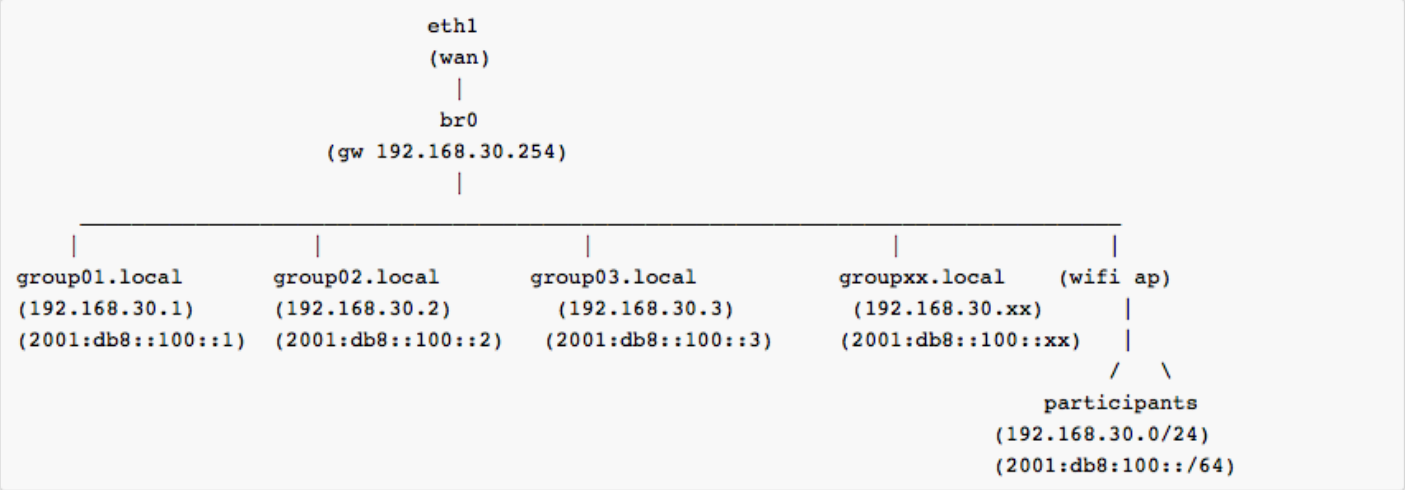
Module: Deploy Security Onion

Objective: As part of this hands-on module, you will be installing and configuring Security Onion (Network Monitoring System).

Prerequisites: Knowledge of Ubuntu, IDS, Packet analysis and security concepts.

Topology

The following will be the topology used for this lab. Note that the IP addresses are examples only. When working on the lab, use the actual IP addresses as indicated by the instructors. For the purpose of this guide, the IP address of 192.168.30.X or 2001:db8:100::X will refer to your Virtual Machine (VM).



Lab Notes

Depending on the workshop, you may be:

1. Given a Virtual Machine (VM) that is already configured with Security Onion. Start the lab at Part 3.
2. Or asked to connect to a container over the network. Refer to Part 1 Setup X11 Forwarding.

- In this guide the interface name is `eth0`. Depending on the version of Ubuntu the interface name may be `enp0s3` or something different. Where `eth0` is used in this guide replace it with your interface name.
- Container details
  - Ubuntu 16.04 LTS/LXC

- Hostname = groupXX.apnictraining.net
- Domain name = apnictraining.net
- IPv4 Address = 192.168.30.xx
- IPv6 Address = 2001:db8:100::xx
- xx = group ID as allocated by the instructor

Lab Exercise - Security Onion setup and configuration

Part 1. Setup X11 Forwarding for client

Windows 10

1. On the Windows machine. Download putty and install

```
https://ninite.com/putty/
or
https://www.putty.org
```

2. On the Windows machine. Download Xming and install

```
https://sourceforge.net/projects/xming/
```

3. Open Xming and allow firewall rules if prompted.

4. Connect to the container

```
putty -ssh -X apnic@192.168.30.xx
```

-X = X11 forwarding .xx = group number

5. Confirm X11Forwarding is set to `Yes` in the /etc/ssh/sshd_config file.

```
cat /etc/ssh/sshd_config | grep X11
```

6. Confirm X11 forwarding is working.

```
firefox https://www.apnic.net
or
chromium-browser https://www.apnic.net
```

MacOS

1. On the MacOS machine. Download XQuartz and install

```
https://support.apple.com/en-au/HT201341
```

2. Connect to the container

```
ssh -v -X apnic@192.168.30.xx
```

-X = X11 forwarding -v = verbose .xx = group number

3. Confirm X11Forwarding is set to `Yes` in the /etc/ssh/sshd_config file.

```
cat /etc/ssh/sshd_config | grep X11
```

4. Confirm display settings.

```
echo $DISPLAY
```

5. If display output is blank. Type the following:

```
export DISPLAY="localhost:10.0"
```

6. Confirm X11 forwarding is working.

```
firefox https://www.apnic.net
or
chromium-browser https://www.apnic.net
```

Ubuntu

1. Connect to the container

```
ssh -v -X apnic@192.168.30.xx
```

-X = X11 forwarding -v = verbose .xx = group number

2. Confirm X11Forwarding is set to `Yes` in the /etc/ssh/sshd_config file.

```
cat /etc/ssh/sshd_config | grep X11
```

3. Confirm X11 forwarding is working.

```
firefox https://www.apnic.net
or
chromium-browser https://www.apnic.net
```

Part 2. Installation of Security Onion

1. To install Security Onion via a package manager type the following commands:

```
echo "debconf debconf/frontend select noninteractive" | sudo debconf-set-selec
tions
sudo apt-get update
sudo apt-get -y install software-properties-common
sudo add-apt-repository -y ppa:securityonion/stable
sudo apt-get update
sudo apt-get -y install securityonion-all syslog-ng-core
```

   NOTE: This should have already been completed on the container or Virtual Machine (VM).

2. Run setup wizard, type the following:

```
sudo sosetup
```

   NOTE: Do not complete these steps on the Virtual Machine (VM) as it has already been done. Instead check the status of the Security Onion by typing `sudo sostatus`

3. Click on `Yes, Continue`
4. Click on `Yes, configure /etc/network/interfaces`
5. Select `static` and click on `Ok`
6. The next few screens will ask for details about the network:

```
IP Address = 192.168.30.XX (XX is group number)
Subnet Mask = 255.255.255.0
Gateway IP = 192.168.30.254 (Confirm with Instructor)
DNS IP = 1.1.1.1 192.168.30.249(Confirm with Instructor)
Domain name = apnictraining.net
```

7. Click on `Yes, make changes!`
8. Click on `Yes, reboot!`

   NOTE: May need to ask the instructor to stop and start the container

9. After the reboot, reconnect to the container.

```
ssh -X apnic@192.168.30.XX
or
putty -ssh -X apnic@192.168.30.XX
```

-X = X11 forwarding -v = verbose .xx = group number

10. restart the setup wizard. Type the following:

```
sudo sosetup
```

11. Click on `Yes, Continue`

12. Click on `Yes, skip network configuration!`

13. Evaluation mode will install all the tools onto the one virtual machine or container. Click on `Evaluation Mode`, then click on `OK`.

14. Create a user account that will be used to log into Kibana, Squert and Sguil.

```
username: apnic
password: training
```

NOTE: In a production environment, this should be a different account to what is used to log into Ubuntu

15. Click on `Yes, proceed with the changes!`. This will take some time to complete as it downloads and configures docker.

16. Once installed, check the status of the Security Onion services by typing the following:

```
sudo sostatus
```

17. Test you can open Squert in a browser.

```
chromium-browser https://localhost/squert
or
firefox https://localhost/squert
```

Part 3. Import Packet Capture File

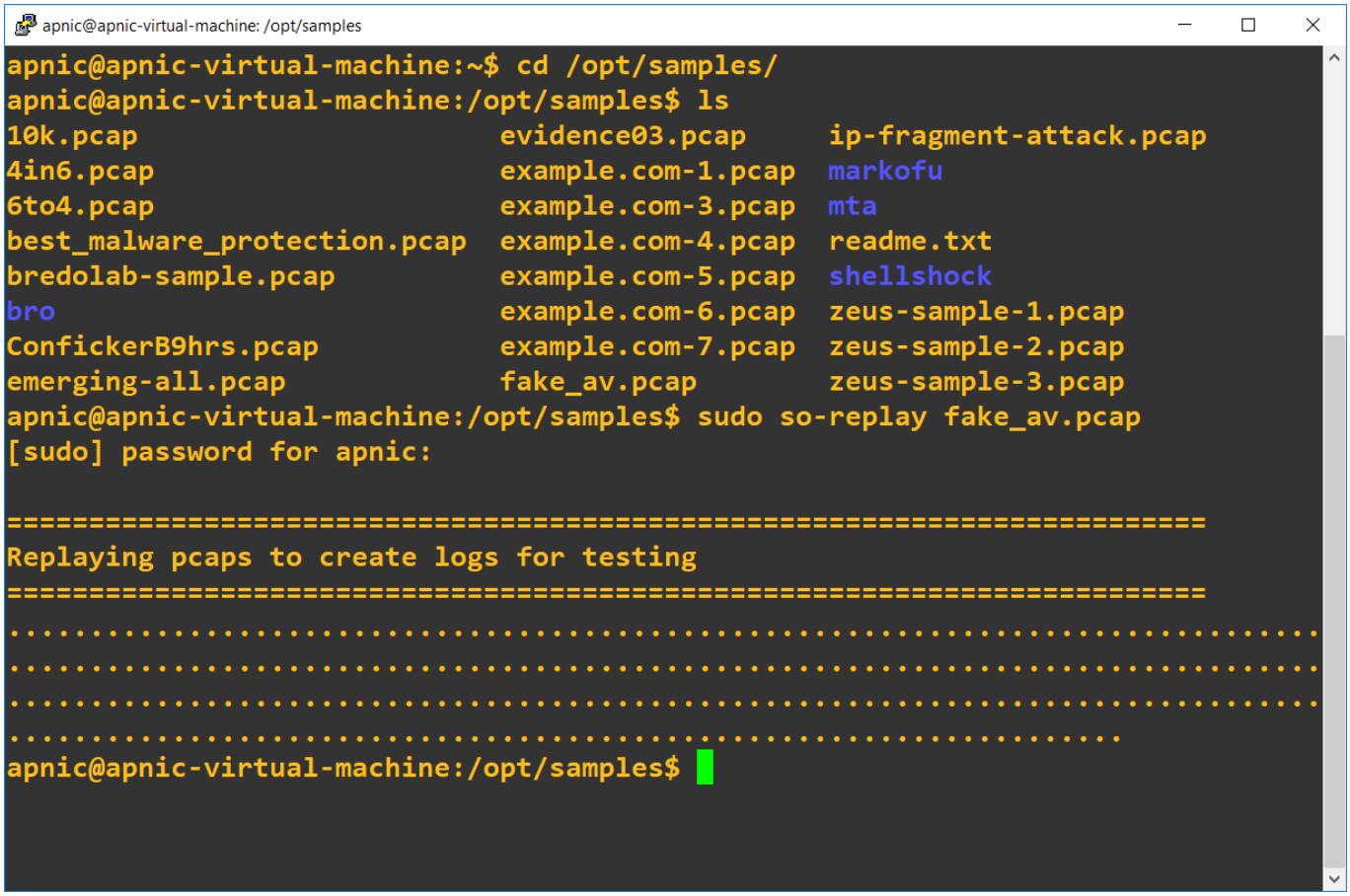
1. To view the list of sample packet captures that are available:

```
cd /opt/samples
ls
```

2. Import the fake_av.pcap file:

```
sudo so-replay fake_av.pcap
```

This will import the pcap file as new traffic with the current date and time.



After importing the packet capture file, we will have a look at the alerts that were generated by SNORT, by utilising a tool called SQUERT.

1. To see a summary of the fake_av.pcap file, type the following:

```
capinfos fake_av.pcap
```

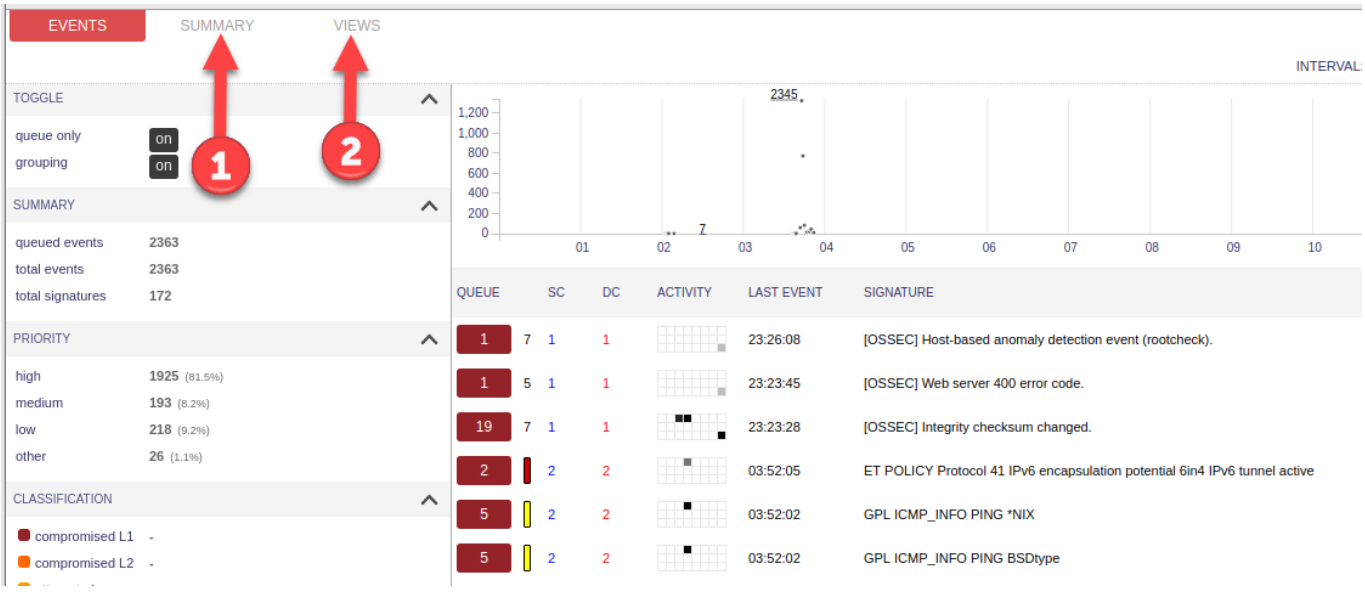
```
apnic@group29:/opt/samples$ capinfos fake_av.pcap
File name:           fake_av.pcap
File type:           Wireshark/tcpdump/... - pcap
File encapsulation:  Linux cooked-mode capture
File timestamp precision:  microseconds (6)
Packet size limit:   file hdr: 65535 bytes
Number of packets:   1,771
File size:           806 kB
Data size:           777 kB
Capture duration:    1063.009039 seconds
First packet time:   2011-04-03 02:02:03.865392
Last packet time:    2011-04-03 02:19:46.874431
Data byte rate:      731 bytes/s
Data bit rate:       5,854 bits/s
Average packet size: 439.27 bytes
Average packet rate: 1 packets/s
SHA256:              ee81ad97420c2395a4c6ec168d26c2562d5cddb8562df7ed6fcfb1164d23521f
RIPEMD160:           f4dd75a157313f2c56ddcaf71e1864c4b31738a6
SHA1:                d216c26392e87f2a7e3dd473f859efb4c8678337
Strict time order:   True
Number of interfaces in file: 1
Interface #0 info:
                     Encapsulation = Linux cooked-mode capture (25 - linux-sll)
                     Capture length = 65535
                     Time precision = microseconds (6)
                     Time ticks per second = 1000000
                     Number of stat entries = 0
                     Number of packets = 1771
```

Part 4. Investigate an Indicator of Compromise (IoC) using SQUERT

1. To start the investigation, open SQUERT:

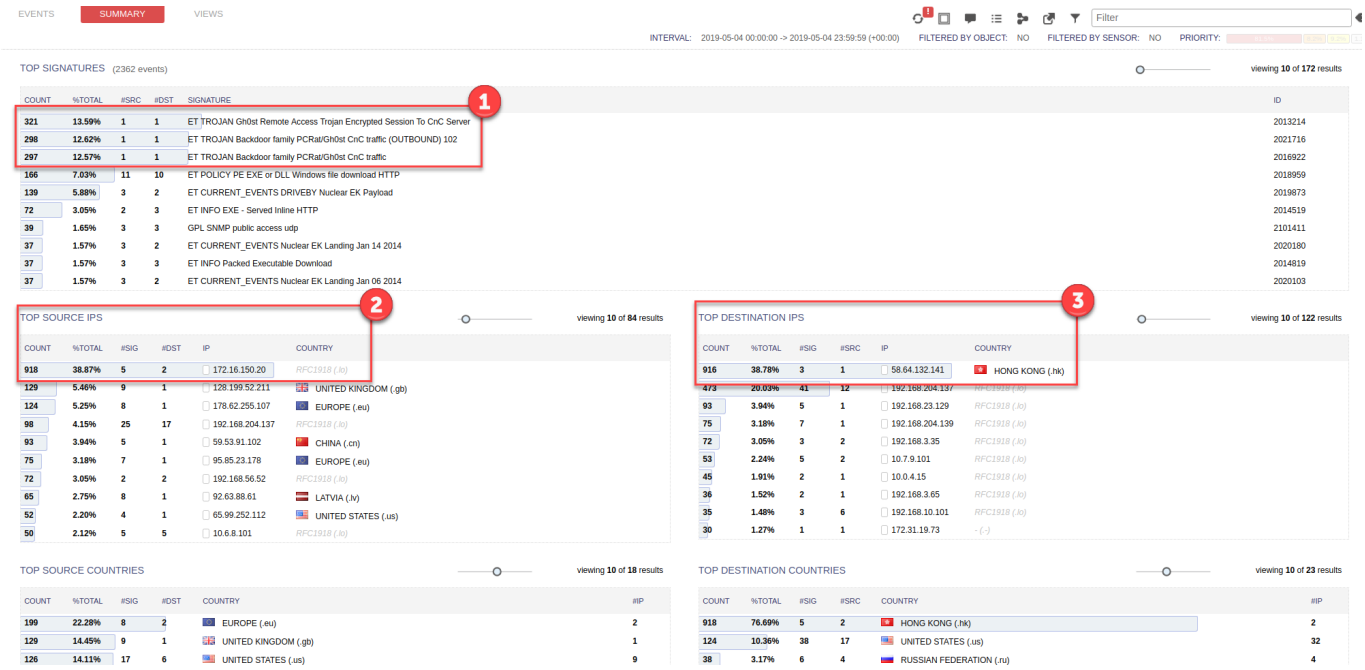
```
firefox https://localhost/squert
```

or double-click on the SQUERT icon on the desktop



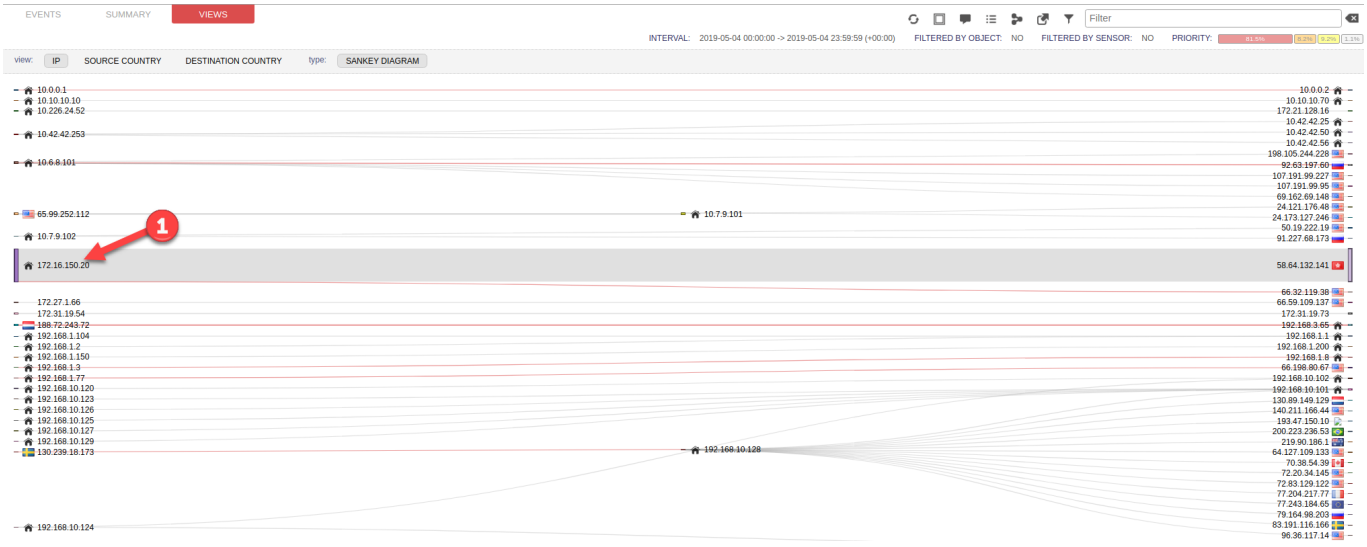
- EVENTS page, shows a list of events that have occured.
- SUMMARY page, shows a list of the top signatures and the top source and destinations via IP address and countries.
- VIEWS page, uses a Sankey diagram to show the relationships between IP addresses, source country and destination country.

2. Click on the SUMMARY page



- Looking at the summary, we can see there may be a Command and Control Trojan (CnC) activity in the packet capture file.
- A lot of traffic is originating from an IP address of 172.16.150.20
- A lot of traffic is going to an IP address of 58.64.132.141

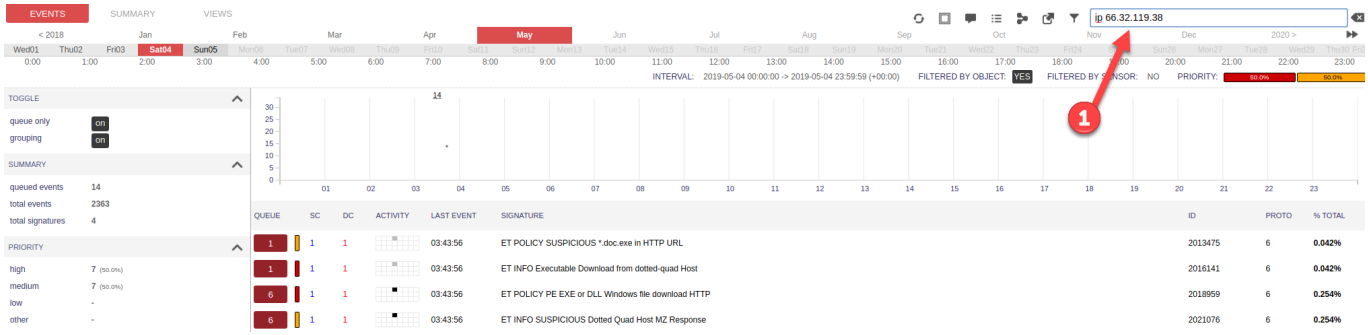
3. Click on the VIEWS page.



- From this Sankey diagram, you can see that the IP address of 172.16.150.20 is indeed mainly

talking to the IP address of 58.64.132.141. But there is also a red line showing a relationship with an IP address of 66.32.119.38

4. Click on the EVENTS page. Apply a filter for the IP address of 66.32.119.38 to see if any events have been logged about this activity.



- From the output it certainly does look like an Indicator of Compromise (IoC) because a suspicious file was downloaded from the IP address of 66.32.119.38.

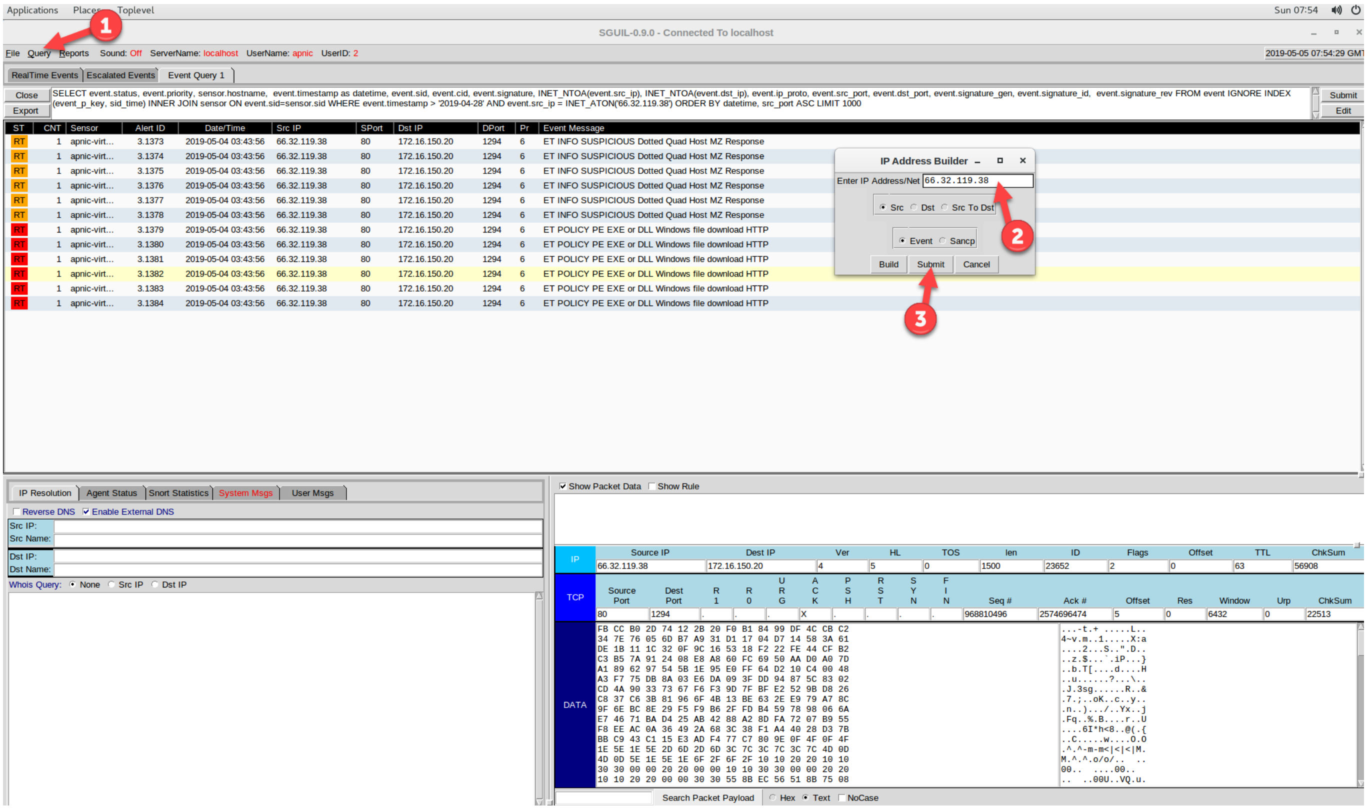
Part 5. Investigate an Indicator of Compromise (IoC) using SGUIL

1. To investigate further open sguil database to view the original logs and filter by the IP address of 66.32.119.38 to get an idea of the time it occurred.

2. To start the investigation, open SGUIL:

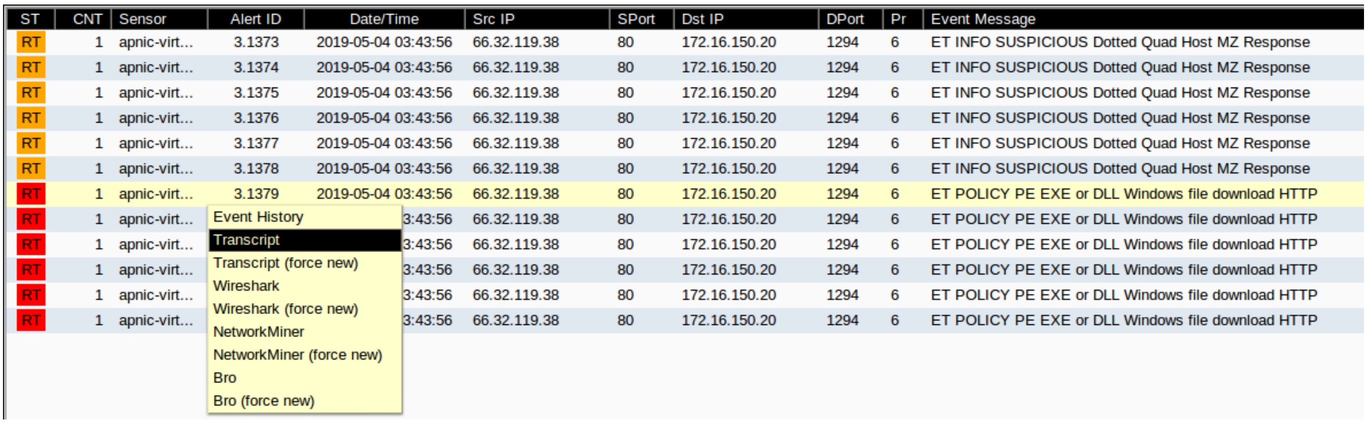
```
sguil.tk
```

or double-click on the SGUIL icon on the desktop.

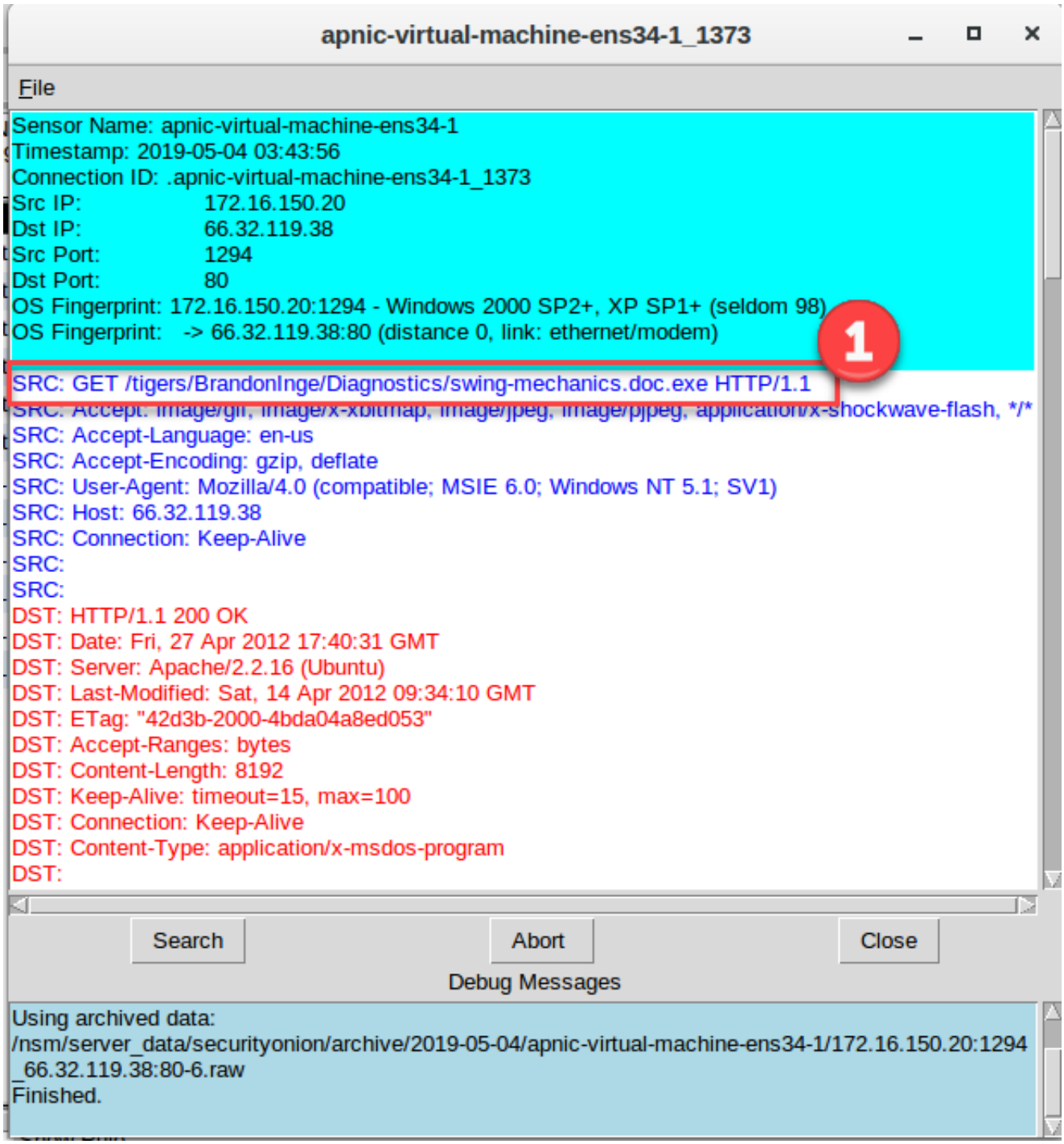


- Click on Query, type in the IP address of 66.32..119.38, then click on submit to apply the filter.

3. To view more details about the potential malicious file, do a ctl+right mouse click on the first PE EXE or DLL eventís Alert ID and click on transcript:



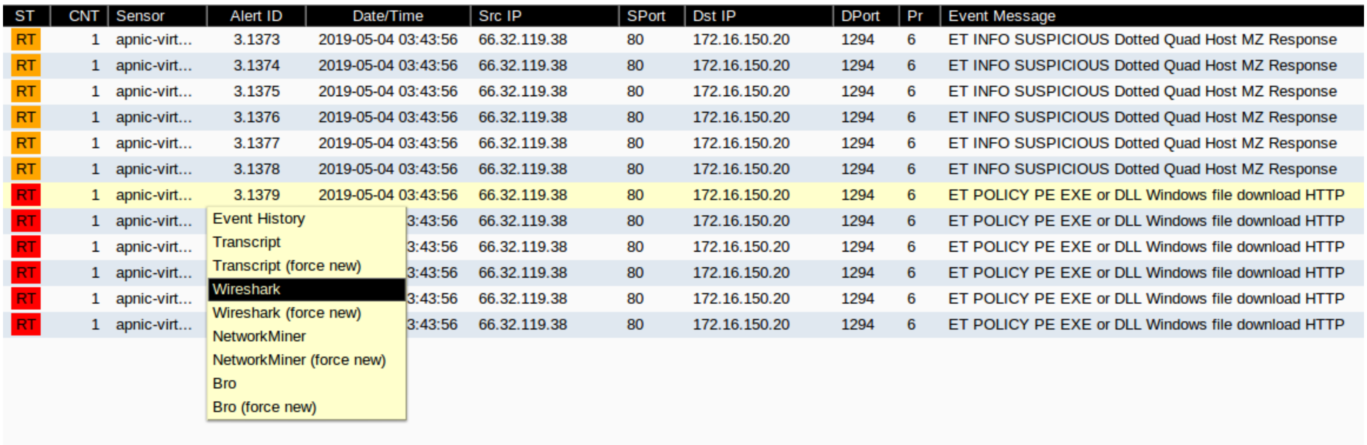
4. From this transcript you can see a document file name and can indeed see that there was an executable file that has potentially two file extensions (.EXE and .DOC) to try and fool the end user into thinking it is a word document extension.



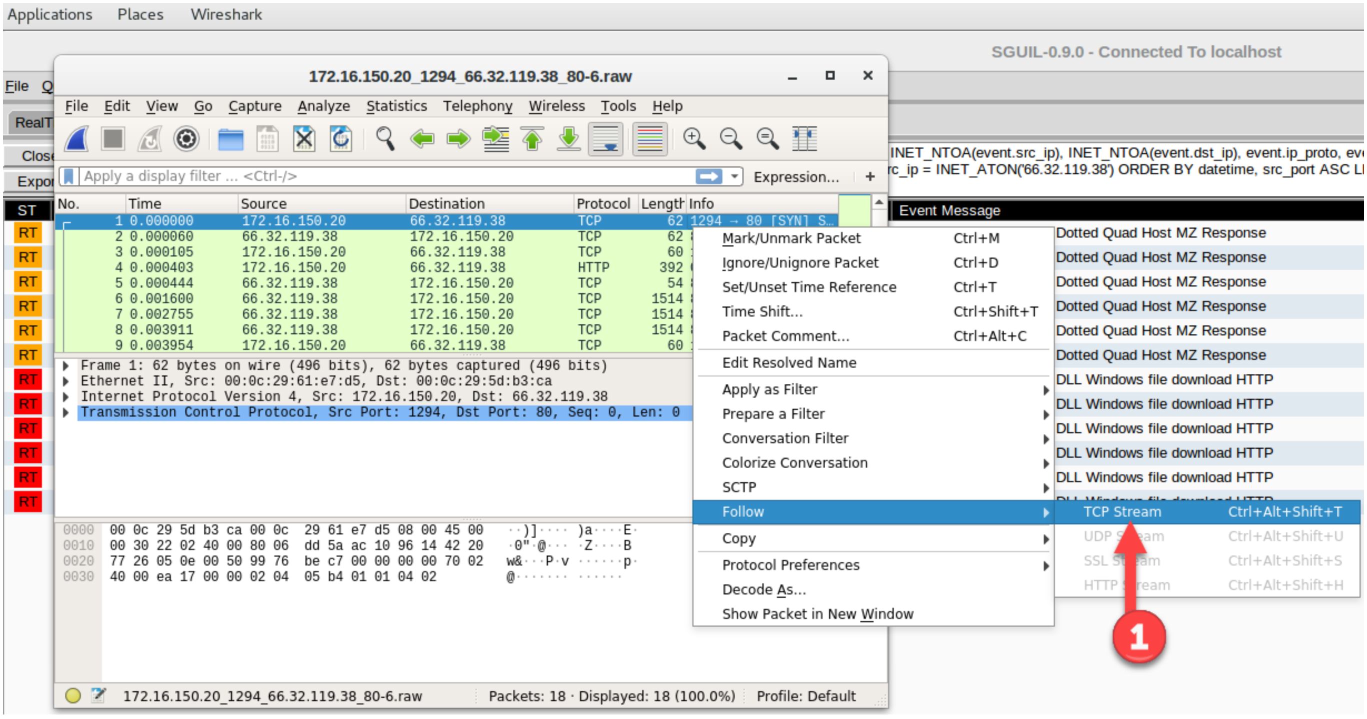
At this point you can extract the file for further analysis.

NOTE: this file could be malicious and should only be extracted on an isolated system.

5. The file can be extracted by using WireShark or NetworkMiner. To use WireShark do a ctl+right mouse click on the first PE EXE or DLL eventís Alert ID and click on WireShark.

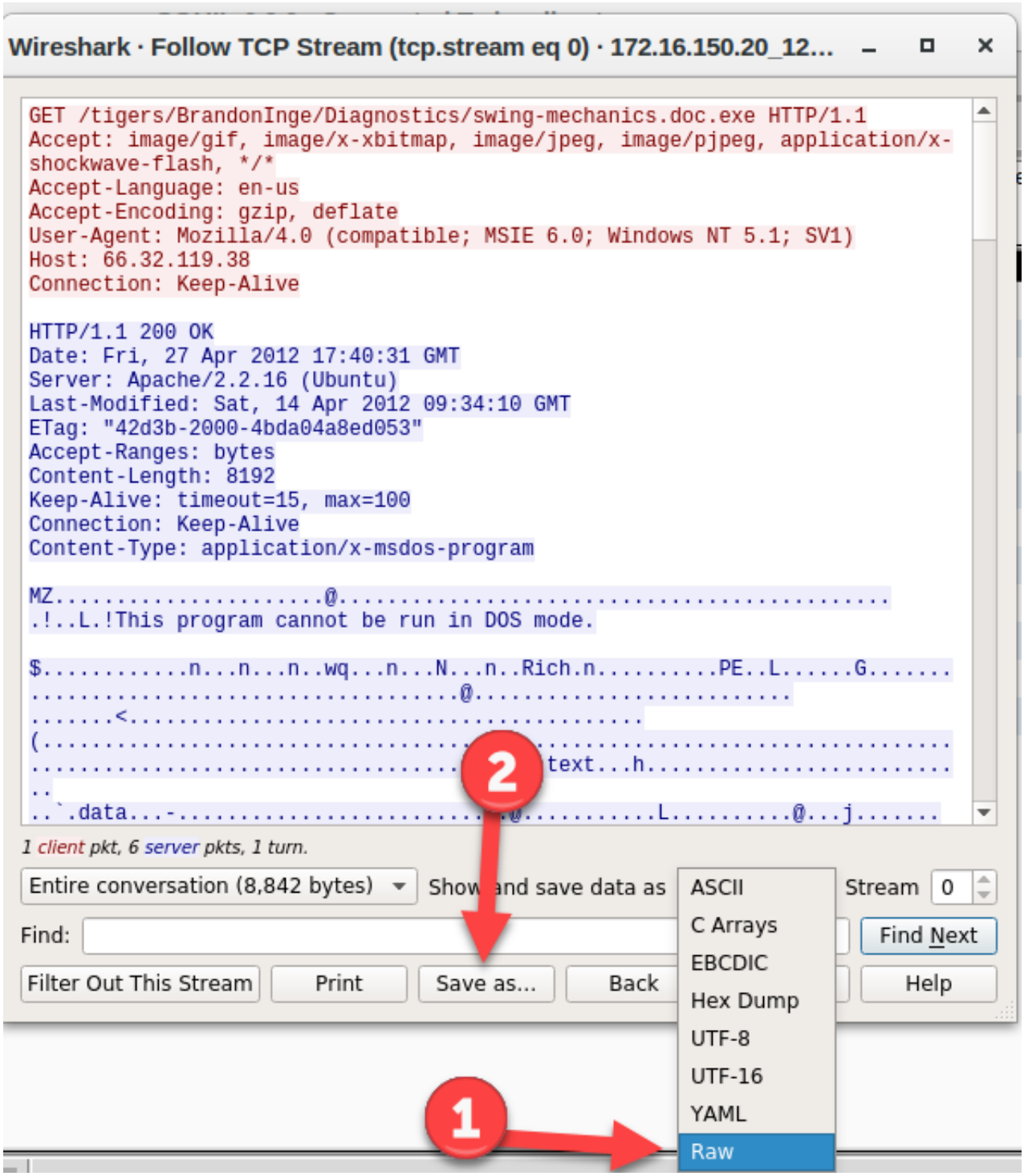


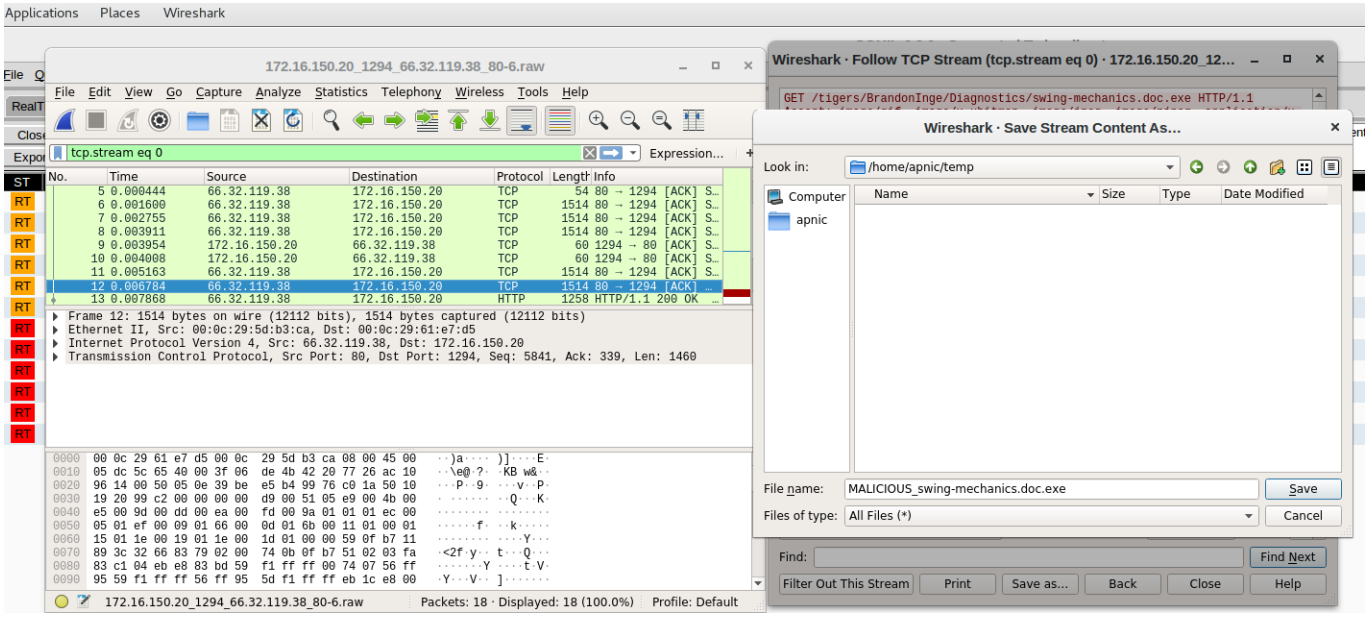
6. After opening WireShark, right-mouse click on the first packet, scroll down to follow and click on TCP stream.



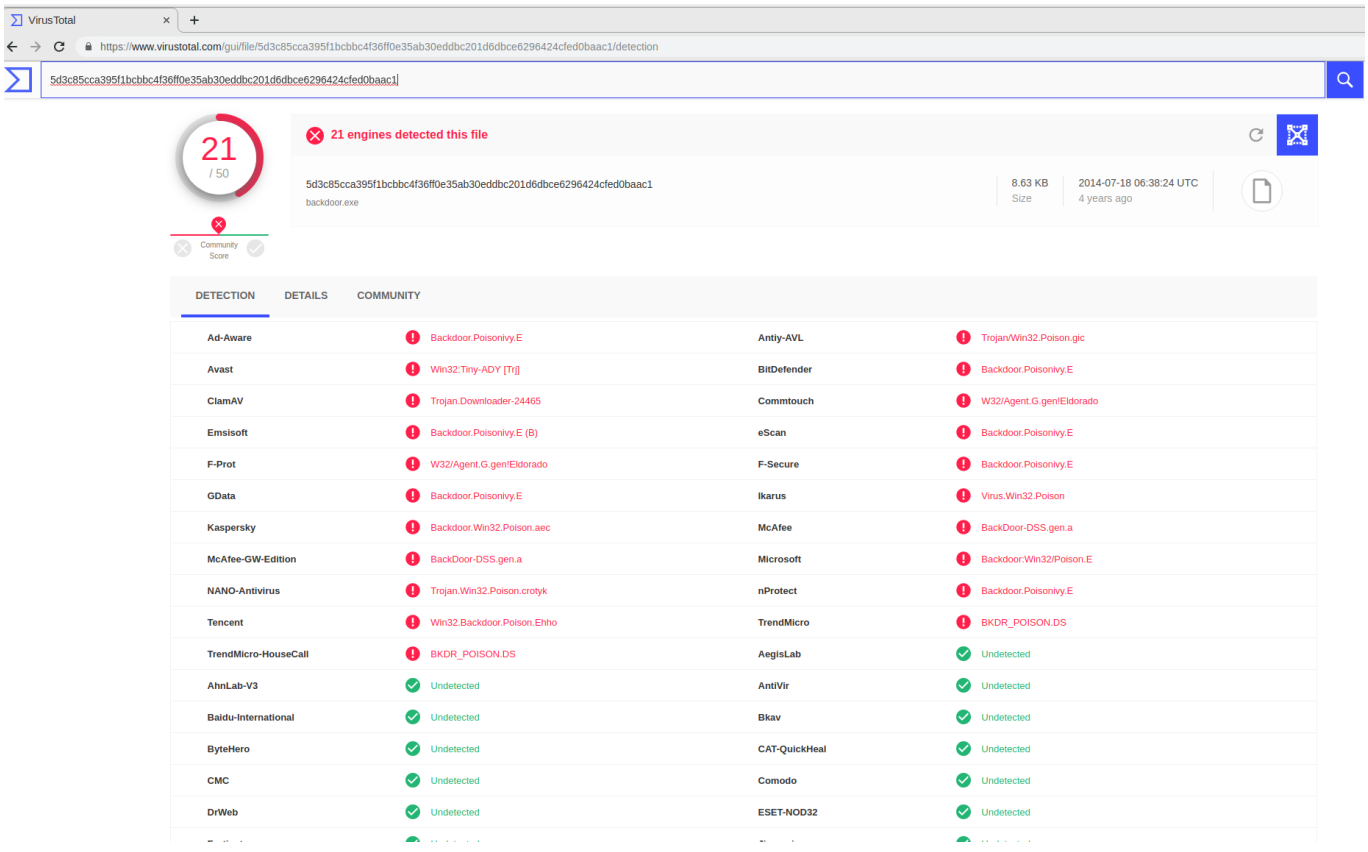
This will piece all the packets together and display the ASCII contents in the packet.

7. To extract the file, change the data type to RAW and click on Save as.


Wireshark · Follow TCP Stream (tcp.stream eq 0) · 172.16.150.20_12...
GET /tigers/BrandonInge/Diagnostics/swing-mechanics.doc.exe HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, */*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)
Host: 66.32.119.38
Connection: Keep-Alive
HTTP/1.1 200 OK
Date: Fri, 27 Apr 2012 17:40:31 GMT
Server: Apache/2.2.16 (Ubuntu)
Last-Modified: Sat, 14 Apr 2012 09:34:10 GMT
ETag: "42d3b-2000-4bda04a8ed053"
Accept-Ranges: bytes
Content-Length: 8192
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: application/x-msdos-program
MZ.....@.....
.!..L.!This program cannot be run in DOS mode.
$.....n...n...n..wq...n...N...n..Rich.n.....PE..L.....G.....
.....@.....
.....<.....
(.....
.....text...h.....
..
..`.data...-.....@.....L.....@...j.....
1 client pkt, 6 server pkts, 1 turn.
Entire conversation (8,842 bytes)
Show and save data as
ASCII
C Arrays
EBCDIC
Hex Dump
UTF-8
UTF-16
YAML
Raw
Stream
0
Find:
Find Next
Filter Out This Stream
Print
Save as...
Back
Help
1
2



8. Once the file is extracted it is a matter of using your favourite tools to analyse for malware. A quick way is to upload to Virus Total https://www.virustotal.com/gui/home/upload and see if the hash has been seen before.



NOTE: This is a public service, do not upload files that may contain company sensitive material.

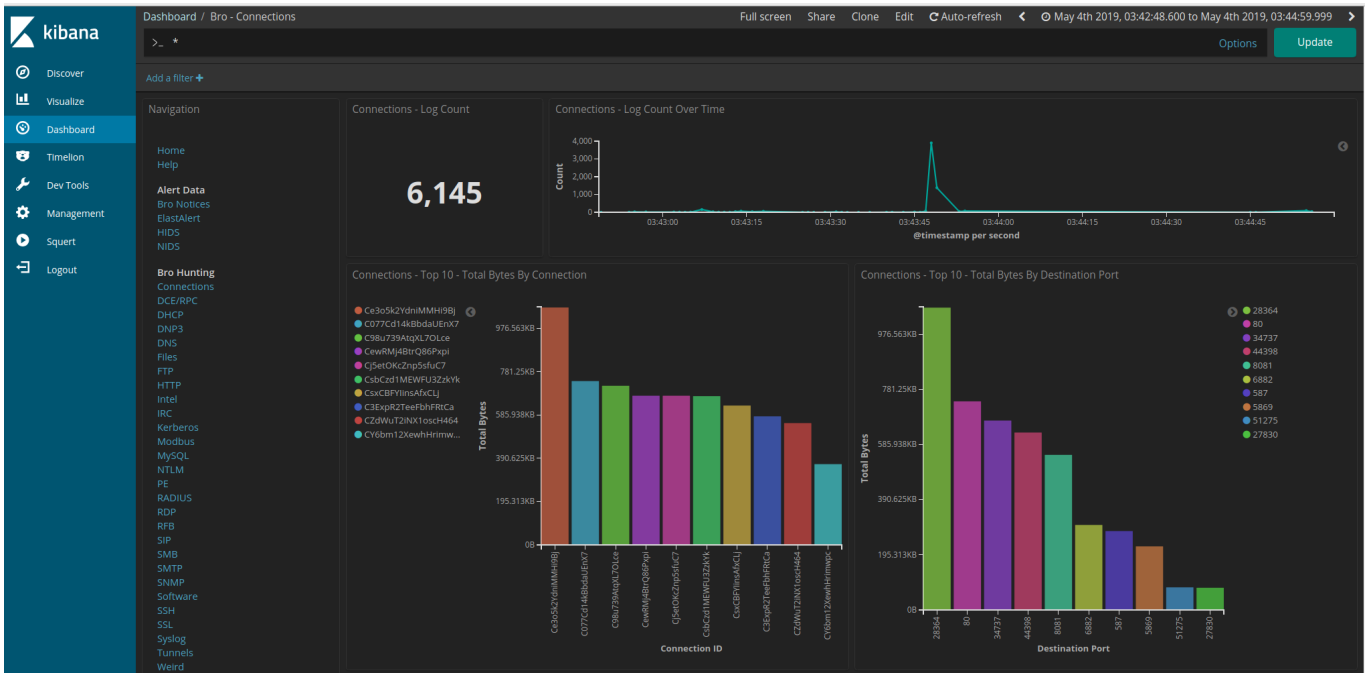
Part 6. Review an Indicator of Compromise (IoC) using Kibana

Another tool, that comes with Security onion that is worth a mention is Kibana. Kibana is an open source tool you can use to query the Elasticsearch database and display the results visually in a dashboard.

1. To open KIBANA:

```
firefox https://localhost/app/kibana
```

or double-click on the KIBANA icon on the desktop.



The malicious activity that was discovered above occurred around 3:43 GMT time. With this information, we can now focus on this time.

2. After opening Kibana, click on dashboard, restrict the time and date, and view events related to files.

And you can see that a file was downloaded during that time period.

Conclusion

So in a short amount of time, using security onion you were able to analysis a packet capture for an Indicator of Compromise or malicious activity, extract a suspicious file and determine that the file was indeed malicious.

With more practice, you should find that Security Onion is a valuable resource when it comes to network forensics and analysing packet captures, SNORT alerts and other logs.

https://github.com/Security-Onion-Solutions/security-onion/wiki/IntroductionWalkthrough