW USER

Step-6-> Now go to <https://www.splunk.com/> . Register and Click on Free Splunk at top right of page. Click on Free Trials and Downloads page link. On the next screen Click on Get My Free Trial under Splunk Enterprise. Next Choose Linux and click on Download for .tgz. Stop the download and click

The screenshot shows the Splunk website interface. At the top, there is a navigation bar with links for Support, a search icon, a language selector (A), and a 'Free Splunk' button. Below this is a large orange header section titled 'GET STARTED' and 'Splunk Cloud Trial'. It contains text about searching, analyzing, and visualizing data in a Splunk hosted cloud environment, along with a 'Start Trial' button. Below this is a white section titled 'Splunk Enterprise'. It has text about downloading and installing the trial on hardware or cloud instances, and a 'Get My Free Trial' button. To the right of this text is a graphic featuring three overlapping triangles in pink, orange, and yellow, each containing a circular icon representing data analysis or visualization. At the bottom of the page, there are download links for different operating systems: Windows, Linux (selected), and Mac OS. For the 64-bit Linux section, there are three download options: '.rpm' (573.05 MB), '.tgz' (572.71 MB), and '.deb' (444.81 MB), each with a 'Download Now' button.

Platform	File Type	Size	Action
Windows	.rpm	573.05 MB	Download Now
Linux	.tgz	572.71 MB	Download Now
Mac OS	.deb	444.81 MB	Download Now

FIGURE 26 - PROCESS TO DOWNLOAD SPLUNK ON LINUX

Step-7-> Stop the download and click on Dowload via Command Line (wget) and copy the command.



FIGURE 27 - COPYING WGET COMMAND TO DOWNLOAD SPLUNK

Step-8-> Now change user using `su - <username>` and paste the command you copied before.

```
$ su -
Password:
[ (root💀 kali)-[~]
# su - splunkdemo
$ wget -O splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz "https://download.splunk.com/products/splunk/releases/9.0.4.1/linux/splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz"
--2023-04-01 00:17:49-- https://download.splunk.com/products/splunk/releases/9.0.4.1/linux/splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz
Resolving download.splunk.com (download.splunk.com) ... 13.227.138.113, 13.227.138.90, 13.227.138.114
Connecting to download.splunk.com (download.splunk.com)|13.227.138.113|:443 ...
HTTP request sent, awaiting response ... 200 OK
Length: 600530118 (573M) [binary/octet-stream]
Saving to: 'splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz'

splunk-9.0.4.1-419ad9369127-Linux-x86_64. 37%[=====→] 258.14M 7.96MB/s
splunk-9.0.4.1-419ad9369127-Linux-x86_64. 45%[=====→] 367.85M 7.72MB/s eta 26s ^
Linux-x86_64.tgz 64%[=====→] 367.85M 7.72MB/s eta 26s ^
splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz 100%[=====→] 572.71M 8.04MB/s in 71s

2023-04-01 00:19:06 (8.05 MB/s) - 'splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz' saved [600530118/600530118]
```

Step-9-> Now run `tar xvzf splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz -C /opt` (Note: the splunk version will differ with time. Change version number accordingly.)

```
$ tar xvzf splunk-9.0.4.1-419ad9369127-Linux-x86_64.tgz -C /opt
splunk/
splunk/swidtag/
splunk/swidtag/splunk-Splunk-Enterprise-primary.swidtag
splunk/ftr
splunk/openssl/
splunk/openssl/misc/
splunk/openssl/misc/c_info
splunk/openssl/misc/tsgt
splunk/openssl/misc/c_issuer
splunk/openssl/misc/CA.sh
```

FIGURE 28 - UNZIPPING TAR FILE

Step-10-> Now access root using su – and run chown -R splunk: /opt/splunk/ and change directory to /opt/splunk/bin

```
$ su -
Password:
[(root㉿kali)-[~]]# chown -R splunkdemo: /opt/splunk/
File System
[(root㉿kali)-[~]]# cd /opt/splunk/bin
```

FIGURE 29 - CHANGING OWNERSHIP

Step-11-> Start splunk service using ./splunk start . Hold spacebar to skip through the EULA.

```
[(root㉿kali)-[/opt/splunk/bin]]# ./splunk start
SPLUNK GENERAL TERMS

Last Updated: August 12, 2021

These Splunk General Terms ("General Terms") between Splunk Inc., a Delaware corporation, with its principal place of business at 270 Brannan Street, San Francisco, California 94107, U.S.A ("Splunk" or "we" or "us" or "our") and you ("Customer" or "you" or "your") apply to the purchase of licenses and subscriptions for Splunk's Offerings. By clicking on the appropriate button, or by downloading, installing, accessing or using the Offerings, you agree to these General Terms. If you are entering into these General Terms on behalf of Customer, you represent that you have the authority to bind Customer. If you do not agree to these General Terms, or if you are not authorized to accept the General Terms on behalf of the Customer, do not download, install, access, or use any of the Offerings.

See the General Terms Definitions Exhibit attached for definitions of capitalized terms not defined herein.

1. License Rights
(A) General Rights. You have the nonexclusive, worldwide, nontransferable and nonsublicensable right, subject to payment of applicable Fees and compliance with the terms of these General Terms, to use your Purchased Offerings for your Internal Business Purposes during the Term and up to the Capacity
```

FIGURE 30 – GENERAL TERMS & CONDITIONS OF SPLUNK

Step-12-> To accept splunk license run `./splunk start --accept-license`. You will be prompted to provide a username and password. This will be your login credentials to splunk on your linux system.

```
[root💀 kali]~[/opt/splunk/bin]
# ./splunk start --accept-license

This appears to be your first time running this version of Splunk.

Splunk software must create an administrator account during startup. Otherwise, you c
Create credentials for the administrator account.
Characters do not appear on the screen when you type in credentials.

Please enter an administrator username: root
Password must contain at least:
    * 8 total printable ASCII character(s).
Please enter a new password: "you become, the more you are able to hear"
Please confirm new password:
ERROR: Password did not meet complexity requirements. Password must contain at least:
    * 8 total printable ASCII character(s).
Please enter a new password:
Please confirm new password:
Copying '/opt/splunk/etc/openldap/ldap.conf.default' to '/opt/splunk/etc/openldap/lda
Generating RSA private key, 2048 bit long modulus
.....+++++
.....e is 65537 (0x10001)
```

FIGURE 31 - ACCEPTING LICENSE

```
File Actions Edit View Help
    All installed files intact.
    Done
All preliminary checks passed.

Starting splunk server daemon (splunkd) ...
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'privKeySecure.pem'

Signature ok
subject=/CN=kali/O=SplunkUser
Getting CA Private Key
writing RSA key
PYTHONHTTPSVERIFY is set to 0 in splunk-launch.conf disabling ce
must be set to "1" for increased security
Done

Waiting for web server at http://127.0.0.1:8000 to be available.

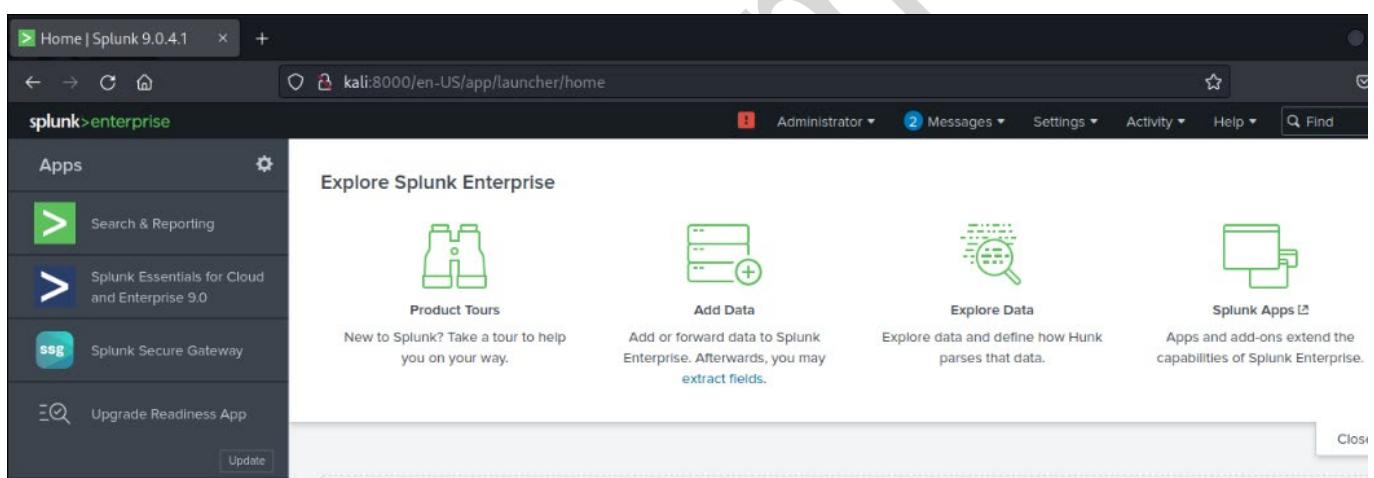
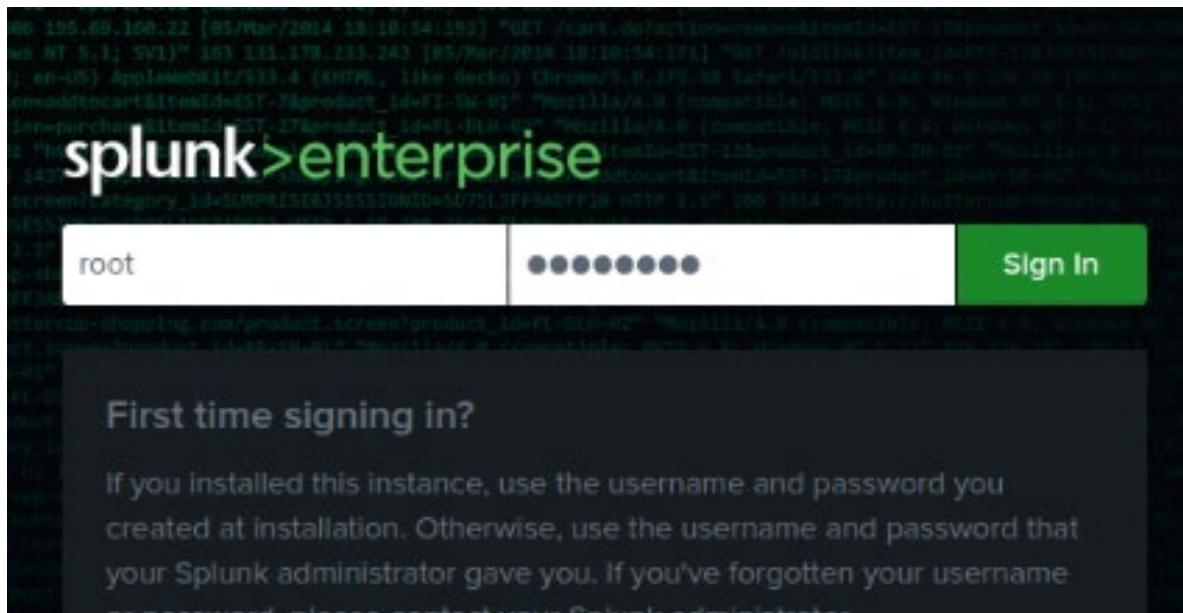
If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com

The Splunk web interface is at http://kali:8000

[root@kali]#/opt/splunk/bin
[root@kali]#/
```

FIGURE 32 - SPLUNK INTERFACE IS NOW ACCESSIBLE ON KALI:8000

Step-13-> Access the splunk web interface on <http://kali:8000>. You will see the following screen if you've followed every step to the mark. Now login using the username and password you were asked to provide while accepting license.



PRACTICAL 7: INSTALL AND CONFIGURE ELK ON LINUX

NOTE: To save a file Press CTRL+X, Y and Press Enter.

Step-1-> Run sudo apt-get update & sudo apt-get upgrade in terminal

```
(root㉿kali)-[~]
# sudo apt-get update
0% [Connecting to http.kali.org (192.99.200.113)] [Connected to brave-browser-apt-release.s3.^
Hit:1 https://brave-browser-apt-release.s3.brave.com stable InRelease
Get:2 https://kali.download/kali kali-rolling InRelease [41.2 kB]
Get:3 https://kali.download/kali kali-rolling/main amd64 Packages [19.3 MB]
Get:4 https://kali.download/kali kali-rolling/main amd64 Contents (deb) [44.4 MB]
Get:5 https://kali.download/kali kali-rolling/non-free amd64 Packages [217 kB]
Get:6 https://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [927 kB]
Fetched 64.8 MB in 29s (2,211 kB/s)
Reading package lists ... Done

(root㉿kali)-[~]
#



(root㉿kali)-[~]
# sudo apt-get upgrade
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Calculating upgrade ... Done
The following packages were automatically installed and are no longer required:
  bluez-firmware catfish cryptsetup-run faraday-client fastjar firmware-atheros
  firmware-brcm80211 firmware-intel-sound firmware-iwlwifi firmware-libertas
  firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211
  fonts-roboto-slab gir1.2-xfconf-0 gstreamer1.0-pulseaudio jarwrapper
  kali-wallpapers-2021.4 libarmadillo10 libatk1.0-data libavfilter7 libavformat58
  libavresample4 libcbor0 libcfitsio9 libcharls2 libdap27 libdapclient6v5 libepsilon1 libev4
  libexporter-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf-xlib-2.0-0
  libgdk-pixbuf2.0-0 libgeos-3.9.0 libgs9-common libhttp-server-simple-perl libidn11
  libigdmm11 liblist-moreutils-perl liblist-moreutils-xs-perl liblttng-ust-ctl4
  liblttng-ust0 libmpdec3 libnetcdf18 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip libntfs-3g883 libodbc1 libodbc2 libopt25 libperl5.32
  libpoppler102 libpostproc55 libproj19 libprotobuf23 libpython3.9 libpython3.9-dev
  libpython3.9-minimal libpython3.9-stdlib libqhull8.0 librest-0.7-0 libsrt1.4-gnutls
  libswscale5 libtbb2 libtiff5 liburcu6 liburing1 libwebsockets16 libwireshark14
  libwiredtap11 libwsutil12 libyara4 maltego nginx-common nginx-core odbcinst
  odbcinst1debian2 openjdk-11-jre perl-modules-5.32 python-mpltoolkits.basemap-data
  python-pastedeploy-tpl python3-deprecation python3-editor python3-exif
  python3-fastjsonschema python3-gevent python3-gevent-websocket python3-iputils
  python3-ipython-genutils python3-jupyter-core python3-m2crypto python3-mistune
```

FIGURE 33 - UPDATING AND UPGRADING KALI LINUX

Step-2-> Install java JDK using sudo apt-get install default-jdk default-jre command

```
(root💀 kali)-[~]
└─# sudo apt-get install default-jdk default-jre
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
default-jre is already the newest version (2:1.17-74).
default-jre set to manually installed.
The following packages were automatically installed and are no longer required:
  bluez-firmware catfish cryptsetup-run faraday-client fastjar firmware-atheros
  firmware-brcm80211 firmware-intel-sound firmware-iwlwifi firmware-libertas
  firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211
  fonts-roboto-slab gir1.2-xfconf-0 gstreamer1.0-pulseaudio jarwrapper
  kali-wallpapers-2021.4 libarmadillo10 libatk1.0-data libavfilter7 libavformat58
  libavresample4 libcbor0 libcfitsio9 libcharls2 libdap27 libdapclient6v5 libepsilon1 libev4
  libexporter-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf-xlib-2.0-0
  libgdk-pixbuf2.0-0 libgeos-3.9.0 libgs9-common libhttp-server-simple-perl libidn11
  libigdmm11 liblist-moreutils-perl liblist-moreutils-xs-perl liblttng-ust-ctl4
  liblttng-ust0 libmpdec3 libnetcdf18 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip libntfs-3g883 libodbc1 libodbccr2 libopts25 libperl5.32
  libpoppler102 libpostproc55 libproj19 libprotobuf23 libpython3.9 libpython3.9-dev
  libpython3.9-minimal libpython3.9-stdlib libqhull8.0 librest-0.7-0 libsrt1.4-gnutls
```

FIGURE 34 - INSTALLING JAVA JDK AND JRE

Step-3-> Check java version using java -version command

```
(root💀 kali)-[~]
└─# java -version
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
openjdk version "17.0.6" 2023-01-17
OpenJDK Runtime Environment (build 17.0.6+10-Debian-1)
OpenJDK 64-Bit Server VM (build 17.0.6+10-Debian-1, mixed mode, sharing)
```

FIGURE 35 - CHECKING JAVA INSTALLATION AND VERSION

Step-4-> Set the java configuration to use using update-alternatives --config java command and select the latest version installed.

```
(root💀 kali)-[~]
└─# update-alternatives --config java
There are 2 choices for the alternative java (providing /usr/bin/java).
      Selection    Path                                Priority      Status
* 0            /usr/lib/jvm/java-17-openjdk-amd64/bin/java  1711        auto mode
    1            /usr/lib/jvm/java-11-openjdk-amd64/bin/java  1111        manual mode
    2            /usr/lib/jvm/java-17-openjdk-amd64/bin/java  1711        manual mode

Press <enter> to keep the current choice[*], or type selection number: 0

(root💀 kali)-[~]
```

FIGURE 36 - SET JAVA VERSION TO USE

Step-5-> Now to update the system environment configuration enter sudo nano /etc/environment command.

The screenshot shows a terminal window with the following sequence of commands:

```
(root💀 kali)#[~]
# sudo nano /etc/environment

File Actions Edit View Help
GNU nano 7.2                               /etc/environment *
# START KALI-DEFAULTS CONFIG
# Everything from here and until STOP KALI-DEFAULTS CONFIG
# was installed by the kali-defaults package, and it will
# be removed if ever the kali-defaults package is removed.
# If you want to disable a line, please do NOT remove it,
# as it would be added back when kali-defaults is upgraded.
# Instead, comment the line out, and your change will be
# preserved across upgrades.
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/usr/games
COMMAND_NOT_FOUND_INSTALL_PROMPT=1
POWERSHELL_TELEMETRY_OPTOUT=1
POWERSHELL_UPDATECHECK=off
DOTNET_CLI_TELEMETRY_OPTOUT=1
JAVA_HOME="/usr/lib/jvm/java-17-openjdk-amd64/bin/java"
# STOP KALI-DEFAULTS CONFIG

"the quieter you become, the more you are able to hear"

^G Help      ^O Write Out    ^W Where Is     ^K Cut        ^T Execute   ^C Location
^X Exit      ^R Read File    ^\ Replace      ^U Paste      ^J Justify   ^/ Go To Line

(roots💀 kali)#[~]
# source /etc/environment

(roots💀 kali)#[~]
#
```

At the bottom of the terminal, there is a watermark that reads "the quieter you become, the more you are able to hear".

```
# echo $JAVA_HOME
/usr/lib/jvm/java-17-openjdk-amd64/bin/java
```

FIGURE 37 - UPDATING SYSTEM CONFIGURATION AND ADD JAVA_HOME VARIABLE

Step-6-> Now to access remote repositories install APT Transport package using `sudo apt-get install apt-transport-https` command.

```
(root㉿kali)-[~]
# sudo apt-get install apt-transport-https
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apt-transport-https is already the newest version (2.6.0).
The following packages were automatically installed and are no longer required:
bluez-firmware catfish cryptsetup-run faraday-client fastjar firmware-atheros
firmware-brcm80211 firmware-intel-sound firmware-iwlwifi firmware-libertas
firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211
fonts-roboto-slab gir1.2-xfconf-0 gstreamer1.0-pulseaudio jarwrapper
kali-wallpapers-2021.4 libarmadillo10 libatk1.0-data libavfilter7 libavformat58
libavresample4 libcbor0 libcfitsio9 libcharls2 libdapclient6v5 libepsilon1 libev4
libexporter-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf-xlib-2.0-0
libgdk-pixbuf2.0-0 libgeos-3.9.0 libgs9-common libhttp-server-simple-perl libidn11
libigdgmm11 liblist-moreutils-perl liblist-moreutils-xs-perl liblttng-ust-ctl4
liblttng-ust0 libmpdec3 libnetcdf18 libnginx-mod-http-geoip libnginx-mod-http-image-filter
libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
libnginx-mod-stream-geoip libntfs-3g883 libodbc1 libodbc2 libopts25 libperl5.32
libpoppler102 libpostproc55 libproj19 libprotobuf23 libpython3.9 libpython3.9-dev
libpython3.9-minimal libpython3.9-stdlib libqhull8.0 librest-0.7-0 libsrt1.4-gnutls
libswscale5 libtbb2 libtiff5 liburcu6 liburing1 libwebsockets16 libwireshark14
libwiredtap11 libwsutil12 libyara4 maltego nginx-common nginx-core odbcinst
odbcinst1debian2 openjdk-11-jre perl-modules-5.32 python-mpltoolkits.basemap-data
python-pastedeploy-tpl python3-deprecation python3-editor python3-exif
python3-fastjsonschema python3-gevent python3-gevent-websocket python3-iprotocols
python3-ipython-genutils python3-jupyter-core python3-m2crypto python3-mistune
python3-nbformat python3-ntlm-auth python3-parameterized python3-pbr python3-plotly
python3-pylnk python3-pyproj python3-pyshp python3-requests-ntlm python3-singledispatch
python3-speaklater python3-stem python3-tenacity python3-zope.event python3.9
python3.9-dev python3.9-minimal ruby-atomic ruby-thread-safe ruby2.7 ruby2.7-dev
starkiller zaproxy
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

FIGURE 38 - INSTALLING APT-TRANSPORT HTTPS

Step-7-> Now run `wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg` command. This command stores a gpg key fetched from the repository URL and decrypts and store it in keyrings folder.

```
(root㉿kali)-[~]
# wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg
#
#
```

FIGURE 39 - STORING GPG KEY FROM ELASTIC SEARCH KEYSTORE

Step-8-> Now run echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elasticsearch-8.x.list command. Also run sudo apt-get update command

```
[root@kali:~]
# echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elasticsearch-8.x.list
deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/8.x/apt stable main
```

```
[root@kali:~]
# sudo apt-get update
Hit:1 https://brave-browser-apt-release.s3.brave.com stable InRelease
Get:2 https://artifacts.elastic.co/packages/8.x/apt stable InRelease [10.4 kB]
Get:3 https://artifacts.elastic.co/packages/8.x/apt stable/main amd64 Packages [48.4 kB]
Get:4 https://artifacts.elastic.co/packages/8.x/apt stable/main amd64 Contents (deb) [1,217 kB]
Hit:5 https://kali.download/kali kali-rolling InRelease
Fetched 1,275 kB in 1s (889 kB/s)
Reading package lists... Done
```

FIGURE 40 - SETTING REPOSITORY LOCATION AND UPDATING KALI AGAIN

Step-9-> Now install elasticsearch using sudo apt-get install elasticsearch command

```
[root@kali:~]
# sudo apt-get install elasticsearch
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
bluez-firmware calfish cryptsetup-rw faraday-client fastjar firmware-atheros firmware-brcm80211 firmware-intel-sound firmware-iwlwifi firmware-libertas
firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211 fonts-roboto-slab girl1.2-xfconf-0 gstreamer1.0-pulseaudio jarwapper
kali-wallpapers-2021.4 libarmadillo0 libatk1.0-data libavfilter7 libavformat58 libavresample4 libchromo0 libcurl27 libdapclientv5 libesilon1
libev libevent-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf2.0-0 libgdk-pixbuf2.0-0 libgeos-3.9.0 libgs9-common libhttp-server-simple-perl libidn11
libigdgmm11 liblist-moreutils-perl liblist-moreutils-xs-perl liblittng-ust0 libmpdec3 libnetcdf18 libnginx-mod-http-geoip
libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libproto19 libprotobuf23 libpython3.9 libpython3.9-dev libpython3.9-minimal libpython3.9-stdlib libqbcr2
libiptc5 libperl5.32 libpoppler102 libpostproc5 libpro19 libprotobuf23 libpython3.9 libpython3.9-dev libpython3.9-minimal libpython3.9-stdlib libqhull8.0
librest-0.7-0 libsrtp1.4-gnutls libwscale5 libtiff2 liburcu6 liburing1 libwebsockets16 libwireshark14 libwiredtap11 libwsutil12 libyara4 maltego nginx-common
nginx-core odbcinst odbcinst1debian2 openjdk-11-jre perl-modules-5.32 python-mpltoolkits.basemap-data python-paste-deploy-tpl python3-deprecation python3-editor
python3-exif python3-fastjsonschema python3-gevent-websocket python3-iputils python3-ipython-genutils python3-jupyter-core python3-m2crypto
python3-mistun python3-nbformat python3-ntlm-auth python3-parameterized python3-pbr python3-plotly python3-pylnk python3-pyproj python3-psyphp python3-requests-ntlm
python3-singledispatch python3-speaklater python3-stem python3-tenacity python3-zope.event python3.9 python3.9-dev python3.9-minimal ruby-atomic ruby-thread-safe
ruby2.7 ruby2.7-dev starkiller zaproxy
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
elasticsearch
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 596 MB of archives.
After this operation, 1,234 MB of additional disk space will be used. "the quieter you become, the more you are able to hear"
Get:1 https://artifacts.elastic.co/packages/8.x/apt stable/main amd64 elasticsearch amd64 8.7.0 [596 kB]
Fetched 596 MB in 1min 44s (5,716 kB/s)
Selecting previously unselected package elasticsearch.
(Reading database ... 420593 files and directories currently installed.)
Preparing to unpack .../elasticsearch_8.7.0_amd64.deb ...
Creating elasticsearch group... OK
Creating elasticsearch user... OK
Unpacking elasticsearch (8.7.0) ...
Setting up elasticsearch (8.7.0) ...
    Security autoconfiguration information
```

FIGURE 41 - INSTALLING ELASTIC SEARCH

Step-10-> Now we have to restart linux daemon, start and enable elasticsearch service

```
[root@kali:~]
# sudo systemctl daemon-reload
[root@kali:~]
# sudo systemctl enable elasticsearch.service
Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service → /lib/systemd/system/elasticsearch.service.

[root@kali:~]
# sudo systemctl start elasticsearch.service
[root@kali:~]
# sudo systemctl status elasticsearch.service
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; preset: disabled)
     Active: active (running) since Mon 2023-04-03 21:43:21 IST; 2min 10s ago
       Docs: https://www.elastic.co
      Main PID: 3776 (java)
         Tasks: 74 (limit: 4594)
        Memory: 2.3G
           CPU: 1min 27.19s
          CGroup: /system.slice/elasticsearch.service
                  └─3776 /usr/share/elasticsearch/jdk/bin/java -Xms4m -Xmx64m -XX:+UseSerialGC -Dcli.name=server -Dcli.script=/usr/share/elasticsearch/bin/elasticsearch -Djava.security.manager=allow -XX:+
                  ├─3835 /usr/share/elasticsearch/jdk/bin/java -Des.networkaddress.cache.ttl=60 -Des.networkaddress.cache.negative.ttl=10 -Djava.security.manager=allow -XX:+
                  ├─3857 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-x86_64/bin/controller

Apr 03 21:42:32 kali systemd[1]: Starting elasticsearch.service - Elasticsearch...
Apr 03 21:43:21 kali systemd[1]: Started elasticsearch.service - Elasticsearch.
Lines 1-15/15 (END)
```

FIGURE 42 - STARTING AND ENABLING ELASTIC SEARCH SERVICE

Step-11-> Now edit the elasticsearch.yml file using the sudo nano /etc/elasticsearch/elasticsearch.yml command. Uncomment network.host and give address 0.0.0.0. In discovery section add this line in it discovery.seed_hosts: [] as it is. Also in same file change the value to false for line xpack.security.enabled: to false.

```
[root@kali]# sudo nano /etc/elasticsearch/elasticsearch.yml
```

```
GNU nano 7.2                                     /etc/elasticsearch/elasticsearch.yml *
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ----- Network -----
#
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: 0.0.0.0
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
#http.port: 9200
#
# For more information, consult the network module documentation.
#
# ----- Discovery -----
#
# Pass an initial list of hosts to perform discovery when this node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
discovery.seed_hosts: []
#
# Bootstrap the cluster using an initial set of master-eligible nodes:
#
#-----
```

```
GNU nano 7.2                                     /etc/elasticsearch/elasticsearch.yml *
#-----[Truncated]
#action.destructive_requires_name: false
#
#----- BEGIN SECURITY AUTO CONFIGURATION -----
#
# The following settings, TLS certificates, and keys have been automatically
# generated to configure Elasticsearch security features on 03-04-2023 16:09:59
#
#----- File System -----
#
#
# Enable security features
xpakc.security.enabled: false
```

FIGURE 43 - EDITING ELASTICSEARCH.YML FILE

Step-12-> Now, restart elastic search service using `sudo systemctl restart elasticsearch.service` command

```
[root@kali]# sudo systemctl restart elasticsearch.service
```

FIGURE 44 - RESTARTING ELASTIC SEARCH SERVICE

Step-13-> Now edit jvmoptions for elastic search using `sudo nano /etc/elasticsearch/jvm.options` command. Change line `-Xms4g` and `-Xmx4g` to `-Xms512m` and `-Xmx512m`.

```
[root@kali]# sudo nano /etc/elasticsearch/jvm.options
```

```
GNU nano 7.2                                     /etc/elasticsearch/jvm.options *
## each node is configured to fulfill. If specifying heap is
## required, it should be done through a file in jvm.options.d,
## which should be named with .options suffix, and the min and
## max should be set to the same value. For example, to set the
## heap to 4 GB, create a new file in the jvm.options.d
## directory containing these lines:
##
-Xms512m
-Xmx512m
##
## See https://www.elastic.co/guide/en/elasticsearch/reference/8.7/heap-size.html
## for more information
```

FIGURE 45 - EDITING JVM OPTIONS FOR ELASTIC SEARCH

Step-14-> Now GET the data at `localhost:9200` using `curl -X GET "localhost:9200"` command. This command is used to check if elastic search is running on localhost.

```
[root@kali]# curl -X GET "localhost:9200"
{
  "name" : "kali",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "zSetRChLROe1U6_NJrk2KQ",
  "version" : {
    "number" : "8.7.0",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "09520b59b6bc1057340b55750186466ea715e30e",
    "build_date" : "2023-03-27T16:31:09.816451435Z",
    "build_snapshot" : false,
    "lucene_version" : "9.5.0",
    "minimum_wire_compatibility_version" : "7.17.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

FIGURE 46 - CURL COMMAND TO CHECK STATUS OF ELASTIC SERARCH SERVICE

Step-15-> Now copy the inet address of ethernet using ifconfig command. Now

```
[root@kali]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.0.103 netmask 255.255.255.0 broadcast 192.168.0.255
        inet6 fe80::a00:27ff:fe30:1c43 prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:30:1c:43 txqueuelen 1000 (Ethernet)
            RX packets 645094 bytes 949813902 (905.8 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 154022 bytes 12094107 (11.5 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 30 bytes 2447 (2.3 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 30 bytes 2447 (2.3 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

FIGURE 47 - INTERFACE CONFIGURATION COMMAND

Step-16-> Enter the address into a browser and append port :9200 to the IP address

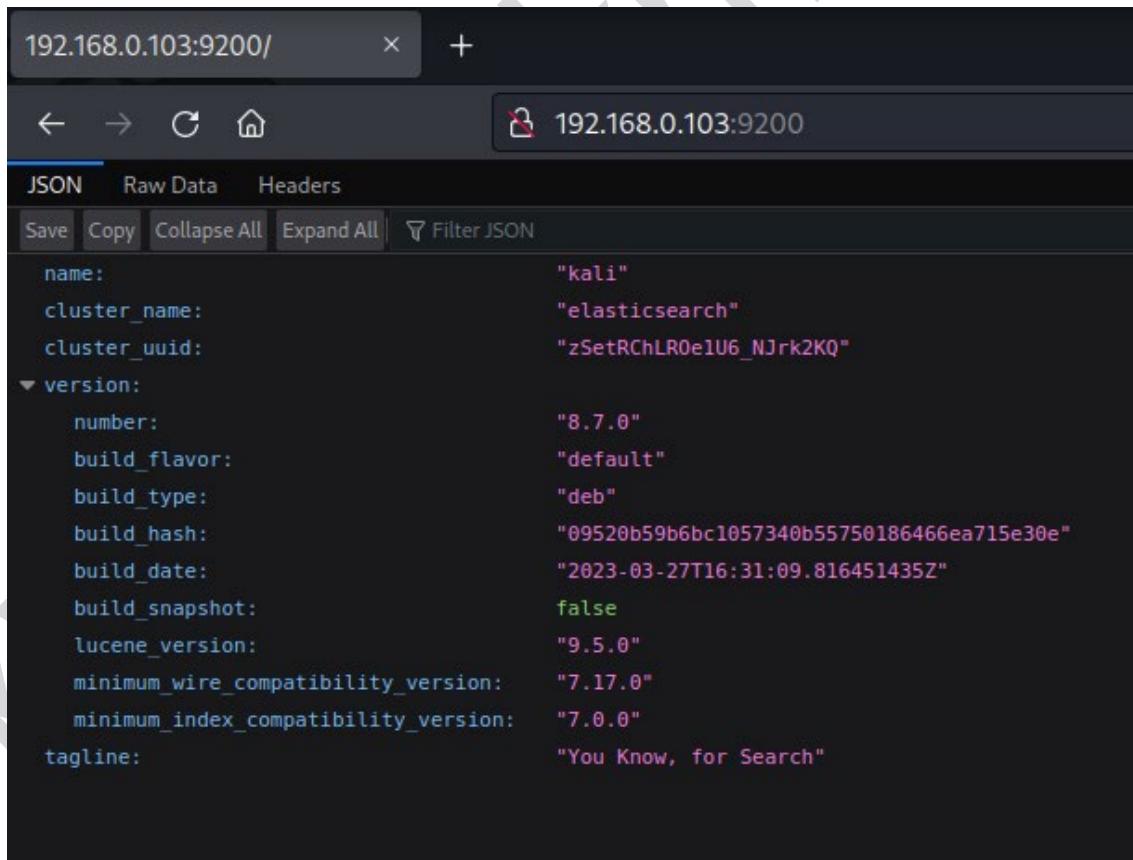


FIGURE 48 - LOCALHOST:9200 IN A BROWSER

Step-17-> Now to install and enable Logstash use the commands `sudo apt-get install logstash`, `sudo systemctl start logstash`, `sudo systemctl enable logstash`, `sudo systemctl status logstash`.

```
(root㉿kali)-[~]
└─# sudo apt-get install logstash
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
bluez-firmware catfish cryptsetup-run faraday-client fastjar firmware-atheros firmware-brcm80211 firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211 fonts-roboto-slab gir1.2-kali-wallpapers-2021.4 libarmadillo10 libatk1.0-data libavfilter7 libavformat58 libavresample4 libcairo2 libev4 libexporter-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf-xlib-2.0-0 libgdk-pixbuf2.0-0 libigdmm11 liblist-moreutils-perl liblist-moreutils-xs-perl liblttng-ust-ctl4 liblttng-ust0 libmpdec2 libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libnptls25 libperl5.32 libpoppler102 libpostproc55 libproj19 libprotobuf23 libpython3.9 libpython3.9-dev libre-0.7-0 librest1.4-gnutls libsswscale5 libtbb2 libtiff5 liburcu6 liburing1 libwebsockets16 libwireless-common nginx-core odbcinst1debian2 openjdk-11-jre perl-modules-5.32 python-mpoolkits python3-editor python3-exif python3-fastjsonschema python3-gevent python3-geventwebsocket python3-ipmi python3-m2crypto python3-mistune python3-nbformat python3-ntlm-auth python3-parameterized python3-pbr python3-requests-ntlm python3-singledispatch python3-speaklater python3-stem python3-tenacity python3-ruby-atomic ruby-thread-safe ruby2.7 ruby2.7-dev starkiller zaproxy
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
logstash
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 327 MB of archives.
After this operation, 579 MB of additional disk space will be used. become, the more you are
Get:1 https://artifacts.elastic.co/packages/8.x/apt stable/main amd64 logstash amd64 1:8.7.0-1 [327 MB]
Fetched 327 MB in 1min 11s (4,633 kB/s)
Selecting previously unselected package logstash.
(Reading database ... 421868 files and directories currently installed.)
Preparing to unpack .../logstash_1%3a8.7.0-1_amd64.deb ...
Unpacking logstash (1:8.7.0-1) ...
Setting up logstash (1:8.7.0-1) ...

[root㉿kali)-[~]
└─#
```



```
(root㉿kali)-[~]
└─# sudo systemctl start logstash
      "8.7.0"
      "default"
      "deb"
[root㉿kali)-[~]
└─# sudo systemctl enable logstash
      "09520659b6bc1057340b55750186466ea715e30e"
Created symlink /etc/systemd/system/multi-user.target.wants/logstash.service → /lib/systemd/system/logstash.service.
      "false"
[root㉿kali)-[~]
└─# sudo systemctl enable logstash
      "9.5.0"
      "7.17.0"
[root㉿kali)-[~]
└─# sudo systemctl status logstash
      "7.0.0"
      "You Know, for Search"
● logstash.service - logstash
   Loaded: loaded (/lib/systemd/system/logstash.service; enabled; preset: disabled)
   Active: active (running) since Mon 2023-04-03 22:14:53 IST; 34s ago
     Main PID: 5160 (java)
        Tasks: 22 (limit: 4594)
       Memory: 381.5M
          CPU: 55.095s
         CGroupl: /system.slice/logstash.service
                   5160 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -Djava.awt.headless=true
Apr 03 22:14:53 kali systemd[1]: Started logstash.service - logstash.
Apr 03 22:14:53 kali logstash[5160]: Using bundled JDK: /usr/share/logstash/jdk
lines 1-12/12 (END)
```

FIGURE 49 - INSTALLING AND ENABLING LOGSTATSH

Step-18-> Now install and enable Kibana use the commands `sudo apt-get install kibana`, `sudo systemctl start kibana`, `sudo systemctl enable kibana`, `sudo systemctl status kibana`.

```
└─# sudo apt-get install kibana
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  bluez-firmware catfish cryptsetup-run faraday-client fastjar firmware-atheros firmware-brcm80211
  firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211 fonts-roboto-slab
  kali-wallpapers-2021.4 libarmadillo10 libatk1.0-data libavfilter7 libavformat58 libavresample4.1
  libev4 libexporter-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf-xlib-2.0-0 libgdk-pixbuf2.
  libigdgmm11 liblist-moreutils-perl liblist-moreutils-xs-perl liblttng-ust-ctl4 liblttng-ust0 lib
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stre
  libopts25 libperl5.32 libpoppler102 libpostproc55 libprotobuf23 libpython3.9 libpython
  librest-0.7-0 libsrt1.4-gnutls libsscale2 libtiff5 liburcu6 liburing1 libwebsockets16 l
  nginx-common nginx-core odbcinst odbcinst1debian2 openjdk-11-jre perl-modules-5.32 python-mpltoo
  python3-editor python3-exif python3-fastjsonschema python3-gevent python3-gevent-websocket python
  python3-m2crypto python3-mistune python3-nbformat python3-ntlm-auth python3-parameterized python
  python3-requests-ntlm python3-singledispatch python3-speaklater python3-stem python3-tenacity py
  ruby-atomic ruby-thread-safe ruby2.7 ruby2.7-dev starkiller zaproxy
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  kibana
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 240 MB of archives.
After this operation, 629 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/8.x/apt stable/main amd64 kibana amd64 8.7.0 [240 MB]
11% [1 kibana 32.4 MB/240 MB 14%]
Fetched 240 MB in 60s (3,980 kB/s)
Selecting previously unselected package kibana.
(Reading database ... 436144 files and directories currently installed.)
Preparing to unpack .../kibana_8.7.0_amd64.deb ...
Unpacking kibana (8.7.0) ...
Setting up kibana (8.7.0) ...
Creating kibana group ... OK
Creating kibana user ... OK
Created Kibana keystore in /etc/kibana/kibana.keystore
```

```
└─(root💀kali)-[~]
  # sudo systemctl start kibana
  ● kibana.service - Kibana
    Loaded: loaded (/lib/systemd/system/kibana.service; enabled; preset: disabled)
    Active: active (running) since Mon 2023-04-03 23:35:00 IST; 33s ago
      Docs: https://www.elastic.co
      Main PID: 13539 (node)
        Tasks: 11 (limit: 4594)
       Memory: 254.5M
          CPU: 28.669s
        CGroup: /system.slice/kibana.service
                  └─13539 /usr/share/kibana/bin/.. /node/bin/node /usr/share/kibana/bin/..
Apr 03 23:35:00 kali systemd[1]: Started kibana.service - Kibana.
Apr 03 23:35:09 kali kibana[13539]: [2023-04-03T23:35:09.227+05:30][INFO ][node] Kib
```

FIGURE 50 - INSTALLING AND ENABLING KIBANA SERVER

Step-19-> Now edit the kibana.yml file using sudo nano /etc/kibana/kibana.yml command.

Uncomment these lines: server.port:5601, server.host: "localhost" to server.host: "0.0.0.0"
elasticsearch.hosts: ["http://localhost:9200"]

```
(root㉿kali)-[~]
# sudo nano /etc/kibana/kibana.yml

GNU nano 7.2                               /etc/kibana/kibana.yml *
# For more configuration options see the configuration guide for Kibana in
# https://www.elastic.co/guide/index.html

# ===== System: Kibana Server =====
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: 5601

# Specifies the address to which the Kibana server will bind. IP addresses and host names are both valid values
# The default is 'localhost', which usually means remote machines will not be able to connect.
# To allow connections from remote users, set this parameter to a non-loopback address.
server.host: "0.0.0.0"

# ===== System: Elasticsearch =====
# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["http://localhost:9200"]
```

FIGURE 51 - EDITING KIBANA.YML

Step-20-> Restart kibana service using sudo systemctl restart kibana command

```
(root㉿kali)-[~]
# sudo systemctl restart kibana
```

FIGURE 52 - RESTARTING KIBANA SERVER

Step-21-> Now install filebeat using sudo apt-get install filebeat command.

```
[root💀kali]-[~] 192.168.0.103:3200
# sudo apt-get install filebeat
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
bluez-firmware catfish cryptsetup-run faraday-client fastjar firmware-atheros firmware-brcm80211 fi
firmware-realtek firmware-sof-signed firmware-ti-connectivity firmware-zd1211 fonts-roboto-slab gi
kali-wallpapers-2021.4 libarmadillo10 libatk1.0-data libavfilter7 libavformat58 libavresample4 lib
libev4 libexporter-tiny-perl libflac8 libfmt7 libgdal28 libgdk-pixbuf-xlib-2.0-0 libgdk-pixbuf2.0-0
libigmm11 liblist-moreutils-perl liblist-moreutils-xs-perl libltng-ust-ctl4 libltng-ust0 libmpc
libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
libopts25 libperl5.32 libpoppler102 libpostproc55 libproj19 libprotobuf23 libpython3.9 libpython3.9
librest-0.7-0 libsrt1.4-gnutls libswscale5 libtbb2 libtiff5 liburcu6 liburing1 libwebsockets16 libw
nginx-common nginx-core odbcinst1debian2 openjdk-11-jre perl-modules-5.32 python-mpoolkit
python3-editor python3-exif python3-fastjsonschema python3-gevent python3-gevent-websocket python3-
python3-m2crypto python3-mistune python3-nbformat python3-ntlm-auth python3-parameterized python3-p
python3-requests-ntlm python3-singledispatch python3-speaklater python3-stem python3-tenacity pytho
ruby-atomic ruby-thread-safe ruby2.7 ruby2.7-dev starkiller zaproxy
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
filebeat
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 42.5 MB of archives.
After this operation, 157 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/8.x/apt stable/main amd64 filebeat amd64 8.7.0 [42.5 MB]
Fetched 42.5 MB in 8s (5,168 kB/s)
Selecting previously unselected package filebeat.
(Reading database ... 482966 files and directories currently installed.)
Preparing to unpack .../filebeat_8.7.0_amd64.deb ...
Unpacking filebeat (8.7.0) ...
Setting up filebeat (8.7.0) ...
Processing triggers for kali-menu (2023.1.7) ...
```

FIGURE 53 - INSTALLING FILEBEAT

Step-22-> Now edit filebeat.yml using sudo nano /etc/filebeat/filebeat.yml command. comment this line: output.elasticsearch: Array of hosts to connect to. hosts: ["localhost:9200"]. Uncomment this line: output.logstash hosts: ["0.0.0.0:5044"]

```
(root㉿kali)-[~]
# sudo nano /etc/filebeat/filebeat.yml

# ----- Elasticsearch Output -----
#output.elasticsearch:
#  # Array of hosts to connect to.
#  #hosts: ["localhost:9200"]

#  # Protocol - either `http` (default) or `https`.
#  #protocol: "https"

#  # Authentication credentials - either API key or username/password.
#  #api_key: "id:api_key"
#  #username: "elastic"
#  #password: "changeme"

# ----- Logstash Output -----
output.logstash:
  # The Logstash hosts
  hosts: ["localhost:5044"]

  # Optional SSL. By default is off.
```

FIGURE 54 - EDITING KIBANA.YML

Step-23-> Now enable filebeat modules for the system using sudo filebeat modules enable system command

```
(root㉿kali)-[~]
# sudo filebeat modules enable system
Enabled system module.

(root㉿kali)-[~]
# sudo filebeat setup --index-management -E output.logstash.enabled=false -E 'output.elasticsearch.hosts=["0.0.0.0:9200"]'
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite: true` for enabling.

Index setup finished.

(root㉿kali)-[~]
# sudo systemctl start filebeat
[root@kali ~]# debconf-set-selections /var/lib/debconf
(root㉿kali)-[~]
# sudo systemctl enable filebeat
Synchronizing state of filebeat.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable filebeat
Created symlink /etc/systemd/system/multi-user.target.wants/filebeat.service → /lib/systemd/system/filebeat.service.
```

FIGURE 55 - STARTING AND ENABLING FILEBEAT

Step-24-> Now run the curl command `curl -X GET http://{your_ip_address}:9200/_cat/indices?v`.



```
File Actions Edit View Help
(kali㉿kali)-[~]
$ curl -X GET http://192.168.1.10:9200/_cat/indices?v
health status index          uuid                                pri rep docs.count docs.deleted store.size pri.store.size
yellow open   .ds-filebeat-8.7.0-2023.04.03-000001 5YWVCG2DStar21GySmMwJQ    1   1      0           0        225b       225b
```

FIGURE 56 - CURL COMMAND TO GET ALL INDICES

Step-25-> Now open any browser, type `http://{your_ip_address}:9200/_cat/indices?v` and check if you get the same output.



FIGURE 57 - SAME COMMAND IN BROWSER

Step-26-> Now change the same url to `http://{your_ip_address}:5601`.

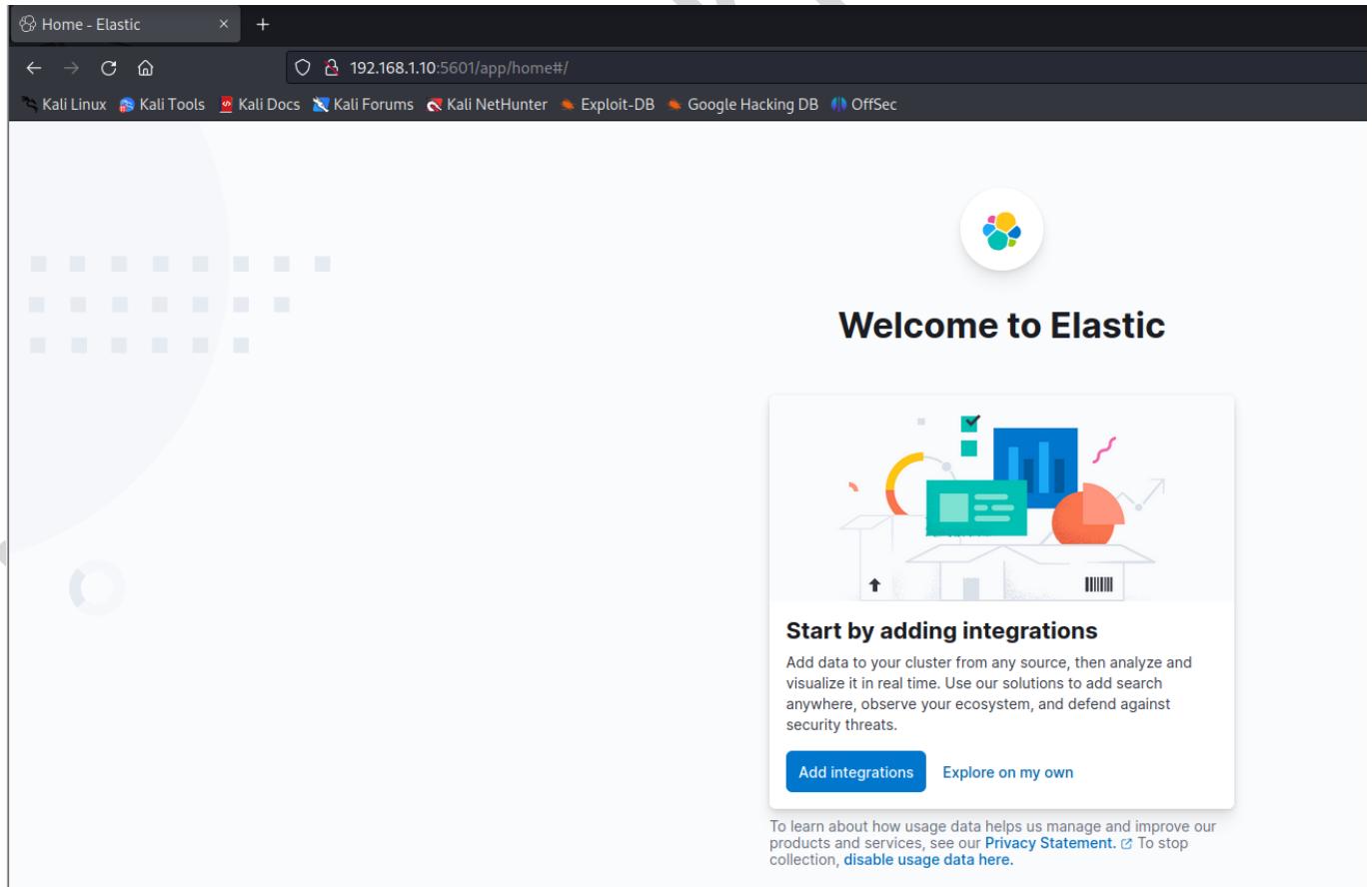


FIGURE 58 - ELASTIC SEARCH RUNNING ON PORT 5601

PRACTICAL 8: INSTALL AND CONFIGURE GRAYLOG ON LINUX

Step 1-> Run sudo apt-get update & sudo apt-get upgrade commands before proceeding further

```
(kali㉿kali)-[~]
$ sudo apt-get update
[sudo] password for kali:
Get:1 http://kali.download/kali kali-rolling InRelease [41.2 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [19.3 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [44.5
MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [116 kB]
Get:5 http://kali.download/kali kali-rolling/non-free amd64 Packages [217 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [9
27 kB]
Fetched 65.1 MB in 37s (1,749 kB/s)
Reading package lists ... Done
```

FIGURE 59 - UPDATING PACKAGES

```
(kali㉿kali)-[~]
$ sudo apt-get upgrade
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Calculating upgrade ... Done
The following packages were automatically installed and are no longer require
d:
  bluez-firmware firmware-ath9k-htc firmware-atheros firmware-brcm80211
  firmware-intel-sound firmware-iwlwifi firmware-libertas firmware-realtek
  firmware-sof-signed firmware-ti-connectivity firmware-zd1211
  kali-linux-firmware
Use 'sudo apt autoremove' to remove them.
The following packages have been kept back:
  kali-desktop-xfce libcurl3-nss libgtk-4-1 libgtk-4-bin libpocl2
  libpocl2-common linux-image-amd64 nginx pocl-opencl-icd python3-pypykatz
The following packages will be upgraded:
  adduser apache2 apache2-bin apache2-data apache2-utils bulk-extractor
  cryptcat curl dirbuster e2fsprogs easy-rsa exploitdb fakeroot faraday
  firefox-esr ipp-usb kali-desktop-core kali-linux-core kali-linux-default
  kali-linux-firmware kali-linux-headless kali-menu kali-system-cli
  kali-system-core kali-system-gui kali-tools-top10 laudanum
  libapache2-mod-php8.2 libbtlapi0.8 libc-bin libc-dev-bin libc-devtools
  libc-l10n libc6 libc6-dev libc6-i386 libcom-err2 libcurl3-gnutls
```

FIGURE 60 - UPGRADING PACKAGES

Step 2-> Install jdk and jre using `sudo apt-get install default-jdk default-jre` command.

```
(kali㉿kali)-[~]
$ sudo apt-get install default-jdk default-jre
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
default-jre is already the newest version (2:1.17-74).
default-jre set to manually installed.
The following packages were automatically installed and are no longer required:
  bluez-firmware firmware-ath9k-htc firmware-atheros firmware-brcm80211
  firmware-intel-sound firmware-iwlwifi firmware-libertas firmware-realtek
  firmware-sof-signed firmware-ti-connectivity firmware-zd1211
  kali-linux-firmware
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  default-jdk-headless openjdk-17-jdk openjdk-17-jdk-headless
Suggested packages:
  openjdk-17-demo openjdk-17-source visualvm
The following NEW packages will be installed:
  default-jdk default-jdk-headless openjdk-17-jdk openjdk-17-jdk-headless
0 upgraded, 4 newly installed, 0 to remove and 10 not upgraded.
Need to get 234 MB of archives.
After this operation, 243 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://http.kali.org/kali kali-rolling/main amd64 openjdk-17-jdk-headless amd64 17.0.6+10-1 [230 MB]
Get:2 http://kali.download/kali kali-rolling/main amd64 default-jdk-headless
```

Step 3-> Check if java is installed using `java -version` command.

```
(kali㉿kali)-[~]
$ java -version
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
openjdk version "17.0.6" 2023-01-17
OpenJDK Runtime Environment (build 17.0.6+10-Debian-1)
OpenJDK 64-Bit Server VM (build 17.0.6+10-Debian-1, mixed mode, sharing)
```

Step 4-> Select java version to use with `update-alternatives --config java` command.

```
(kali㉿kali)-[~]
$ update-alternatives --config java
There is 1 choice for the alternative java (providing /usr/bin/java).

  Selection    Path                                Priority      Status
* 0            /usr/lib/jvm/java-17-openjdk-amd64/bin/java  1711      auto mode
  1            /usr/lib/jvm/java-17-openjdk-amd64/bin/java  1711      manual mode

Press <enter> to keep the current choice[*], or type selection number: 0
```

Step 5-> Run `sudo nano /etc/environment` command to edit the environment variables. Append the line `JAVA_HOME="/usr/lib/jvm/java-17-openjdk-amd64/bin/java"`

```
(kali㉿kali)-[~]
$ sudo nano /etc/environment
```

```
GNU nano 7.2                                         /etc/environment *
# START KALI-DEFAULTS CONFIG
# Everything from here and until STOP KALI-DEFAULTS CONFIG
# was installed by the kali-defaults package, and it will
# be removed if ever the kali-defaults package is removed.
# If you want to disable a line, please do NOT remove it,
# as it would be added back when kali-defaults is upgraded.
# Instead, comment the line out, and your change will be
# preserved across upgrades.
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/usr/games
COMMAND_NOT_FOUND_INSTALL_PROMPT=1
POWERSHELL_UPDATECHECK=Off
POWERSHELL_TELEMETRY_OPTOUT=1
DOTNET_CLI_TELEMETRY_OPTOUT=1
JAVA_HOME="/usr/lib/jvm/java-17-openjdk-amd64/bin/java"
# STOP KALI-DEFAULTS CONFIG
```

Step 6-> Now save the environment variables using `source /etc/environment` command. Check if `JAVA_HOME` variable is correctly set using `echo $JAVA_HOME`.

```
(kali㉿kali)-[~]
$ source /etc/environment
```

```
(kali㉿kali)-[~]
$ echo $JAVA_HOME
/usr/lib/jvm/java-17-openjdk-amd64/bin/java
```

Step 7-> Now run `sudo apt-get install apt-transport-https` command.

```
(kali㉿kali)-[~]
$ sudo apt-get install apt-transport-https
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apt-transport-https is already the newest version (2.6.0).
The following packages were automatically installed and are no longer required
:
  cryptsetup-run libexporter-tiny-perl libhttp-server-simple-perl
  liblist-moreutils-perl liblist-moreutils-xs-perl libltnng-ust-ctl4
  libltnng-ust0 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip libquvi-0.9-0.9.3 libquvi-scripts-0.9 lua-bitop
  lua-expat lua-json lua-socket nginx-core python3-singledispatch
  python3-speaklater python3-twisted-bin sphinx-rtd-theme-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 533 not upgraded.
```

Step 8-> Run the command `wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -`.

```
(kali㉿kali)-[~]
$ wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead
(see apt-key(8)).
OK
```

Step 9-> Run the command `echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-7.x.list`.

```
(kali㉿kali)-[~]
$ echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee
/etc/apt/sources.list.d/elastic-7.x.list
deb https://artifacts.elastic.co/packages/7.x/apt stable main
```

Step 10-> Now run `sudo apt-get update` command again.

```
(kali㉿kali)-[~]
$ sudo apt-get update
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable InRelease [13.7 kB]
Hit:2 http://kali.download/kali kali-rolling InRelease
Get:3 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 Packages [109 kB]
Get:4 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 Contents (deb) [3,177 kB]
Fetched 3,300 kB in 5s (637 kB/s)
Reading package lists... Done
W: https://artifacts.elastic.co/packages/7.x/apt/dists/stable/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
```

Step 11-> Now install elasticsearch using `sudo apt-get install elasticsearch`

```
(kali㉿kali)-[~]
$ sudo apt-get install elasticsearch
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
elasticsearch is already the newest version (7.17.9).
The following packages were automatically installed and are no longer required:
  cryptsetup-run libexporter-tiny-perl libhttp-server-simple-perl liblist-moreutils
  libnginx-mod-http-geoip libnginx-mod-http-image-filter libnginx-mod-http-xslt-
  libquvi-0.9-0.9.3 libquvi-scripts-0.9 lua-bitop lua-expat lua-json lua-socket
  sphinx-rtd-theme-common
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 533 not upgraded.
```

Step 12-> Now run sudo nano /etc/elasticsearch/elasticsearch.yml to edit the yml file.

```
[kali㉿kali)-[~] $ sudo nano /etc/elasticsearch/elasticsearch.yml  
[sudo] password for kali: [REDACTED]
```

Step 13-> Add the lines cluster.name: graylog, network.host: your_linux_ip_address, discovery.seed_host: []. Add action.auto_create_index: false, xpack.security.enabled: false.

```
# ----- Cluster -----  
#  
# Use a descriptive name for your cluster:  
#  
cluster.name: graylog  
#  
# ----- Node -----  
  
# ----- Network -----  
#  
# By default Elasticsearch is only accessible on localhost. Set a different  
# address here to expose this node on the network:  
#  
network.host: 10.10.9.166  
  
# ----- Discovery -----  
#  
# Pass an initial list of hosts to perform discovery when this node is started:  
# The default list of hosts is ["127.0.0.1", "[::1]"]  
#  
discovery.seed_hosts: []  
  
GNU nano 7.2 /etc/elasticsearch/elasticsearch.yml *  
#  
#cluster.initial_master_nodes: ["node-1", "node-2"]  
#  
# For more information, consult the discovery and cluster formation module documentation.  
#  
# ----- Various -----  
#  
# Require explicit names when deleting indices:  
#  
#action.destructive_requires_name: true  
action.auto_create_index: false  
xpack.security.enabled: false  
#  
# ----- Security -----
```

Step 14-> Run `sudo systemctl daemon-reload`, `sudo systemctl enable elasticsearch.service`, `sudo systemctl start elasticsearch.service`, `sudo systemctl status elasticsearch.service` commands in the given order.

```
(kali㉿kali)-[~]
$ sudo nano /etc/elasticsearch/elasticsearch.yml

(kali㉿kali)-[~]
$ sudo systemctl daemon-reload

(kali㉿kali)-[~]
$ sudo systemctl enable elasticsearch.service
Synchronizing state of elasticsearch.service with SysV service script with /lib/systemd/sys
Executing: /lib/systemd/systemd-sysv-install enable elasticsearch
Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service → /lib/sy

(kali㉿kali)-[~]
$ sudo systemctl start elasticsearch.service

(kali㉿kali)-[~]
$ sudo systemctl status elasticsearch.service
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; preset: disabled)
   Active: active (running) since Wed 2023-04-05 16:54:44 IST; 5s ago
     Docs: https://www.elastic.co
 Main PID: 10162 (java)
    Tasks: 44 (limit: 3457)
   Memory: 1.4G
      CPU: 1min 40.679s
     CGroup: /system.slice/elasticsearch.service
             └─10162 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networkaddress

Apr 05 16:53:19 kali systemd[1]: Starting elasticsearch.service - Elasticsearch...
Apr 05 16:54:44 kali systemd[1]: Started elasticsearch.service - Elasticsearch.
[lines 1-14/14 (END)]
```

Step 15-> Run the curl command curl -X GET "localhost:9200/?pretty" where localhost is your ip address.

```
(kali㉿kali)-[~]
$ curl -X GET "10.10.9.166:9200/?pretty"
{
  "name" : "kali",
  "cluster_name" : "graylog",
  "cluster_uuid" : "_na_",
  "version" : {
    "number" : "7.17.9",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "ef4822227ee6b9e70e502f0f0daa52435ee634d",
    "build_date" : "2023-01-31T05:34:43.305517834Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

Step 16-> Now run sudo apt-get install uuid-runtime pwgen command.

```
(kali㉿kali)-[~]
$ sudo apt-get install uuid-runtime pwgen
```

Step 17-> Now run sudo apt-get install gnupg command.

```
(kali㉿kali)-[~]
└$ sudo apt-get install gnupg
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gnupg is already the newest version (2.2.40-1.1).
The following packages were automatically installed and are no longer
needed:
  cryptsetup-run libexporter-tiny-perl libhttp-server-simple-perl
  liblttng-ust0 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-stream libnginx-mod-stream-geoip libquvi-0.9-0.9.3
  nginx-core python3-singledispatch python3-speaklater python3-twink
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 533 not upgraded.
```

Step 18-> Now install gnupg using sudo apt-get install gnupg command.

```
(kali㉿kali)-[~]
└$ sudo apt-get install gnupg
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
gnupg is already the newest version (2.2.40-1.1).
The following packages were automatically installed and are no longer
needed:
  cryptsetup-run libexporter-tiny-perl libhttp-server-simple-perl
  liblttng-ust0 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-stream libnginx-mod-stream-geoip libquvi-0.9-0.9.3
  nginx-core python3-singledispatch python3-speaklater python3-twink
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 533 not upgraded.
```

Step 19-> Now login to root user with su - command.

```
(kali㉿kali)-[~]
$ su -
Password:
(Message from Kali developers)

We have kept /usr/bin/python pointing to Python 2 for backwards
compatibility. Learn how to change this and avoid this message:
⇒ https://www.kali.org/docs/general-use/python3-transition/
(Run: "touch ~/.hushlogin" to hide this message)
(root💀kali)-[~]
```

Step 20-> Now run the command wget -qO - https://www.mongodb.org/static/pgp/server-6.0.asc | sudo apt-key add -

```
(root💀kali)-[~]
# wget -qO - https://www.mongodb.org/static/pgp/server-6.0.asc | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
```

Step 21-> Next run command echo "deb [arch=amd64,arm64] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/6.0 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-6.0.list. Exit from root access using exit command.

```
[root💀kali]-[~]
# echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu ja
mmy/mongodb-org/6.0 multiverse" | sudo tee /etc/apt/sources.list.d/mongod
b-org-6.0.list
deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu jammy/mongod
b-org/6.0 multiverse
#
[root💀kali]-[~]
# exit
```

Step 22-> Run sudo apt-get update again

```
(kali㉿kali)-[~]
$ sudo apt-get update
Hit:1 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Ign:2 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0 InRelease
Get:3 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0 Release [3,094 B]
Get:5 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0 Release.g
pg [801 B]
Hit:4 http://kali.download/kali kali-rolling InRelease
Get:6 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0/multivers
e amd64 Packages [14.3 kB]
Get:7 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0/multivers
e arm64 Packages [12.6 kB]
Fetched 30.8 kB in 2s (13.7 kB/s)
Reading package lists... Done
W: https://artifacts.elastic.co/packages/7.x/apt/dists/stable/InRelease:
Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see t
he DEPRECATION section in apt-key(8) for details.
W: https://repo.mongodb.org/apt/ubuntu/dists/jammy/mongodb-org/6.0/Releas
e.gpg: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg)
, see the DEPRECATION section in apt-key(8) for details.
```

Step 23-> Now install mongodb using sudo apt-get install mongodb-org command

```
└─(kali㉿kali)-[~]
└$ sudo apt-get install mongodb-org
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  cryptsetup-run libexporter-tiny-perl libhttp-server-simple-perl
  liblist-moreutils-perl liblist-moreutils-xs-perl liblttng-ust-ctl4
  liblttng-ust0 libnginx-mod-http-geoip libnginx-mod-http-image-filter
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip libquvi-0.9-0.9.3 libquvi-scripts-0.9
  lua-bitop lua-expat lua-json lua-socket nginx-core
  python3-singledispatch python3-speaklater python3-twisted-bin
  sphinx-rtd-theme-common
Use 'sudo apt autoremove' to remove them.
```

Step 24-> Enter root user using su - again. Run command wget -qO-'http://keyserver.ubuntu.com/pks/lookup?op=get&search=0xf5679a222c647c87527c2f8cb00a0bd1e2c63c11' | sudo apt-key add -.

```
└─(kali㉿kali)-[~]
└$ su -
Password:
(Message from Kali developers)

We have kept /usr/bin/python pointing to Python 2 for backwards
compatibility. Learn how to change this and avoid this message:
⇒ https://www.kali.org/docs/general-use/python3-transition/

(Run: "touch ~/.hushlogin" to hide this message)
└─(root💀kali㉿kali)-[~]
└# wget -qO-'http://keyserver.ubuntu.com/pks/lookup?op=get&search=0xf5679a222c647c87527c2f8cb00a0bd1e2c63c11' | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
```

Step 25-> Exit root access using exit command

```
└─(root💀kali㉿kali)-[~]
└# exit
```

Step 26-> Now run commands `sudo systemctl daemon-reload` `sudo systemctl enable mongod.service` `sudo systemctl start mongod.service` `sudo systemctl --type=service --state=active | grep mongod` `sudo systemctl status mongod.service`.

```
(kali㉿kali)-[~]
└─$ sudo systemctl daemon-reload

(kali㉿kali)-[~]
└─$ sudo systemctl enable mongod.service
Created symlink /etc/systemd/system/multi-user.target.wants/mon

(kali㉿kali)-[~]
└─$ sudo systemctl start mongod.service

(kali㉿kali)-[~]
└─$ sudo systemctl --type=service --state=active | grep mongod
mongod.service                                loaded active running MongoDB
mongod.service - MongoDB Database Server
● mongod.service - MongoDB Database Server
  Loaded: loaded (/lib/systemd/system/mongod.service; enabled;
  Active: active (running) since Wed 2023-04-05 17:10:09 IST
    Docs: https://docs.mongodb.org/manual/
   Main PID: 16846 (mongod)
     Memory: 72.2M
        CPU: 1.710s
      CGroup: /system.slice/mongod.service
              └─16846 /usr/bin/mongod --config /etc/mongod.conf

Apr 05 17:10:09 kali systemd[1]: Started mongod.service - MongoDB
lines 1-11/11 (END)...skipping...
● mongod.service - MongoDB Database Server
  Loaded: loaded (/lib/systemd/system/mongod.service; enabled;
```

Step 27-> Change directory to Downloads using cd Downloads .Now download graylog Debian package using sudo wget https://packages.graylog2.org/repo/packages/graylog-5.0-repository_latest.deb command

```
(kali㉿kali)-[~]
$ cd Downloads

(kali㉿kali)-[~/Downloads]
$ sudo wget https://packages.graylog2.org/repo/packages/graylog-5.0-repository_latest.deb
--2023-04-05 17:17:40-- https://packages.graylog2.org/repo/packages/graylog-5.0-repository_
Resolving packages.graylog2.org (packages.graylog2.org)... 104.21.88.209, 172.67.153.95, 260
Connecting to packages.graylog2.org (packages.graylog2.org)|104.21.88.209|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://graylog-package-repository.s3.eu-west-1.amazonaws.com/packages/graylog-5.0-
-HMAC-SHA256&X-Amz-Date=20230405T114740Z&X-Amz-SignedHeaders=host&X-Amz-Expires=600&X-Amz-Cr
eu-west-1%2Fs3%2Faws4_request&X-Amz-Signature=7f312877d1574aece8fdd73ca65082b2d4c06c8316537
--2023-04-05 17:17:42-- https://graylog-package-repository.s3.eu-west-1.amazonaws.com/packa
-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20230405T114740Z&X-Amz-SignedHeaders=host&X-Amz-Expri
A%2F20230405%2Feu-west-1%2Fs3%2Faws4_request&X-Amz-Signature=7f312877d1574aece8fdd73ca65082
Resolving graylog-package-repository.s3.eu-west-1.amazonaws.com (graylog-package-repository.
52.92.0.162, 52.218.101.200, ...
Connecting to graylog-package-repository.s3.eu-west-1.amazonaws.com (graylog-package-reposit
|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2084 (2.0K) [application/x-debian-package]
Saving to: 'graylog-5.0-repository_latest.deb'

graylog-5.0-repository_latest.deb 100%[=====
2023-04-05 17:17:42 (34.4 MB/s) - 'graylog-5.0-repository_latest.deb' saved [2084/2084]
```

Step 28-> Install graylog server using command sudo dpkg -i graylog-5.0-repository_latest.deb.
Change directory once more using cd command.

```
(kali㉿kali)-[~/Downloads]
$ sudo dpkg -i graylog-5.0-repository_latest.deb
Selecting previously unselected package graylog-5.0-repository.
(Reading database ... 354407 files and directories currently installed.)
Preparing to unpack graylog-5.0-repository_latest.deb ...
Unpacking graylog-5.0-repository (1-2) ...
Setting up graylog-5.0-repository (1-2) ...

(kali㉿kali)-[~/Downloads]
$ cd
```

Step 29-> Run sudo apt-get update once more.

```
[kali㉿kali)-[~]
$ sudo apt-get update
Hit:1 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Ign:2 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0 InRelease
Hit:4 https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/6.0 Release
Hit:3 http://kali.download/kali kali-rolling InRelease
Get:5 https://packages.graylog2.org/repo/debian stable InRelease [54.5 kB]
Get:7 https://packages.graylog2.org/repo/debian stable/5.0 amd64 Packages
Fetched 60.9 kB in 4s (15.6 kB/s)
Reading package lists... Done
W: https://artifacts.elastic.co/packages/7.x/apt/dists/stable/InRelease:1
gpg: https://artifacts.elastic.co/packages/7.x/apt/dists/stable/InRelease:1
gpg:   ... trust issues
gpg: Please see apt-key(8) for details.
W: https://repo.mongodb.org/apt/ubuntu/dists/jammy/mongodb-org/6.0/Releas
rusted.gpg:1
gpg: https://repo.mongodb.org/apt/ubuntu/dists/jammy/mongodb-org/6.0/Releas
rusted.gpg:1
gpg:   ... trust issues
gpg: Please see apt-key(8) for details.
```

Step 30-> Now install graylog-server using sudo apt-get install graylog-server command.

```
[kali㉿kali)-[~]
$ sudo apt-get install graylog-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer
needed:
  cryptsetup-run libexporter-tiny-perl libhttp-server-simple-perl lib
  liblttng-ust0 libnginx-mod-http-geoip libnginx-mod-http-image-filte
  libnginx-mod-stream libnginx-mod-stream-geoip libquvi-0.9-0.9.3 lib
  nginx-core python3-singledispatch python3-speaklater python3-twiste
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  util-linux util-linux-extra
Suggested packages:
  util-linux-locales
The following NEW packages will be installed:
  graylog-server util-linux-extra
The following packages will be upgraded:
  util-linux
1 upgraded, 2 newly installed, 0 to remove and 532 not upgraded.
Need to get 300 MB of archives.
After this operation, 420 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Step 31-> Run the commands `sudo systemctl enable graylog-server.service` `sudo systemctl start graylog-server.service` `sudo systemctl status graylog-server.service`.

```
(kali㉿kali)-[~]
$ sudo systemctl enable graylog-server.service
Synchronizing state of graylog-server.service with SysV service script with /lib/s
Executing: /lib/systemd/systemd-sysv-install enable graylog-server
Created symlink /etc/systemd/system/multi-user.target.wants/graylog-server.service
```

```
(kali㉿kali)-[~]
$ sudo systemctl start graylog-server.service
```

```
(kali㉿kali)-[~]
$ sudo systemctl status graylog-server.service
● graylog-server.service - Graylog server
  Loaded: loaded (/lib/systemd/system/graylog-server.service; enabled; pres
  Active: activating (auto-restart) (Result: exit-code) since Thu 2023-04-20 14:50:17
    Docs: http://docs.graylog.org/
  Process: 102689 ExecStart=/usr/share/graylog-server/bin/graylog-server (>
  Main PID: 102689 (code=exited, status=1/FAILURE)
    CPU: 3.240s

Apr 20 14:50:17 kali systemd[1]: graylog-server.service: Main process exited>
Apr 20 14:50:17 kali systemd[1]: graylog-server.service: Failed with result >
Apr 20 14:50:17 kali systemd[1]: graylog-server.service: Consumed 3.240s CPU>
log file: █
```

```
(kali㉿kali)-[~]
$ sudo systemctl status graylog-server.service
● graylog-server.service - Graylog server
  Loaded: loaded (/lib/systemd/system/graylog-server.service; enabled; preset: disabled)
  Active: active (running) since Wed 2023-04-05 17:35:25 IST; 1min 15s ago
    Docs: http://docs.graylog.org/
  Main PID: 24447 (graylog-server)
    Tasks: 20 (limit: 3457)
   Memory: 118.0M
      CPU: 47.987s
     CGroup: /system.slice/graylog-server.service
             └─24447 /bin/sh /usr/share/graylog-server/bin/graylog-server
                  ├─24449 /usr/share/graylog-server/jvm/bin/java -Xms1g -Xmx1g -server -XX:+UseG1
```

Step 32-> Now edit the server.conf file of graylog server using sudo nano `/etc/graylog/server/server.conf` command.

```
(kali㉿kali)-[~]
$ sudo nano /etc/graylog/server/server.conf
```

Step 33-> Now open another terminal and run echo -n "Enter Password: " && head -1 </dev/stdin | tr -d '\n' | sha256sum | cut -d" " -f1 to generate root password SHA2 hash and pwgen -N 1 -s 96 to generate password secret. Save both in a text file.

```
(kali㉿kali)-[~]
$ echo -n "Enter Password: " && head -1 </dev/stdin | tr -d '\n' | sha256sum | cut -d" " -f1
Enter Password: kali
fc5669b52ce4e283ad1d5d182de88ff9faec6672bace84ac2ce4c083f54fe2bc

(kali㉿kali)-[~]
$

(kali㉿kali)-[~]
$ pwgen -N 1 -s 96
p6aH1oWCKwAsIbHRGhoZmb3atG2mGfhUUFRh6gszvqFDqaSp1kW3p3IJ5LUzrnBDfijpirblkBFsgibvm5xbur0NSVWngbkG
```

Step 34-> Now run sudo nano /etc/graylog/server/server.conf. Use the keys you generated and paste sha256 hash after root_password_sha2 and password_secret for pwgen key accordingly. Uncomment line http_bind_address and add http_bind_address = localhost:9000 (in place of localhost replace with your system ip). Uncomment this line elasticsearch_hosts = http://localhost:9200/ (in place of localhost replace with your system ip) comment this line after above line by pressing enter and then type "#" <http://user:password@192.168.4.84:19200>. Save the file

```
GNU nano 7.2                               /etc/graylog/server/server.conf
# HTTP settings
#####
#### HTTP bind address
#
# The network interface used by the Graylog HTTP interface.
#
# This network interface must be accessible by all Graylog nodes
# using the Graylog web interface.
#
# If the port is omitted, Graylog will use port 9000 by default
#
# Default: 127.0.0.1:9000
http_bind_address = 10.10.9.166:9000
#http_bind_address = [2001:db8::1]:9000
#### HTTP publish URI
```

```
GNU nano 7.2          /etc/graylog/server/server.conf
#trusted_proxies = 127.0.0.1/32, 0:0:0:0:0:0:0:1/128

# List of Elasticsearch hosts Graylog should connect to.
# Need to be specified as a comma-separated list of valid
# If one or more of your elasticsearch hosts require authen
# requires authentication.
#
# Default: http://127.0.0.1:9200
elasticsearch_hosts = http://10.10.9.166:9200
#http://user:password@node2:19200

# Maximum number of attempts to connect to elasticsearch o
#
# Default: 0, retry indefinitely with the given delay until
#elasticsearch_version_probe_attempts = 5
```

```
# Changing this value after installation will render all user sessions and encrypted values in the database invali
password_secret =J0KsT9LwA7ao3mfAvaveMDE1Zbz3h8LMTn4Cdierjcqj0WkQd8crD0DtCC2yua9PxGWiCiCDA1pLbDQb7F7m37AQIFKgiIuW
```

```
# and put the resulting hash value into the following line
root_password_sha2 =fc5669b52ce4e283ad1d5d182de88ff9faec6672bace84ac2ce4c083f54fe2bc
```

Step 35-> Run the commands `sudo systemctl restart graylog-server.service`, `sudo systemctl --type=service --state=active | grep graylog`, `sudo systemctl restart graylog-server.service`, `sudo systemctl status graylog-server.service` commands in same order.

```
(kali㉿kali)-[~]
└─$ sudo systemctl --type=service --state=active | grep graylog
graylog-server.service          loaded active running Graylog server

(kali㉿kali)-[~]
└─$ sudo systemctl restart graylog-server.service

(kali㉿kali)-[~]
└─$ sudo systemctl status graylog-server.service
● graylog-server.service - Graylog server
  Loaded: loaded (/lib/systemd/system/graylog-s
  Active: active (running) since Thu 2023-04-06
    Docs: http://docs.graylog.org/
  Main PID: 12013 (graylog-server)
    Tasks: 23 (limit: 2251)
   Memory: 188.3M
      CPU: 5.305s
     CGroup: /system.slice/graylog-server.service
             └─12013 /bin/sh /usr/share/graylog-se
                 ├─12015 /usr/share/graylog-server/jvm

Apr 06 15:45:45 kali systemd[1]: Started graylog-s
```

Step 36-> Now run <http://localhost:9000/dashboard> and check if graylog gui is shown as below in a browser.

The screenshot shows the Graylog 5.0.5 dashboard interface. At the top, there's a navigation bar with links for Search, Streams, Alerts, Dashboards (which is underlined), Enterprise, Security, and System. A red notification badge with the number '2' is visible on the System icon. On the right side of the header, there are icons for VPN, a user profile, and other system status indicators. Below the header, the main content area has a title 'Dashboards' and a sub-section titled 'Sources'. It says, 'This is a list of all sources that sent messages to Graylog. You can narrow the timerange by zooming in on the message histogram, or you can increase the time range by specifying a broader one in the controls at the top. You can also specify filters to limit the results you are seeing. You can also add additional widgets to this dashboard, or adapt the appearance of existing widgets to suit your needs.' A note at the bottom indicates it was last saved at 2019-11-22 10:54:50. The footer of the page displays the text 'Graylog 5.0.5+d61a926 on kali (Eclipse Adoptium 17.0.6 on Linux 6.1.0-kali7-amd64)'.