

## **Lab 10 – Log and Dynamic Malware Analysis**

Sai Sasaank Srivatsa Pallerla

University of Maryland, Baltimore County

Presented to: Gina Scaldaferri

Date: 11/17/2019

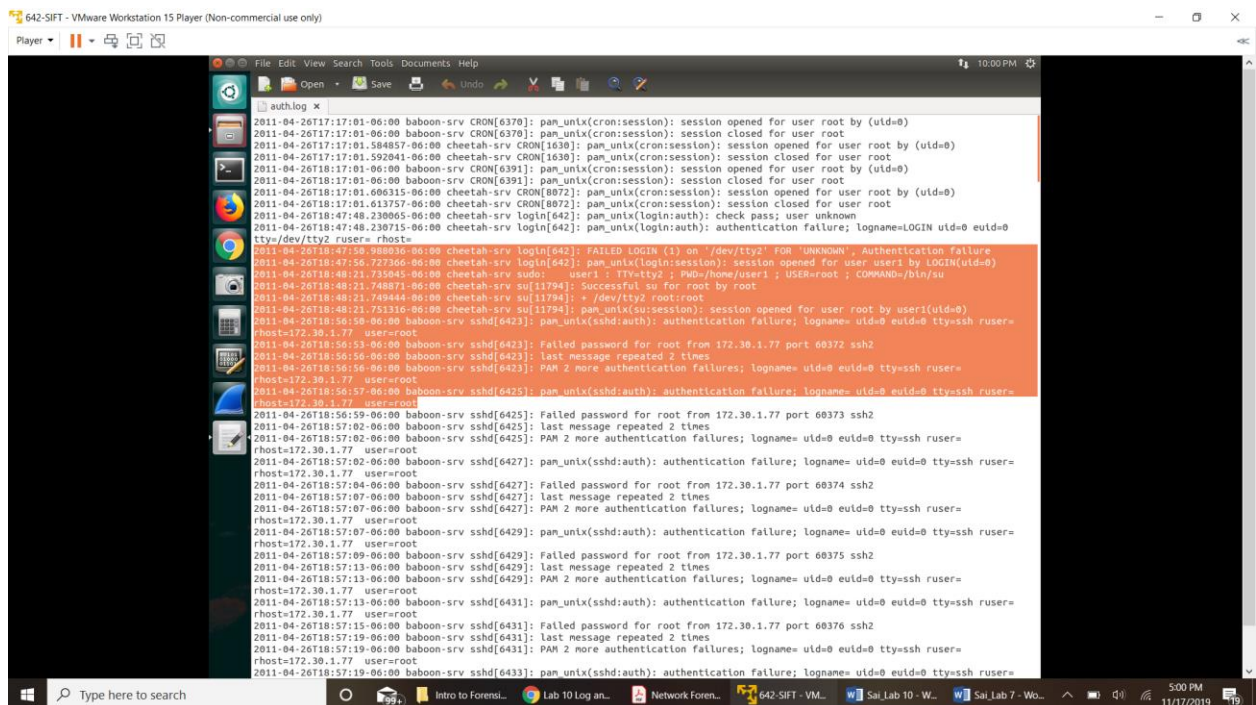
## About the Lab

Splunk software for searching, monitoring, and analyzing machine-generated big data, via a Web-style interface. It captures, indexes, and correlates real-time data in a searchable repository from which it can generate graphs, reports, alerts, dashboards, and visualizations. It is used for application management, security and compliance, as well as business and web analytics.

Wireshark is a network packet analyzer. A network packet analyzer presents captured packet data in as much detail as possible. It is used for network troubleshooting, analysis, software and communications, protocol development. It is one of the most famous tools that is used to monitor network traffic and protocols used. It lets us monitor the network at a microscopic level, both wired and wireless.

## Part 1. Log Analysis

1. Evaluate whether the failed login attempts were indicative of a deliberate attack. If so, identify the source and the target(s).



```
auth.log x
2011-04-26T17:17:01:00:00 baboon-srv CRON[6370]: pam_unix(cron:session): session opened for user root by (uid=0)
2011-04-26T17:17:01:00:00 cheetah-srv CRON[1630]: pam_unix(cron:session): session opened for user root by (uid=0)
2011-04-26T17:17:01:00:00 cheetah-srv CRON[1630]: pam_unix(cron:session): session closed for user root
2011-04-26T17:17:01:00:00 baboon-srv CRON[6391]: pam_unix(cron:session): session opened for user root by (uid=0)
2011-04-26T17:17:01:00:00 baboon-srv CRON[6391]: pam_unix(cron:session): session closed for user root
2011-04-26T18:17:01:00:00 cheetah-srv CRON[8072]: pam_unix(cron:session): session opened for user root by (uid=0)
2011-04-26T18:17:01:00:00 cheetah-srv CRON[8072]: pam_unix(cron:session): session closed for user root
2011-04-26T18:47:48:230065:00:00 cheetah-srv login[642]: pam_unix(login:auth): check pass; user unknown
2011-04-26T18:47:48:230715:00:00 cheetah-srv login[642]: pam_unix(login:auth): authentication failure; logname=LOGIN uid=0 euid=0
tty=/dev/tty2 ruser= rhost=
2011-04-26T18:47:58:988036:00:00 cheetah-srv login[642]: FAILED LOGIN (1) on '/dev/tty2' FOR 'UNKNOWN': Authentication failure
2011-04-26T18:47:58:727366:00:00 cheetah-srv login[642]: pam_unix(login:session): session opened for user user1 by LOGIN(uid=0)
2011-04-26T18:48:21:735045:00:00 cheetah-srv sudo: user1 : TTY=ttty2 ; PWD=/home/user1 : USER=root : COMMAND=/bin/su
2011-04-26T18:48:21:748871:00:00 cheetah-srv su[11794]: Successful su for root by root
2011-04-26T18:48:21:749444:00:00 cheetah-srv su[11794]: + /dev/tty2 root:root
2011-04-26T18:48:21:751316:00:00 cheetah-srv su[11794]: pam_unix(su:session): session opened for user root by user1(uid=0)
2011-04-26T18:50:58:00:00 baboon-srv sshd[6423]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6423]: Failed password for root from 172.30.1.77 port 60372 ssh2
2011-04-26T18:50:58:00:00 baboon-srv sshd[6423]: last message repeated 2 times
2011-04-26T18:50:58:00:00 baboon-srv sshd[6423]: PAR 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6425]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6425]: Failed password for root from 172.30.1.77 port 60373 ssh2
2011-04-26T18:50:58:00:00 baboon-srv sshd[6425]: last message repeated 2 times
2011-04-26T18:50:58:00:00 baboon-srv sshd[6425]: PAR 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6427]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6427]: Failed password for root from 172.30.1.77 port 60374 ssh2
2011-04-26T18:50:58:00:00 baboon-srv sshd[6427]: last message repeated 2 times
2011-04-26T18:50:58:00:00 baboon-srv sshd[6427]: PAR 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6429]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6429]: Failed password for root from 172.30.1.77 port 60375 ssh2
2011-04-26T18:50:58:00:00 baboon-srv sshd[6429]: last message repeated 2 times
2011-04-26T18:50:58:00:00 baboon-srv sshd[6429]: PAR 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6431]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6431]: Failed password for root from 172.30.1.77 port 60376 ssh2
2011-04-26T18:50:58:00:00 baboon-srv sshd[6431]: last message repeated 2 times
2011-04-26T18:50:58:00:00 baboon-srv sshd[6431]: PAR 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T18:50:58:00:00 baboon-srv sshd[6433]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
```

When examining the auth.log file we can see there are many failed authentication sessions. To be more specific, we see one failed authentication on the host cheetah-srv and too many authentication failures (214) on baboon-srv.

Understanding the auth.log file.

```
2011-04-26T18:56:50-06:00 baboon-srv sshd[6423]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
```

- 2011-04-26T18:56:50-06:00: This is the time stamp of the failed authentication log
- baboon-srv: Host on which the authentication failed
- pam\_unix(sshd:auth): authentication failure: Linux Pluggable Authentication Modules (PAM) provide dynamic authentication support for applications and services in a Linux system and this mentioned that the failure occurred at this interface/ plugin
- logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root – These the details submitted during the authentication, like source IP, what kind of user, login name submitted and its related user id, type of connection (ssh).

On the server baboon-srv we see there many failed authentications within a couple of minutes, which is very suspicious and just by the looks of it we can say it's a brute-force attack. Examining the log file, we see the source is 172.30.1.77 and the target is the user bob and root

```
2011-04-26T19:00:51-06:00 baboon-srv sshd[6501]: Failed password for root from 172.30.1.77 port 49184 ssh2
2011-04-26T19:00:54-06:00 baboon-srv sshd[6501]: last message repeated 2 times
2011-04-26T19:00:54-06:00 baboon-srv sshd[6501]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T19:00:54-06:00 baboon-srv sshd[6503]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=root
2011-04-26T19:00:56-06:00 baboon-srv sshd[6503]: Failed password for root from 172.30.1.77 port 49185 ssh2
2011-04-26T19:00:57-06:00 baboon-srv sshd[6505]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=bob
2011-04-26T19:00:59-06:00 baboon-srv sshd[6505]: Failed password for bob from 172.30.1.77 port 49186 ssh2
```

2. Determine whether any systems were compromised. If so, describe the extent of the compromise.

Yes, the user account named bob was compromised. We can be sure of this after seeing the words “Accepted password for bob” in the auth.logs file.

```
2011-04-26T19:04:05-06:00 baboon-srv sshd[6559]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=bob
2011-04-26T19:04:05-06:00 baboon-srv sshd[6561]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=172.30.1.77 user=bob
2011-04-26T19:04:07-06:00 baboon-srv sshd[6561]: Failed password for bob from 172.30.1.77 port 49214 ssh2
2011-04-26T19:04:07-06:00 baboon-srv sshd[6561]: Accepted password for bob from 172.30.1.77 port 49214 ssh2
2011-04-26T19:04:07-06:00 baboon-srv sshd[6561]: pam_unix(sshd:session): session opened for user bob by (uid=0)
2011-04-26T19:04:08-06:00 baboon-srv sshd[6631]: Received disconnect from 172.30.1.77: 11:
2011-04-26T19:04:08-06:00 baboon-srv sshd[6561]: pam_unix(sshd:session): session closed for user bob
2011-04-26T19:04:33-06:00 baboon-srv sshd[6632]: Accepted password for bob from 172.30.1.77 port 49215 ssh2
2011-04-26T19:04:33-06:00 baboon-srv sshd[6632]: pam_unix(sshd:session): session opened for user bob by (uid=0)
2011-04-26T19:05:10-06:00 baboon-srv sudo: pam_unix(sudo:auth): authentication failure; logname=bob uid=0 euid=0 tty=/dev/pts/0
ruser= rhost= user=bob
```

We can also see what the attacker did after successfully logging in as bob

```
2011-04-26T19:04:33-06:00 baboon-srv sshd[6632]: pam_unix(sshd:session): session opened for user bob by (uid=0)
2011-04-26T19:05:10-06:00 baboon-srv sudo: pam_unix(sudo:auth): authentication failure; logname=bob uid=0 euid=0 tty=/dev/pts/0
ruser= rhost= user=bob
2011-04-26T19:05:18-06:00 baboon-srv sudo: bob : TTY=pts/0 ; PWD=/home/bob ; USER=root ; COMMAND=/usr/bin/vi /var/log/auth.log
2011-04-26T19:05:34-06:00 baboon-srv sudo: bob : TTY=pts/0 ; PWD=/home/bob ; USER=root ; COMMAND=/usr/sbin/tcpdump -nni eth0
2011-04-26T19:07:03-06:00 baboon-srv sudo: bob : TTY=pts/0 ; PWD=/home/bob ; USER=root ; COMMAND=/usr/bin/apt-get update
2011-04-26T19:07:15-06:00 baboon-srv sudo: bob : TTY=pts/0 ; PWD=/home/bob ; USER=root ; COMMAND=/usr/bin/apt-get install nmap
2011-04-26T19:14:53-06:00 baboon-srv sshd[6632]: pam_unix(sshd:session): session closed for user bob
2011-04-26T19:17:01-06:00 baboon-srv CRON[6797]: pam_unix(cron:session): session opened for user root by (uid=0)
2011-04-26T19:17:01-06:00 baboon-srv CRON[6797]: pam_unix(cron:session): session closed for user root
2011-04-26T19:17:01.627528-06:00 cheetah-srv CRON[11843]: pam_unix(cron:session): session opened for user root by (uid=0)
2011-04-26T19:17:01.634766-06:00 cheetah-srv CRON[11843]: pam_unix(cron:session): session closed for user root
```

The attacker opened the auth.log file and tcpdump of bobs network and made changes to them. We could get the unaltered log file because the company must have stored log files in a remote system, unreachable by the attacker. Later the attacker also installed nmap, maybe to check what system and services are running in the network.

Now, we move on to the next log file which is the firewall.log, Here we search for entries from the suspicious IP address 172.30.1.77 and we can see that he was communicating with the IP address 10.30.30.20 (DMZ network) and 10.30.30.20 in turn communicates with 192.168.30.101 (after the attacker successfully logged in we see an internal IP address 192.168.30.101 which could be bob's IP address).



Upon later inspection we see 192,168.30.101 communicating with 172.30.1.77, meaning something was sent from an internal system to the external address.

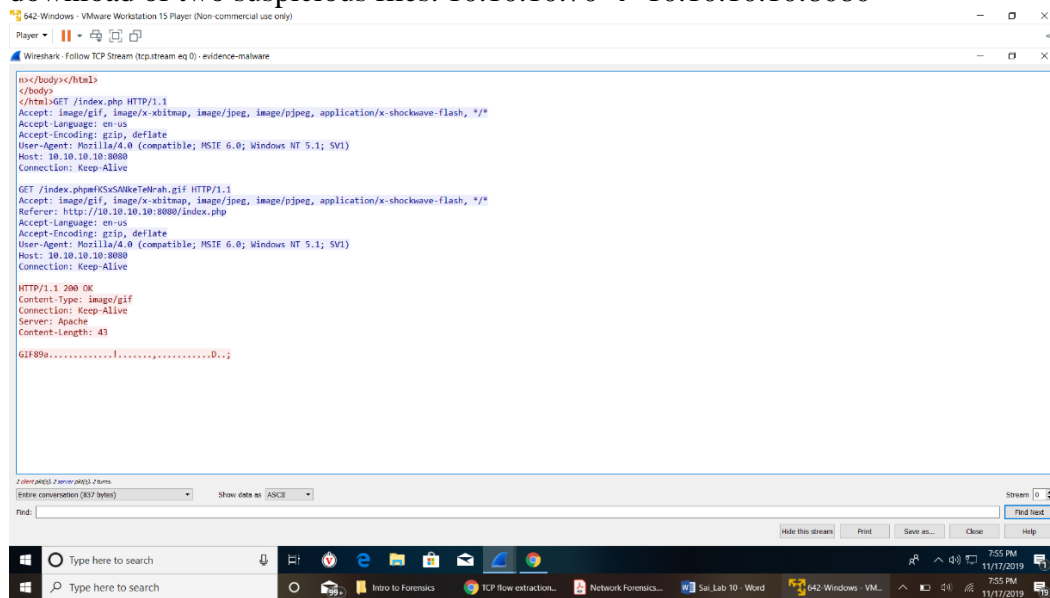
```
2011-04-26T19:11:35-06:00 ant-fw : %ASA-6-106100: access-list inside permitted udp inside/192.168.30.30(59302) -> dmz/10.30.30.20(53)
hit-cnt 1 first hit [0xb820d39, 0x0]
2011-04-26T19:11:35-06:00 ant-fw : %ASA-6-106100: access-list inside permitted udp inside/192.168.30.30(42598) -> dmz/10.30.30.20(53)
hit-cnt 1 first hit [0xb820d39, 0x0]
2011-04-26T19:11:38-06:00 ant-fw : %ASA-6-106100: access-list inside permitted udp inside/192.168.30.30(55176) -> dmz/10.30.30.20(53)
hit-cnt 1 first hit [0xb820d39, 0x0]
2011-04-26T19:11:39-06:00 ant-fw : %ASA-6-106100: access-list inside permitted tcp inside/192.168.30.101(1399) -> outside/172.30.1.77
(21) hit-cnt 1 first hit [0x2989a4a8, 0x0]
2011-04-26T19:11:39-06:00 ant-fw : %ASA-6-106100: access-list inside permitted udp inside/192.168.30.30(37711) -> dmz/10.30.30.20(53)
hit-cnt 1 first hit [0xb820d39, 0x0]
```

## Part 2. Dynamic Malware Analysis

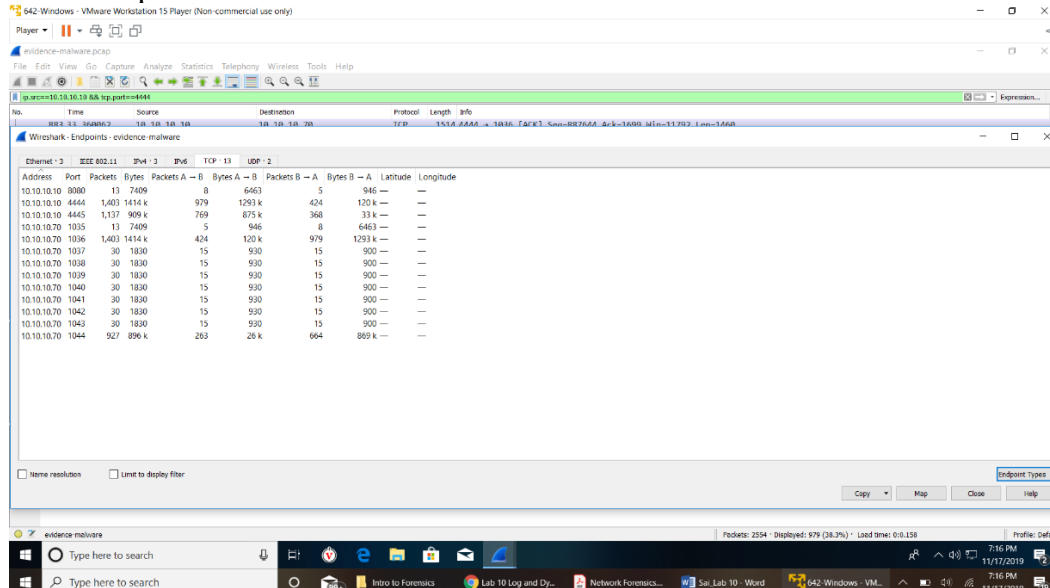
1. Identify the source of the compromise.
2. Recover malware from the packet capture and provide it to investigators for further analysis.

Initial examination of the pacp doesn't give out any suspicious behavior. To examine it further we check the endpoint communications in the pacp. Here we see that there is communication with an external address 10.10.10.10 and the client's computer 10.10.10.70, the communication when about like this

1. The user might have accessed some website that runs on 10.10.10.10 that lead to download of two suspicious files. 10.10.10.70 → 10.10.10.10:8080



2. Server response 10.10.10.10:4444 → 10.10.10.70:1036
3. Server response 10.10.10.10:4445 → 10.10.10.70:1044



10.10.10.10:4444  $\rightarrow$  10.10.10.70:1036

- 
- The screenshot displays a Windows desktop environment. The primary application is Wireshark, which is capturing a NetStream connection. The main pane shows the raw packet data in hexadecimal and ASCII. The ASCII column contains the following text: "This program cannot be run in DOS mode." The packet list on the right shows a single packet of type "SACK\_PACKET". The bottom of the screen shows the Windows taskbar with various icons and the system clock.

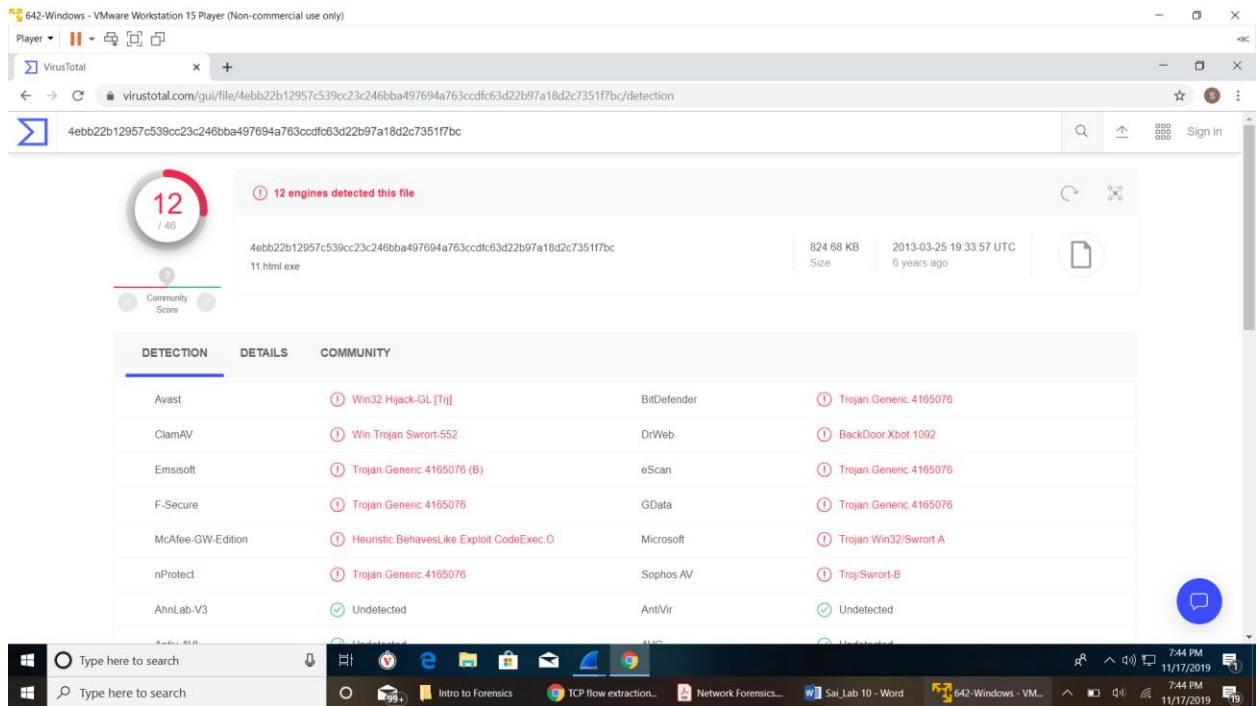
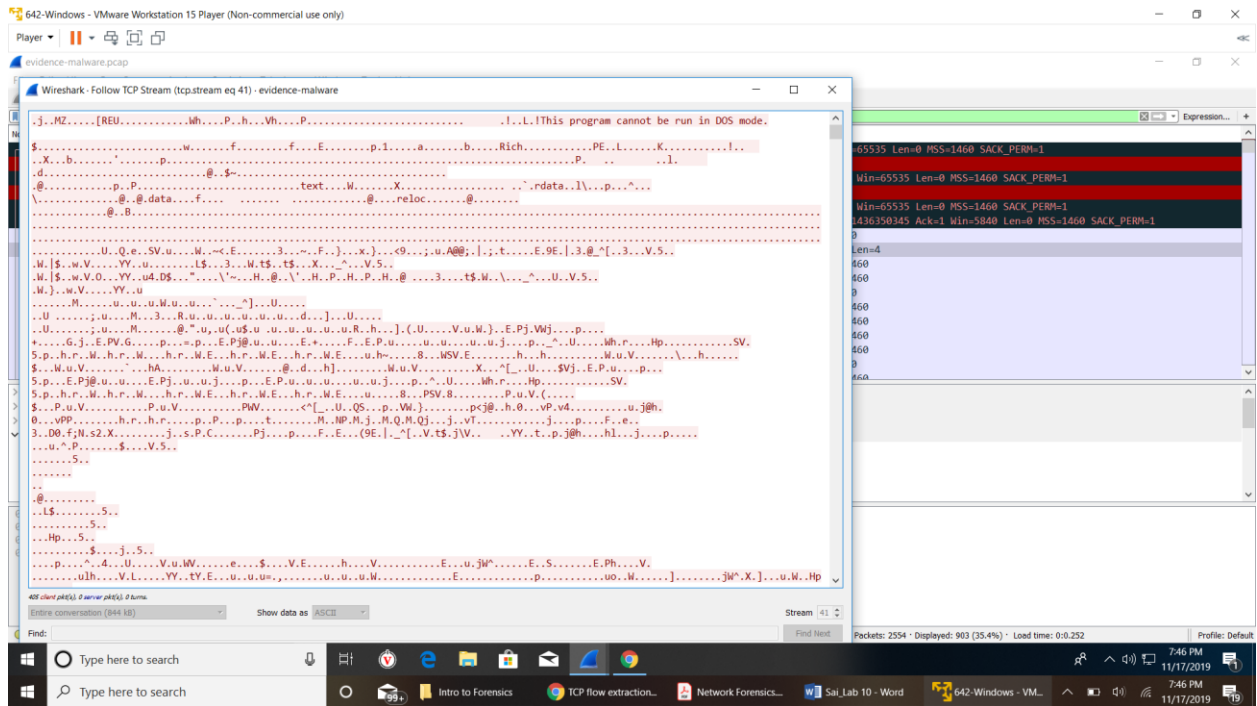
The screenshot displays the VirusTotal web interface for a specific file hash. The file is identified as 'a185891f23a44eac440681019325d3d589fae346f3a0ba86ef8234e68e908ec1'. A red circular badge indicates that 22 out of 59 engines have detected the file as malicious. The file's metadata shows it is 1.27 MB and was analyzed on December 20, 2017, at 09:20 UTC. Below this, a table titled 'DETECTION' lists the results from various antivirus engines. The engines are organized into three columns: Arcabit, AVG, BitDefender, Comodo, Emsisoft, F-Secure, Ikarus, and others. Each engine's verdict is shown, with many detecting the file as a Trojan or malware. The bottom of the image shows a Windows taskbar with various open applications and the system clock.

DETECTION	DETAILS	COMMUNITY
Arcabit	Gen HackTool MeterPreter 1	Avast
AVG	Multi:Sworot A [Trij]	Avira (no cloud)
BitDefender	Gen HackTool MeterPreter 1	Win Tool MeterPreter-6264292-0
Comodo	Unclassified/Malware	DrWeb
Emsisoft	Gen HackTool MeterPreter 1 (5)	eScan
F-Secure	Gen HackTool MeterPreter 1	GData
Ikarus	Trojan Win32_Sworot	Kaspersky



10.10.10.10:4445 → 10.10.10.70:1044

1. Apply the filter “ip.src==10.10.10.10 && tcp.port==4445”
2. Right click on the first packet and select “Follow TCP Stream”
3. Save the TCP stream using “raw” option and name it as 4445.exe
4. Upload the exe to Virustotal to scan the executable



## Conclusion

From this lab we know how important it is to keep your security team in the loop. Because Vick Times was cautious enough, he could avert a huge attack against him. The same thing could be told about Bob's Dry Cleaners, they were cautious enough to save a copy of logs in a remote server that enabled the investigators to pinpoint how and what the attacker did.

## References

- [1] <https://stackoverflow.com/questions/1385059/tcp-flow-extraction>
- [2] <https://answers.splunk.com/answers/91419/how-to-get-remote-linux-log-into-splunk.html>
- [3] [https://en.wikipedia.org/wiki/Linux\\_PAM](https://en.wikipedia.org/wiki/Linux_PAM)
- [4] [https://www.wireshark.org/docs/wsug\\_html\\_chunked/ChAdvFollowStreamSection.html](https://www.wireshark.org/docs/wsug_html_chunked/ChAdvFollowStreamSection.html)