

# Snort Traffic Analysis

The Snort Traffic Analysis lab provides practical experience in configuring Snort, analyzing network traffic, and creating custom rules for enhanced security.

I successfully downloaded the lab6 file and used the command `xxd` `~/Downloads/paper.pdf | head -n 5` to check its headers. The headers are displayed in the provided screenshot.

```
(kali㉿kali)-[~]  
$ xxd ~/Downloads/lab6.pdf | head -n 5  
00000000: 2550 4446 2d31 2e33 0a25 c4e5 f2e5 eba7  %PDF-1.3.%.....  
00000010: f3a0 d0c4 c60a 3320 3020 6f62 6a0a 3c3c  ....3 0 obj.<<  
00000020: 202f 4669 6c74 6572 202f 466c 6174 6544  /Filter /FlateD  
00000030: 6563 6f64 6520 2f4c 656e 6774 6820 3537  ecode /Length 57  
00000040: 3030 203e 3e0a 7374 7265 616d 0a78 01bd  00 >>.stream.x..
```

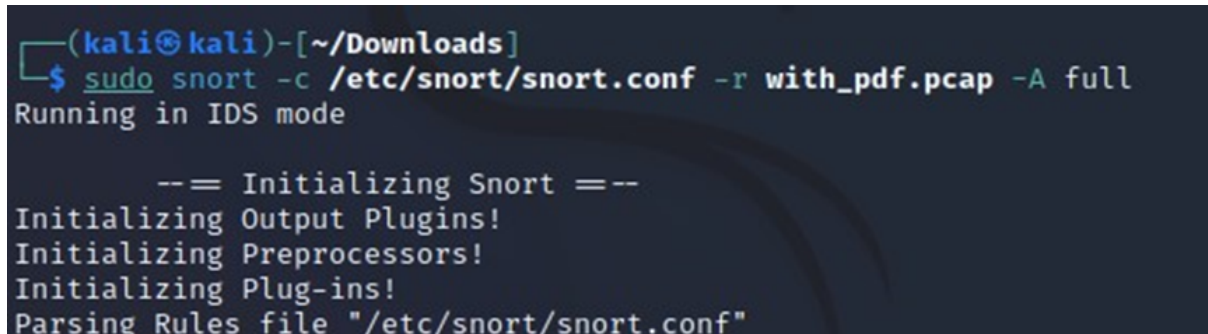
To ensure a clean slate, I deleted all the Snort logs using the given command.

```
(kali㉿kali)-[~]  
$ sudo rm -r /var/log/snort/*  
[sudo] password for kali:  
  
(kali㉿kali)-[~]  
$
```

I then extracted the contents of the with\_pdf.zip file into the Downloads directory. The screenshot demonstrates the extraction process.

```
(kali㉿kali)-[~/Downloads]  
$ unzip with_pdf.zip  
Archive:  with_pdf.zip  
inflating: with_pdf.pcap
```

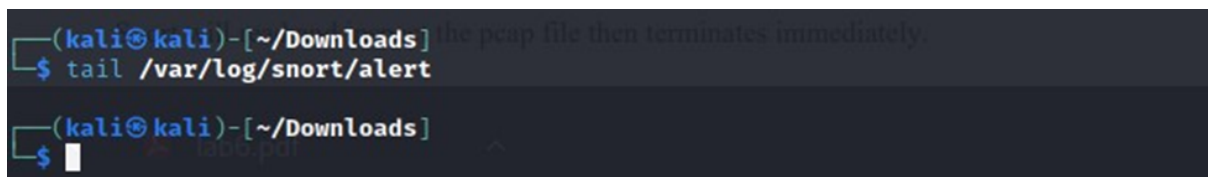
Running Snort with the options `-r` to read the pcap file and `-A full` to save alerts in `/var/log/`, I used the command `sudo snort -c /etc/snort/snort.conf -r with_pdf.pcap -A full`. The screenshot confirms the execution of this command.

A terminal window with a dark background. The prompt is `(kali㉿kali)-[~/Downloads]`. The user enters `$ sudo snort -c /etc/snort/snort.conf -r with_pdf.pcap -A full`. The output shows "Running in IDS mode" followed by a separator `--= Initializing Snort ==--` and several initialization messages: "Initializing Output Plugins!", "Initializing Preprocessors!", "Initializing Plug-ins!", and "Parsing Rules file "/etc/snort/snort.conf"".

```
(kali㉿kali)-[~/Downloads]
$ sudo snort -c /etc/snort/snort.conf -r with_pdf.pcap -A full
Running in IDS mode

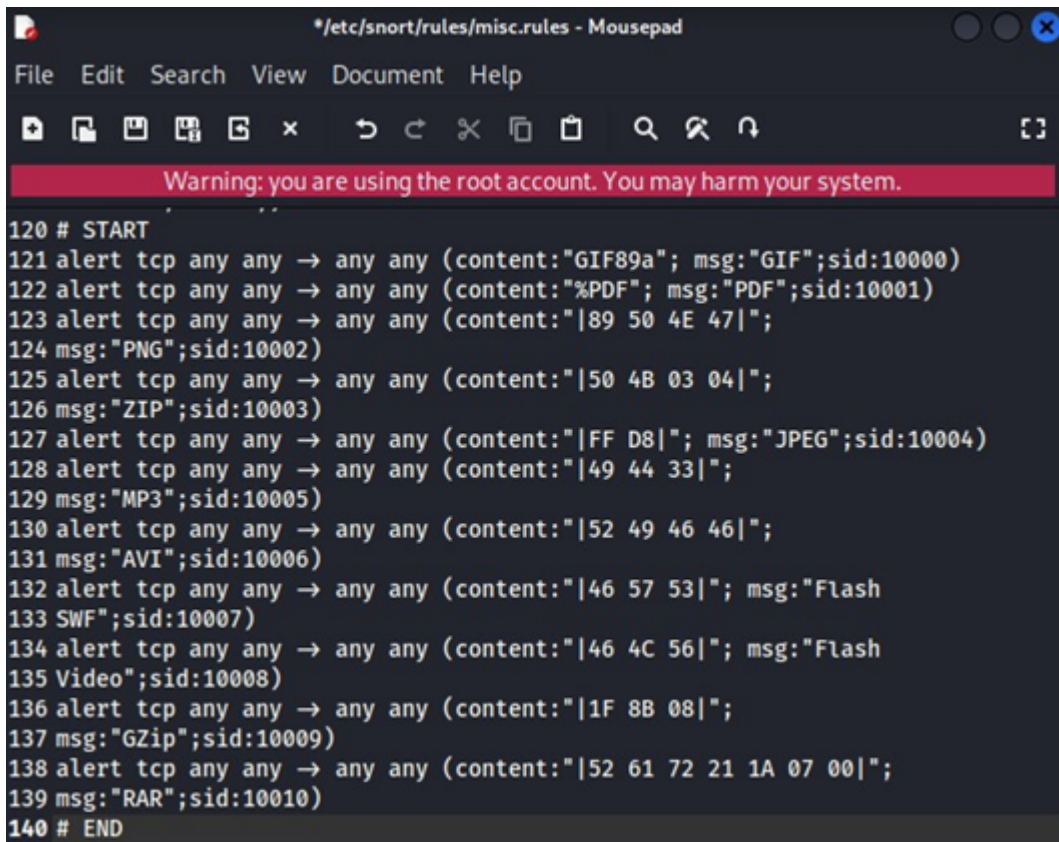
--= Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
```

Afterwards, I checked the alert files, but no alerts were found, as indicated in the corresponding screenshot.

A terminal window with a dark background. The prompt is `(kali㉿kali)-[~/Downloads]`. The user enters `$ tail /var/log/snort/alert`. The output is empty.

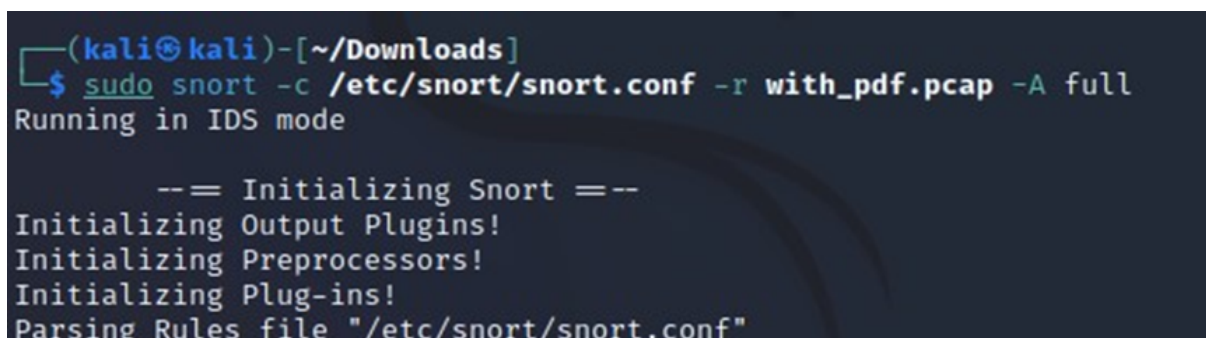
```
(kali㉿kali)-[~/Downloads]
$ tail /var/log/snort/alert
```

Moving on, I proceeded to write the rules in the specified file. The screenshot showcases the rule-writing process.



```
* /etc/snort/rules/misc.rules - Mousepad
File Edit Search View Document Help
Warning: you are using the root account. You may harm your system.
120 # START
121 alert tcp any any -> any any (content:"GIF89a"; msg:"GIF";sid:10000)
122 alert tcp any any -> any any (content:"%PDF"; msg:"PDF";sid:10001)
123 alert tcp any any -> any any (content:"|89 50 4E 47|";
124 msg:"PNG";sid:10002)
125 alert tcp any any -> any any (content:"|50 4B 03 04|";
126 msg:"ZIP";sid:10003)
127 alert tcp any any -> any any (content:"|FF D8|"; msg:"JPEG";sid:10004)
128 alert tcp any any -> any any (content:"|49 44 33|";
129 msg:"MP3";sid:10005)
130 alert tcp any any -> any any (content:"|52 49 46 46|";
131 msg:"AVI";sid:10006)
132 alert tcp any any -> any any (content:"|46 57 53|"; msg:"Flash
133 SWF";sid:10007)
134 alert tcp any any -> any any (content:"|46 4C 56|"; msg:"Flash
135 Video";sid:10008)
136 alert tcp any any -> any any (content:"|1F 8B 08|";
137 msg:"GZip";sid:10009)
138 alert tcp any any -> any any (content:"|52 61 72 21 1A 07 00|";
139 msg:"RAR";sid:10010)
140 # END
```

Running the Snort command again, the alerts were detected this time, as depicted in the screenshot.



```
(kali@kali)-[~/Downloads]
$ sudo snort -c /etc/snort/snort.conf -r with_pdf.pcap -A full
Running in IDS mode

--= Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
```

```

(kali㉿kali)-[~/Downloads]
$ cat /var/log/snort/alert
[**] [1:10009:0] GZip [**]
[Priority: 0]
01/05-14:16:41.637650 173.194.34.88:80 → 192.168.47.171:2574
TCP TTL:128 TOS:0x0 ID:56984 IpLen:20 DgmLen:305
***AP*** Seq: 0x7756221B Ack: 0x96EC4B61 Win: 0xFAF0 TcpLen: 20

[**] [1:10001:0] PDF [**]
[Priority: 0]
01/05-14:16:42.194095 98.139.134.174:80 → 192.168.47.171:2575
TCP TTL:128 TOS:0x0 ID:56989 IpLen:20 DgmLen:1500
***AP*** Seq: 0x71B91DD6 Ack: 0xB4068CF7 Win: 0xFAF0 TcpLen: 20

[**] [1:10000:0] GIF [**]
[Priority: 0]
01/05-14:16:42.645899 98.139.134.174:80 → 192.168.47.171:2577
TCP TTL:128 TOS:0x0 ID:56999 IpLen:20 DgmLen:301
***AP*** Seq: 0x305B3584 Ack: 0xFBDCF123 Win: 0xFAF0 TcpLen: 20

```

I chose two pcap files from the website [malware-traffic-analysis.net](http://malware-traffic-analysis.net) and performed the Snort command on each file. Screenshots are provided for both cases, showing the successful detection of alerts based on the previously set rules.

Case 1:

```

(kali㉿kali)-[~/Downloads]
$ ls
2022-01-07-traffic-analysis-exercise.pcap  lab6.pdf  with_pdf.pcap
analysis(1).pcap                         paper.pdf  with_pdf.zip

(kali㉿kali)-[~/Downloads]
$ sudo snort -c /etc/snort/snort.conf -r 2022-01-07-traffic-analysis-exercise.pcap -A full

```

```

(kali㉿kali)-[~/Downloads]
$ tail /var/log/snort/alert
01/07-11:16:09.964527 192.168.1.216:49761 → 192.168.1.2:139
TCP TTL:128 TOS:0x0 ID:52736 IpLen:20 DgmLen:136 DF
***AP*** Seq: 0x9F1155AD Ack: 0xC8D6F3C6 Win: 0x2010 TcpLen: 20

[**] [1:538:15] NETBIOS SMB IPC$ unicode share access [**]
[Classification: Generic Protocol Command Decode] [Priority: 3]
01/07-11:16:39.981840 192.168.1.216:49763 → 192.168.1.2:139
TCP TTL:128 TOS:0x0 ID:52751 IpLen:20 DgmLen:136 DF
***AP*** Seq: 0x15E4B123 Ack: 0xDD225076 Win: 0x2010 TcpLen: 20

```



## Case 2:

```
(kali㉿kali)-[~/Downloads]
$ ls
2022-01-07-traffic-analysis-exercise.pcap  'analysis(1).pcap'  paper.pdf  with_pdf.zip
2022-02-23-traffic-analysis-exercise.pcap  lab6.pdf           with_pdf.pcap

(kali㉿kali)-[~/Downloads]
$ sudo snort -c /etc/snort/snort.conf -r 2022-02-23-traffic-analysis-exercise.pcap -A full
```

```
(kali㉿kali)-[~/Downloads]
$ tail /var/log/snort/alert
02/23-14:07:05.139489 172.16.0.149:49852 → 116.254.112.253:25
TCP TTL:128 TOS:0x0 ID:56240 IpLen:20 DgmLen:1500 DF
***A*** Seq: 0xBCFC6E14 Ack: 0xB6EF2208 Win: 0xF95D TcpLen: 20

[**] [1:10005:0] MP3 [**]
[Priority: 0]
02/23-14:07:05.139489 172.16.0.149:49852 → 116.254.112.253:25
TCP TTL:128 TOS:0x0 ID:56240 IpLen:20 DgmLen:1500 DF
***A*** Seq: 0xBCFC6E14 Ack: 0xB6EF2208 Win: 0xF95D TcpLen: 20
```

## Conclusion

In conclusion, I successfully performed various tasks in the Snort Traffic Analysis lab. This included checking the headers of a PDF file and creating an alert to detect it based on its header signature. By running Snort with the provided pcap files, I detected and logged alerts according to the predefined rules. This hands-on experience with Snort enhanced my understanding of network security, intrusion detection, and the importance of file headers in identifying specific file formats. Overall, the lab provided practical skills in utilizing Snort for traffic analysis and improving network security.