Lab 10 – Log and Malware Analysis

Shrutika Joshi

University of Maryland Baltimore County

Presented To – Gina Marie

Date – 02nd Aug 2023

Scenario

Bob's Dry Cleaners keeps credit card numbers and personal contact information for their Platinum Dry Cleaning customers (many of whom are executives). They need to make sure that this credit card data remains secure. If you find evidence of a compromise, provide an analysis of the risk that confidential information was stolen. Be sure to carefully justify your conclusions. Security staff at Bob's Dry Cleaners collects operating system logs from servers and workstations, as well as firewall logs. These are automatically sent over the network from each system to a central log collection server running rsyslogd (192.168.30.30). Security staffs have provided you with log files from the time period in question. These log files include:

auth.log—System authentication and privileged command logs from Linux servers

workstations.log—Logs from Windows workstations

firewall.log—Cisco ASA firewall logs

Introduction

Investigate potentially malicious traffic by analyzing authentication logs, windows logs, and Cisco ASA firewall logs. Also, perform dynamic malware analysis on the pcap file provided and analyze what type of malware running on the user's machine in order to determine what it did, and how it communicated.

Pre-Lab

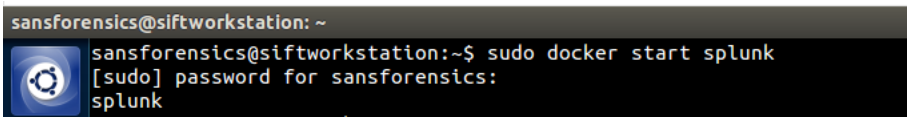
For this lab, we will be using a Linux machine, Splunk docker container, Wireshark, open source malware analysis tools like Virustotal

Analysis

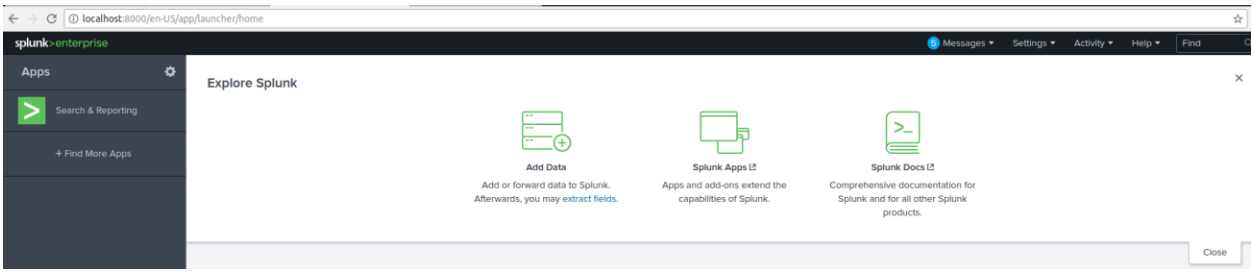
1. Log Analysis

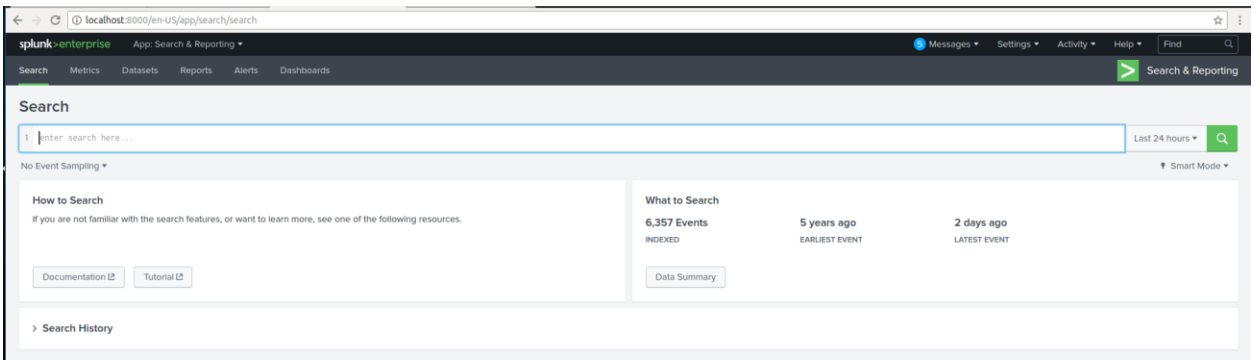
First, run the command – 'sudo docker start splunk'. Once started, browse to Splunk using 'localhost:8000'.

Docker Container - A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application. It offers portability, consistency, and scalability for deploying applications in different environments. Hence it is widely being used.

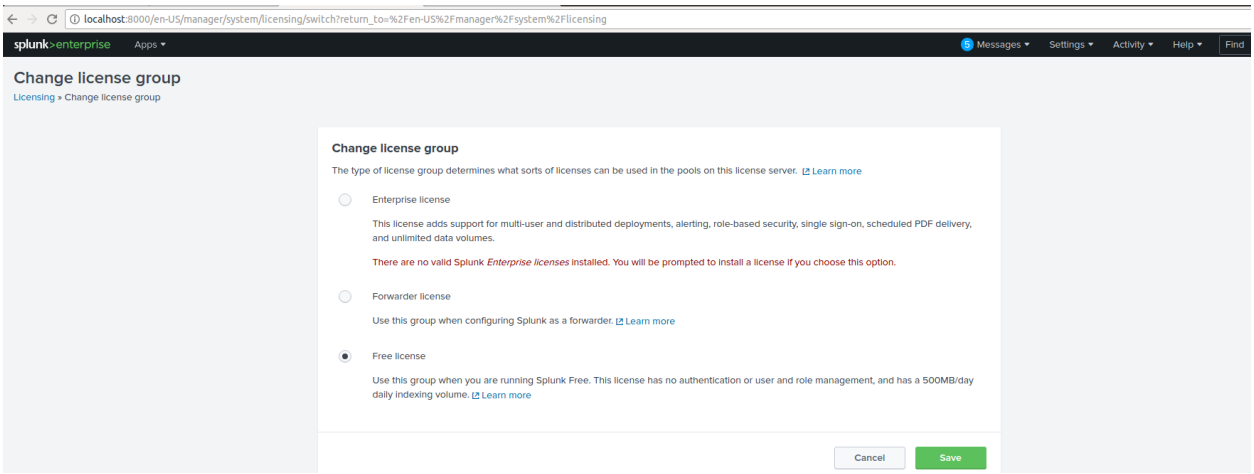


Log in to Splunk and move into the 'Search & Reporting' tab.





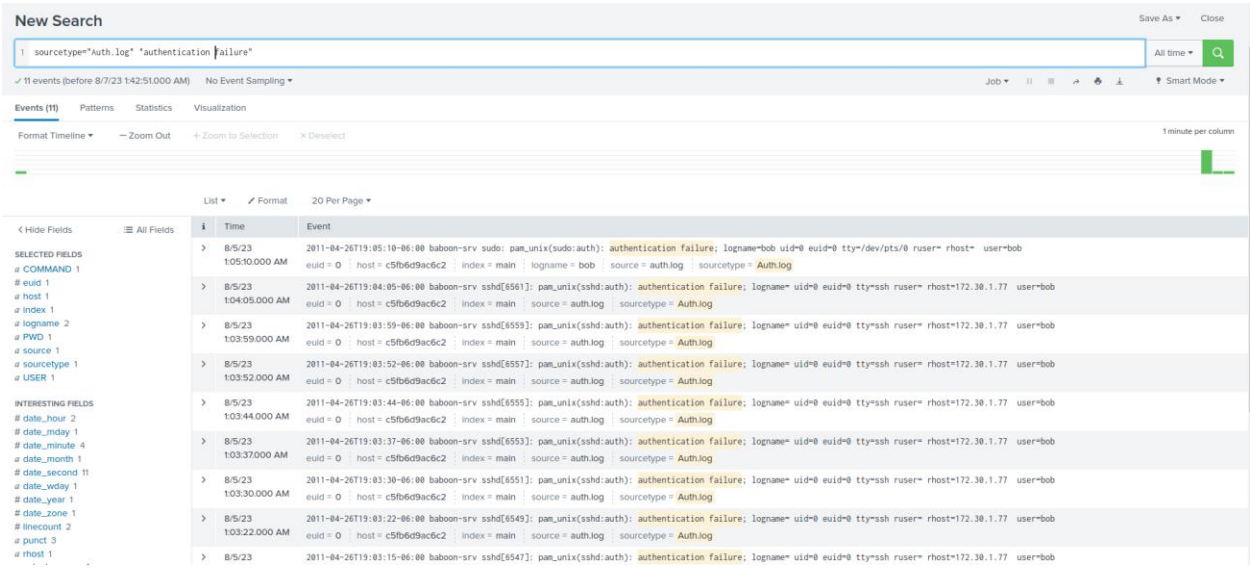
Click on Licensing → change license group → Free license



Network: Internal network: 192.168.30.0/24 DMZ network: 10.30.30.0/24 “Internet”: 172.30.1.0/24

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

I have filter logs for sourcetype=”auth.logs” and added keyword ‘authentication failure’ to check the failed login attempts and found that there are ssh authentication failure logs for user ‘bob’ and the target hostname seems to be ‘baboon-srv’. The source IP address is 172.30.1.77 which is external IP address.



Also, there are some authentication failure logs for user ‘root’ for system ‘baboon-srv’ and ‘cheetah-srv’.

Since there are more logs for the system ‘’baboon-srv’. Let’s filter the logs for this system using Splunk query - sourcetype="Auth.log" "baboon-srv". After searching we found the authentication failure for a single event is happening three times.

First event have one authentication attempt- Failed password for bob from 172.30.1.77 port 49207 ssh2

Second event have two authentication attempt - PAM 2 more authentication failures

Also, this authentication failure attempt started at 18:56:50 and ended at 19:03:08 and each authentication attempt have 7 seconds gap which indicates that it is a brute force password guessing attack.

What is a brute force password attack? It is a hacking method that uses trial and error to crack login credentials, and encryption keys for gaining unauthorized access to the organisation's system and networks. The attack utility is typically configured to run either until the attack is successful or the wordlist is exhausted. (Brute Force Attack, n.d.)

i	Time	Event
		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
		2011-04-26T18:56:53-06:00 baboon-srv sshd[6423]: Failed password for root from 172.30.1.77 port 60372 ssh2
		2011-04-26T18:56:56-06:00 baboon-srv sshd[6423]: last message repeated 2 times
		2011-04-26T18:56:56-06:00 baboon-srv sshd[6423]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:56:57-06:00 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
		2011-04-26T18:56:59-06:00 baboon-srv sshd[6425]: Failed password for root from 172.30.1.77 port 60373 ssh2
		2011-04-26T18:57:02-06:00 baboon-srv sshd[6425]: last message repeated 2 times
		2011-04-26T18:57:02-06:00 baboon-srv sshd[6425]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:02-06:00 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
		2011-04-26T18:57:04-06:00 baboon-srv sshd[6427]: Failed password for root from 172.30.1.77 port 60374 ssh2
		2011-04-26T18:57:07-06:00 baboon-srv sshd[6427]: last message repeated 2 times
		2011-04-26T18:57:07-06:00 baboon-srv sshd[6427]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:07-06:00 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
		2011-04-26T18:57:09-06:00 baboon-srv sshd[6429]: Failed password for root from 172.30.1.77 port 60375 ssh2
		2011-04-26T18:57:13-06:00 baboon-srv sshd[6429]: last message repeated 2 times
		2011-04-26T18:57:13-06:00 baboon-srv sshd[6429]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:13-06:00 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
		2011-04-26T18:57:15-06:00 baboon-srv sshd[6431]: Failed password for root from 172.30.1.77 port 60376 ssh2
		2011-04-26T18:57:19-06:00 baboon-srv sshd[6431]: last message repeated 2 times
		2011-04-26T18:57:19-06:00 baboon-srv sshd[6431]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:19-06:00 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
		2011-04-26T18:57:21-06:00 baboon-srv sshd[6433]: Failed password for root from 172.30.1.77 port 60377 ssh2
		2011-04-26T18:57:26-06:00 baboon-srv sshd[6433]: last message repeated 2 times
		2011-04-26T18:57:26-06:00 baboon-srv sshd[6433]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:26-06:00 baboon-srv sshd[6435]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:28-06:00 baboon-srv sshd[6435]: Failed password for root from 172.30.1.77 port 60378 ssh2
		2011-04-26T18:57:31-06:00 baboon-srv sshd[6435]: last message repeated 2 times
		2011-04-26T18:57:31-06:00 baboon-srv sshd[6435]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:31-06:00 baboon-srv sshd[6437]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:34-06:00 baboon-srv sshd[6437]: Failed password for root from 172.30.1.77 port 60379 ssh2
		2011-04-26T18:57:37-06:00 baboon-srv sshd[6437]: last message repeated 2 times
		2011-04-26T18:57:37-06:00 baboon-srv sshd[6437]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:37-06:00 baboon-srv sshd[6439]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:39-06:00 baboon-srv sshd[6439]: Failed password for root from 172.30.1.77 port 60380 ssh2
		2011-04-26T18:57:43-06:00 baboon-srv sshd[6439]: last message repeated 2 times
		2011-04-26T18:57:43-06:00 baboon-srv sshd[6439]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:43-06:00 baboon-srv sshd[6441]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=root
		2011-04-26T18:57:46-06:00 baboon-srv sshd[6441]: Failed password for root from 172.30.1.77 port 60381 ssh2

i	Time	Event
		2011-04-26T19:02:09-06:00 baboon-srv sshd[6527]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:11-06:00 baboon-srv sshd[6527]: Failed password for bob from 172.30.1.77 port 49197 ssh2
		2011-04-26T19:02:15-06:00 baboon-srv sshd[6527]: last message repeated 2 times
		2011-04-26T19:02:15-06:00 baboon-srv sshd[6527]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:15-06:00 baboon-srv sshd[6529]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:18-06:00 baboon-srv sshd[6529]: Failed password for bob from 172.30.1.77 port 49198 ssh2
		2011-04-26T19:02:22-06:00 baboon-srv sshd[6529]: last message repeated 2 times
		2011-04-26T19:02:22-06:00 baboon-srv sshd[6529]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:22-06:00 baboon-srv sshd[6531]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:24-06:00 baboon-srv sshd[6531]: Failed password for bob from 172.30.1.77 port 49199 ssh2
		2011-04-26T19:02:28-06:00 baboon-srv sshd[6531]: last message repeated 2 times
		2011-04-26T19:02:28-06:00 baboon-srv sshd[6531]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:28-06:00 baboon-srv sshd[6533]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:31-06:00 baboon-srv sshd[6533]: Failed password for bob from 172.30.1.77 port 49200 ssh2
		2011-04-26T19:02:35-06:00 baboon-srv sshd[6533]: last message repeated 2 times
		2011-04-26T19:02:35-06:00 baboon-srv sshd[6533]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:36-06:00 baboon-srv sshd[6535]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:37-06:00 baboon-srv sshd[6535]: Failed password for bob from 172.30.1.77 port 49201 ssh2
		2011-04-26T19:02:41-06:00 baboon-srv sshd[6535]: last message repeated 2 times
		2011-04-26T19:02:41-06:00 baboon-srv sshd[6535]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:42-06:00 baboon-srv sshd[6537]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:44-06:00 baboon-srv sshd[6537]: Failed password for bob from 172.30.1.77 port 49202 ssh2
		2011-04-26T19:02:48-06:00 baboon-srv sshd[6537]: last message repeated 2 times
		2011-04-26T19:02:48-06:00 baboon-srv sshd[6537]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:48-06:00 baboon-srv sshd[6539]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:50-06:00 baboon-srv sshd[6539]: Failed password for bob from 172.30.1.77 port 49203 ssh2
		2011-04-26T19:02:54-06:00 baboon-srv sshd[6539]: last message repeated 2 times
		2011-04-26T19:02:54-06:00 baboon-srv sshd[6539]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:55-06:00 baboon-srv sshd[6541]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:02:57-06:00 baboon-srv sshd[6541]: Failed password for bob from 172.30.1.77 port 49204 ssh2
		2011-04-26T19:03:01-06:00 baboon-srv sshd[6541]: last message repeated 2 times
		2011-04-26T19:03:01-06:00 baboon-srv sshd[6541]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:03:01-06:00 baboon-srv sshd[6543]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=172.30.1.77 user=bob
		2011-04-26T19:03:03-06:00 baboon-srv sshd[6543]: Failed password for bob from 172.30.1.77 port 49205 ssh2
		2011-04-26T19:03:08-06:00 baboon-srv sshd[6543]: last message repeated 2 times
		Collapse
		COMMAND = /bin/su | PWD = /home/user1 | USER = root | euid = 0 | host = c5fb6d9ac6c2 | index = main | logname = LOGIN | source = auth.log | sourcetype = Auth.log

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

After several unsuccessful password attempt logs, we found a password-accepted log for the system 'baboon-srv'. Afterward, the attacker tried to escalate the privileges of the user using the sudo command which got failed. Sudo is widely used in Linux for running privileged commands.

>	8/5/23 1:05:10.000 AM	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 euid = 0 | host = c5fb6d9ac6c2 | index = main | logname = bob | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:04:33.000 AM	2011-04-26T19:04:33-06:00 baboon-srv sshd[6632]: pam_unix(sshd:session): session opened for user bob by (uid=0) host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:04:33.000 AM	2011-04-26T19:04:33-06:00 baboon-srv sshd[6632]: Accepted password for bob from 172.30.1.77 port 49215 ssh2 host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log

After further checking the logs, we got to know a few of the logs where the attacker tried to run some commands on the user's system as shown in the snapshot.

>	8/5/23 1:14:53.000 AM	2011-04-26T19:14:53-06:00 baboon-srv sshd[6632]: pam_unix(sshd:session): session closed for user bob host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:07:15.000 AM	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 COMMAND = /usr/bin/apt-get | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:07:03.000 AM	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 COMMAND = /usr/bin/apt-get | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:05:34.000 AM	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 COMMAND = /usr/sbin/tcpdump | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:05:18.000 AM	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 COMMAND = /usr/bin/vi | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log

The below log show that, the attacker got successful in executing the sudo command as an attacker used a text editor 'vi' to open authentication logs on the local server. It states that hacker edited the authentication logs stored locally.

>	8/5/23 1:05:10.000 AM	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 euid = 0 | host = c5fb6d9ac6c2 | index = main | logname = bob | source = auth.log | sourcetype = Auth.log

After that, the attacker ran the ‘tcpdump’ command, a utility that sniffs traffic on the local network. It is often used to troubleshoot network issues. The hacker has used network card eth0 to capture network packets on the "eth0".

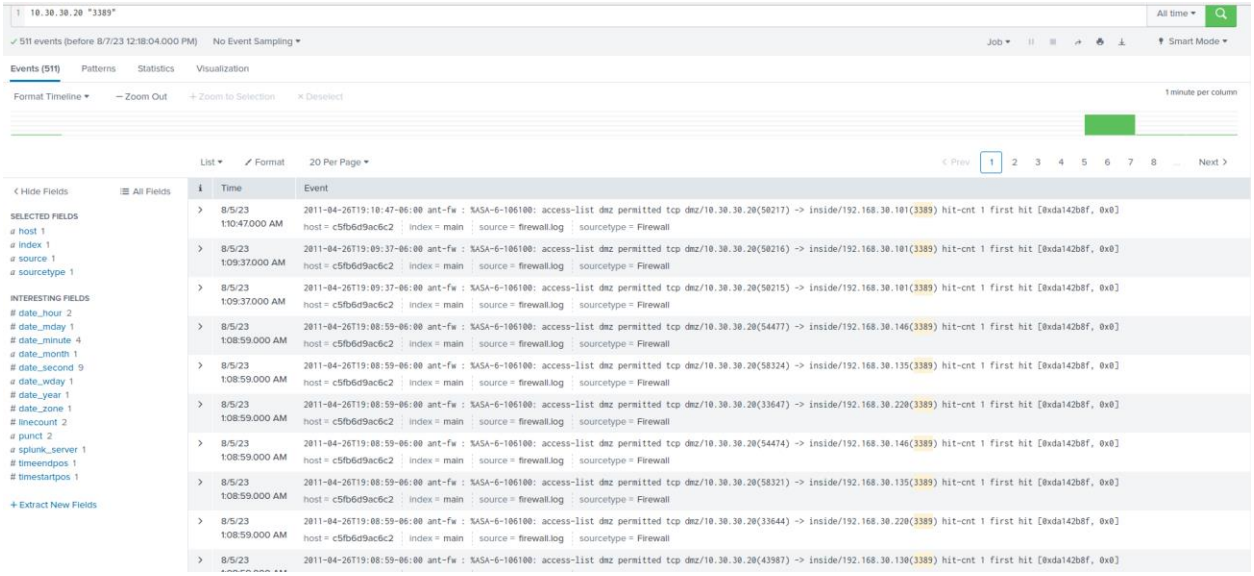
>	8/5/23 1:05:34.000 AM	2011-04-26T19:05:34-06:00 baboon-srv sudo: bob : TTY=pts/0 ; PWD=/home/bob ; USER=root ; COMMAND=/usr/sbin/tcpdump -nni eth0 COMMAND = /usr/sbin/tcpdump | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log

Afterward, the hacker updated the packages on system using ‘apt-get update’ command.

And have install Nmap using command ‘apt-get install nmap’. Nmap is a widely-used tool for network discovery and security scanning. It is possible that hackers might have installed nmap to run port scanning activity on the network and to check open ports.

>	8/5/23 1:07:15.000 AM	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 COMMAND = /usr/bin/apt-get | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log
>	8/5/23 1:07:03.000 AM	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 COMMAND = /usr/bin/apt-get | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log

After going through firewall logs, port scanning activity was detected from IP address 10.10.30.20 towards IP address 192.168.30.101 over port 3389 which is RDP (Remote Desktop Protocol) port. The IP address ’10.10.30.20’ seems to be user Bob’s IP address considering previously install nmap on his machine.



After checking the Windows logs, we found that there is a successful login for user bob on system ‘fox-ws’ which seems that hacker got successful in traversing through the network after compromising user bob’s system and now have gain access of system ‘fox-ws’. This events seems concerning as now hacker can traverse through the network and can gain access to sensitive data files.

```
1/15/19
12:52:58.000 AM
... 28 lines omitted ...
2011-04-26T18:56:04-06:00 fox-ws MSWinEventLog#0111#011Security#011447#011Tue Apr 26 18:56:02 2011#0114648#011Microsoft-Windows-Security-Auditing#011bob#011N/A#011Success Audit#011fox-
ws#011None#011#011A logon was attempted using explicit credentials.    Subject:   Security ID:  S-1-5-18   Account Name:  FOX-WS$   Account Domain:  WORKGROUP   Logon ID:  0x3e7   Logon GUID:  {0
0000000-0000-0000-0000-000000000000}    Account Whose Credentials Were Used:   Account Name:  bob   Account Domain:  fox-ws   Logon GUID:  {00000000-0000-0000-0000-000000000000}    Target Server:
   Target Server Name: localhost   Additional Information: localhost    Process Information:   Process ID:  0xae4   Process Name:  C:\Windows\System32\winlogon.exe    Network Information:   Netwo
rk Address: 127.0.0.1   Port:   0    This event is generated when a process attempts to log on an account by explicitly specifying that account\x92s credentials.  This most commonly occurs in bat
ch-type configurations such as scheduled tasks, or when using the RUNAS command.#011290
2011-04-26T18:56:04-06:00 fox-ws MSWinEventLog#0111#011Security#011448#011Tue Apr 26 18:56:02 2011#0114624#011Microsoft-Windows-Security-Auditing#011bob#011N/A#011Success Audit#011fox-
ws#011None#011#011An account was successfully logged on.    Subject:   Security ID:  S-1-5-18   Account Name:  FOX-WS$   Account Domain:  WORKGROUP   Logon ID:  0x3e7    Logon Type:   2    New Lo
gon:   Security ID:  S-1-5-21-29357171-1333843320-2140510157-1002   Account Name:  bob   Account Domain:  fox-ws   Logon ID:  0x66cee   Logon GUID:  {00000000-0000-0000-0000-000000000000}    Proc
ess Information:   Process ID:  0xae4   Process Name:  C:\Windows\System32\winlogon.exe    Network Information:   Workstation Name: FOX-WS   Source Network Address: 127.0.0.1   Source Port:  0
 Detailed Authentication Information:   Logon Process:  User32    Authentication Package: Negotiate   Transited Services: -   Package Name (NTLM only): -   Key Length:  0    This event is generat
ed when a logon session is created. It is generated on the computer that was accessed.    The subject fields indicate the account on the local system which requested the logon. This is most commo
nly a service such as the Server service, or a local process such as Winlogon.exe or Services.exe.    The logon type field indicates the kind of logon that occurred. The most common types are 2
(interactive) and 3 (network).    The New Logon fields indicate the account for whom the new logon was created, i.e. the account that was logged on.    The network fields indicate where a remote
logon request originated. Workstation name is not always available and may be left blank in some cases.    The authentication information fields provide detailed information about this specific l
ogon request.   - Logon GUID is a unique identifier that can be used to correlate this event with a KDC event.   - Transited services indicate which intermediate services have participated in thi
s logon request.   - Package name indicates which sub-protocol was used among the NTLM protocols.   - Key length in
... 4 lines omitted ...
2011-04-26T18:56:06-06:00 fox-ws MSWinEventLog#0111#011Security#011453#011Tue Apr 26 18:56:03 2011#0114673#011Microsoft-Windows-Security-Auditing#011bob#011N/A#011Failure Audit#011fox-
ws#011None#011#011A privileged service was called.    Subject:   Security ID:  S-1-5-21-29357171-1333843320-2140510157-1002   Account Name:  bob   Account Domain:  fox-ws   Logon ID:  0x2ddf0
Service:   Server: Security Account Manager   Service Name: Security Account Manager    Process:   Process ID: 0x1e8   Process Name: C:\Windows\System32\lsass.exe    Service Request Information:
  Privileges:  SeTcbPrivilege#011296
2011-04-26T18:56:08-06:00 fox-ws MSWinEventLog#0111#011Security#011454#011Tue Apr 26 18:56:05 2011#0114673#011Microsoft-Windows-Security-Auditing#011bob#011N/A#011Failure Audit#011fox-
ws#011None#011#011A privileged service was called.    Subject:   Security ID:  S-1-5-21-29357171-1333843320-2140510157-1002   Account Name:  bob   Account Domain:  fox-ws   Logon ID:  0x2ddf0
Service:   Server: Security Account Manager   Service Name: Security Account Manager    Process:   Process ID: 0x1e8   Process Name: C:\Windows\System32\lsass.exe    Service Request Information:
  Privileges:  SeTcbPrivilege#011297
2011-04-26T18:56:08-06:00 fox-ws MSWinEventLog#0111#011Security#011455#011Tue Apr 26 18:56:05 2011#0114634#011Microsoft-Windows-Security-Auditing#011bob#011N/A#011Success Audit#011fox-
ws#011None#011#011An account was logged off.    Subject:   Security ID:  S-1-5-21-29357171-1333843320-2140510157-1002   Account Name:  bob   Account Domain:  fox-ws   Logon ID:  0x66d06    Logon
Type:   2    This event is generated when a logon session is destroyed. It may be positively correlated with a logon event using the Logon ID value. Logon IDs are only unique between reboots on t
he same computer.#011298
```

2. Dynamic Malware Analysis

Scenario

Ann Dercover is after SaucyCorp's Secret Sauce recipe. She's been trailing the lead developer, Vick Timmes, to figure out how she can remotely access SaucyCorp's servers. She sees him log into his laptop (10.10.10.70) and VPN into SaucyCorp's headquarters. Leveraging her connections with international hacking organizations, Ann obtains an exploit for Internet Explorer and launches a client-side spear phishing attack against Vick Timmes. Ann carefully crafts an email to Vick containing tips on how to improve secret sauce recipes and sends it. Seeing an opportunity that could get him that Vice President of Product Development title (and corner office) that he's been coveting, Vick clicks on the link. long ago Vick Timmes set up a traffic monitoring system on his home network. When suspicious activity is discovered relating to Vick's account at SaucyCorp, he provides investigators with packet captures so they can help him identify a compromise.

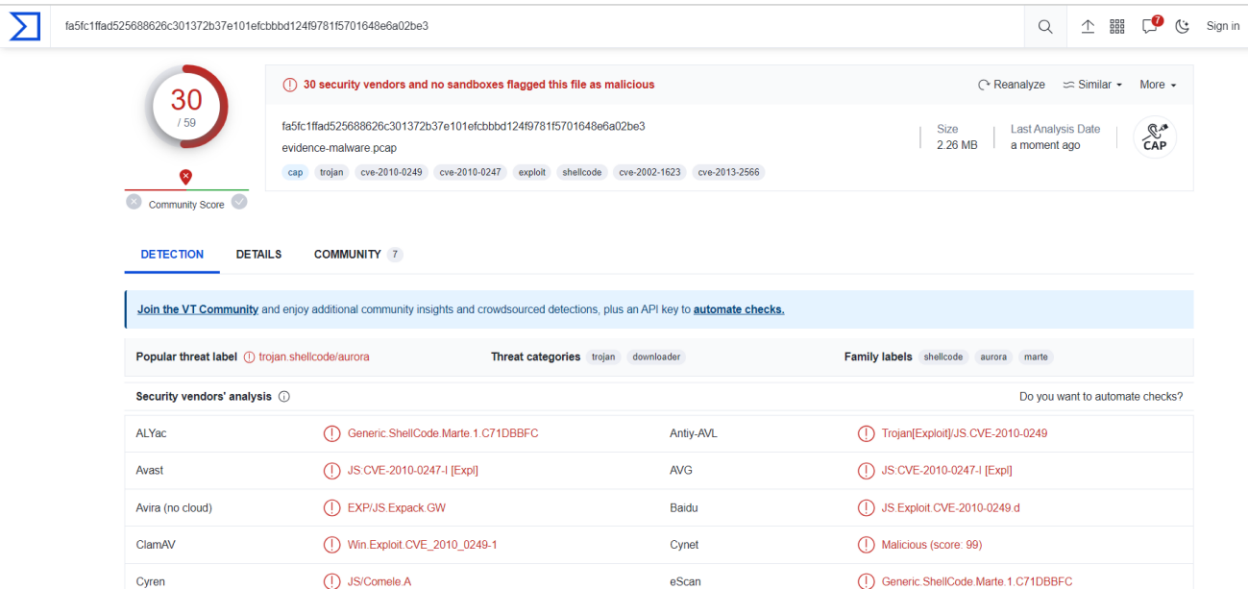
Network:

Vick Timmes's internal computer: 10.10.10.70

External host: 10.10.10.10 [Note that for the purposes of this case study, we are treating 10.10.10.70 as a system on "the Internet." In real life, this is a reserved nonroutable IP address space.]

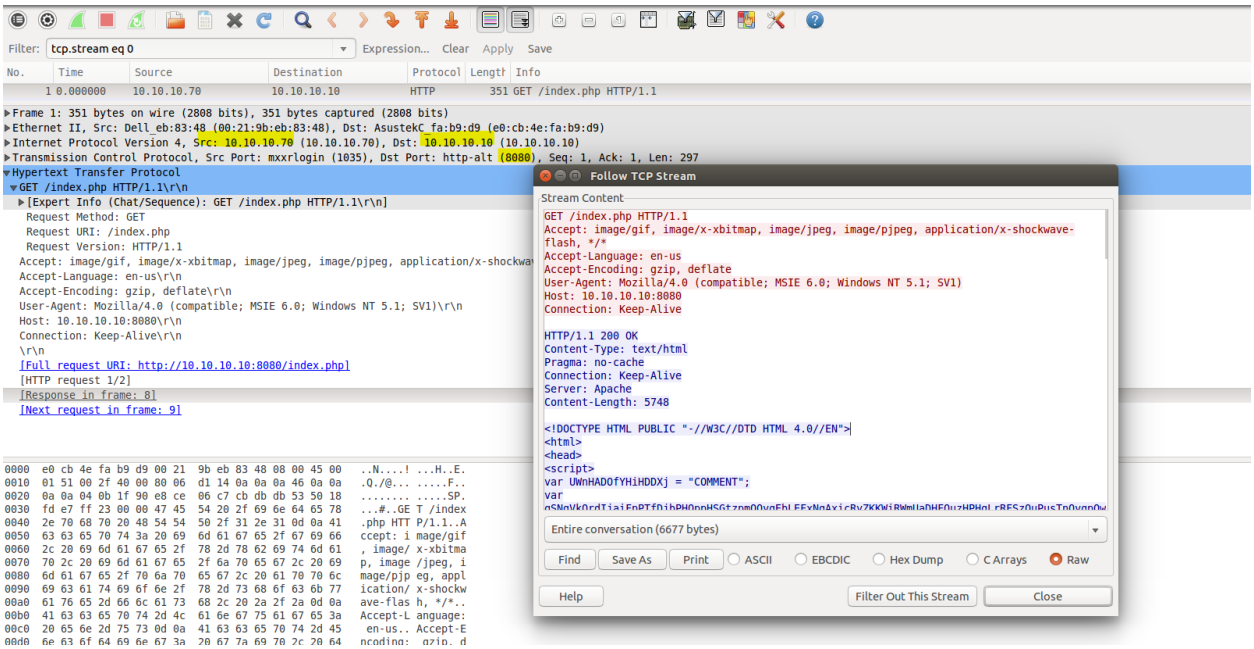
`1. Identify the source of the compromise.

To perform dynamic malware analysis on the pcap file, first I uploaded the file onto Virustotal. From the Virustotal result, we can see that 30/59 security vendors have detected the pcap file as malicious. Also, it is detected as a Trojan by the name 'trojan.shellcode/aurora'. From this, we can say that the pcap file has some malicious traffic.

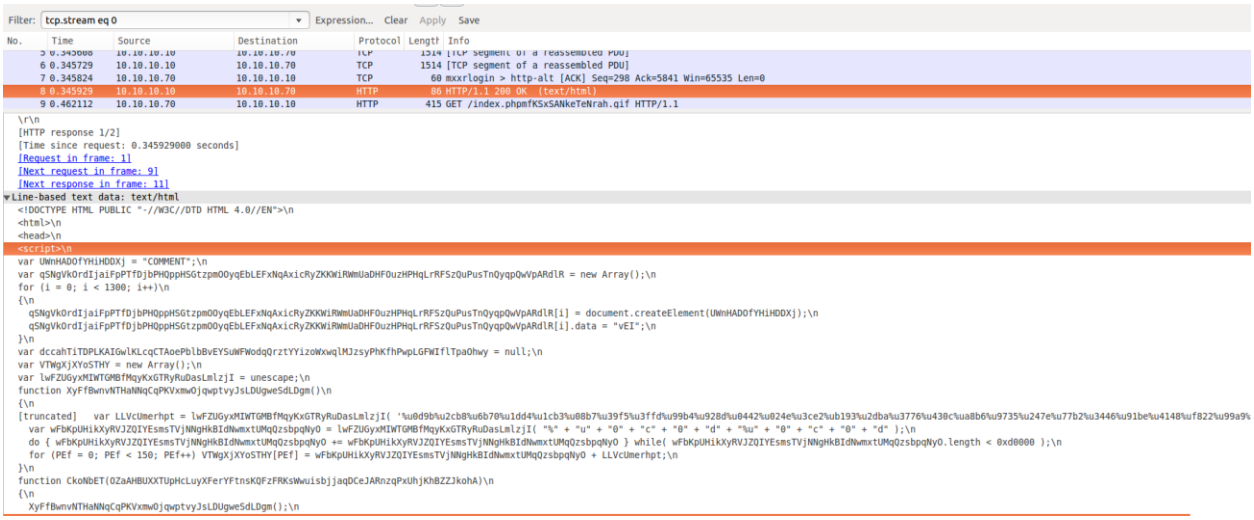


To further analyze pcap file, I have opened the file into Wireshark tools. After analyzing the packet capture, there is a GET request. It appears that 10:10:10:70 sent a "GET" request to

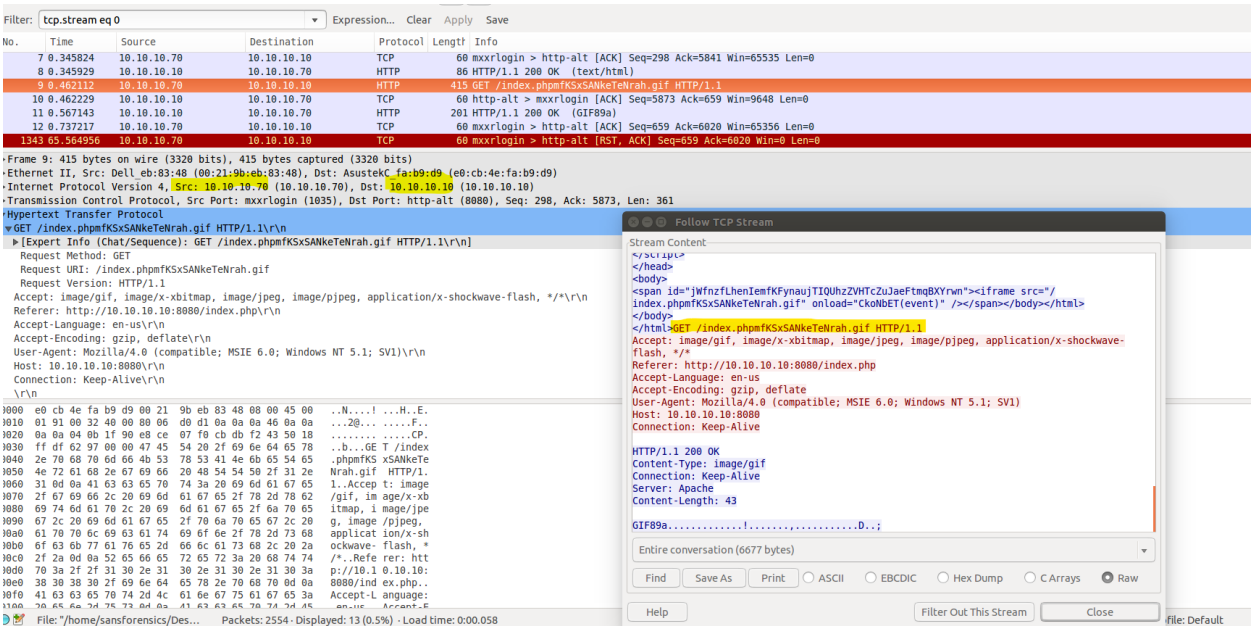
destination IP 10:10:10:10 on port 8080. **The source of the compromise was from 10.10.10.10 on port 8080.** GET request is used to retrieve a resource from the server. [5]



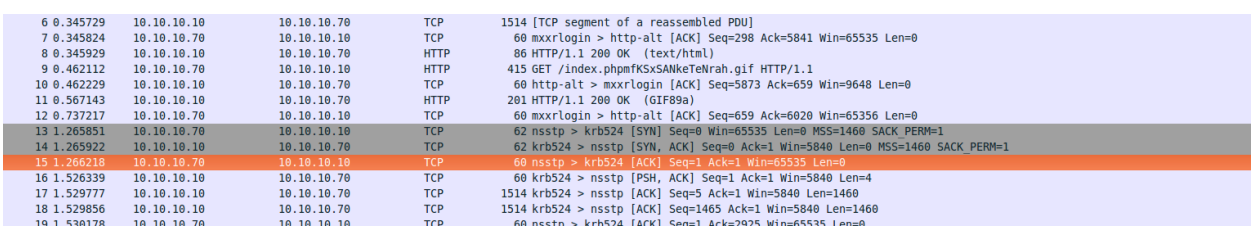
After the GET request the IP address 10:10:10:10 have responded with script which seems to be JavaScript content.



Once 10:10:10:10 responded, another “GET” request was sent for index.phpmfKSxSANkeTeHrah.gif. It looks as if the user on 10:10:10:70 was trying to retrieve a gif file from 10:10:10:10.



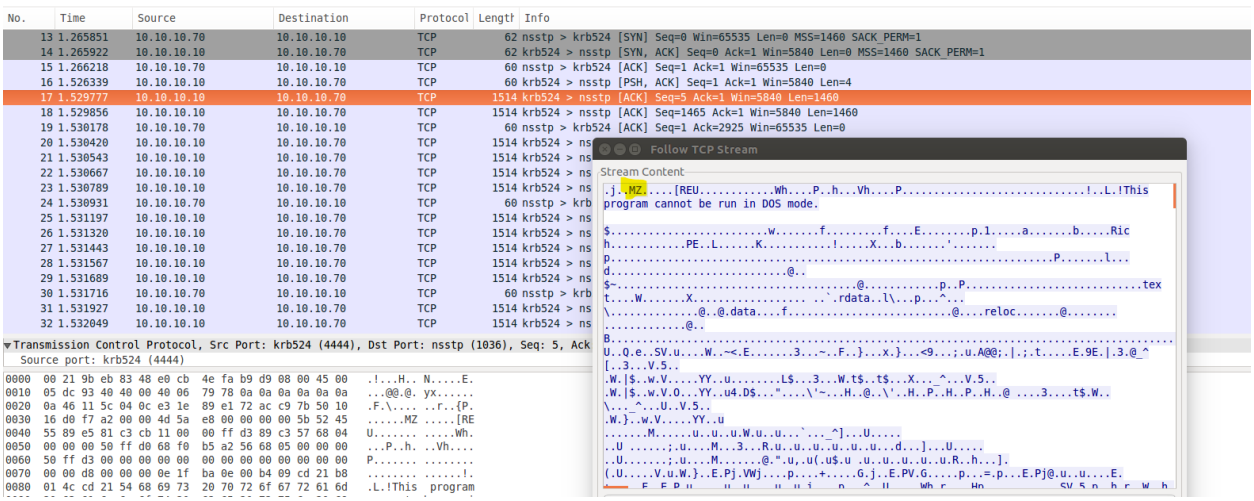
After receiving the .gif file, TCP traffic was seen between 10:10:10:10 and 10:10:10:70 with SYNs, SYN-ACK, and ACK. This is a full duplex connection. TCP uses a three-way handshake to establish a reliable connection. During this, Client requests connection by sending SYN (synchronize) message to the server. Server acknowledges by sending SYN-ACK (synchronize-acknowledge) message back to the client. Client responds with an ACK (acknowledge) message, and the connection is established. [2] (Williams, 2023)



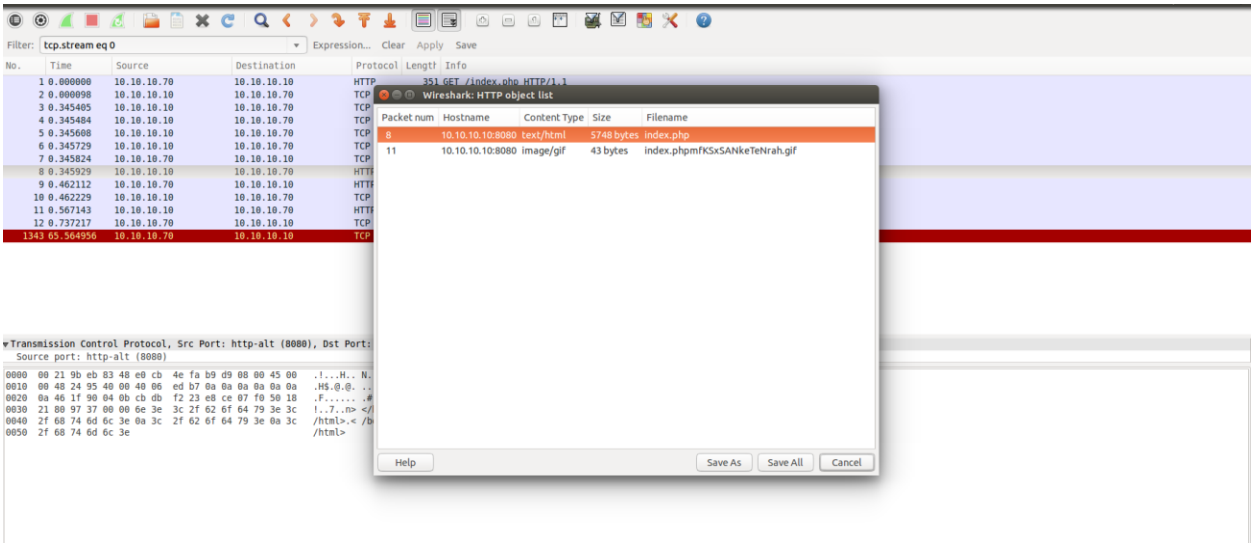
2. Recover malware from the packet capture and provide it to investigators for further analysis.

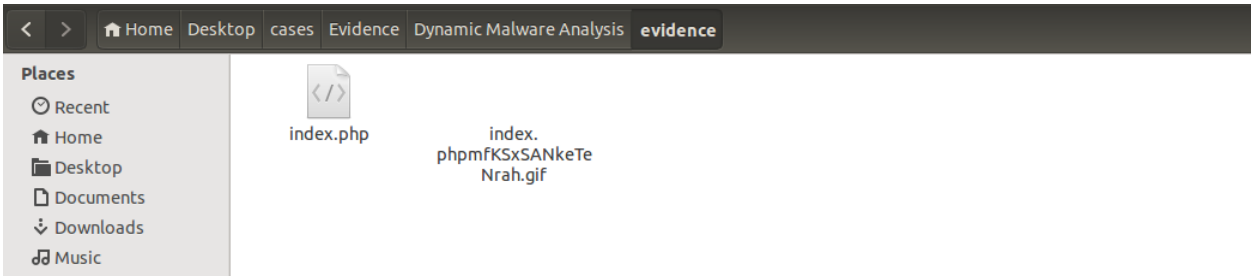
Further checking the ACK request and following the ‘TCP Stream’. The content isn't looks like .gif file. The GIF file extension starts with a specific signature, or "magic number," that helps identify the file type. For a GIF file, the first three bytes of the file will typically be 47 49 46, corresponding to the ASCII characters "GIF". The next three characters describe the version, which is usually "87a" or "89a", so a full GIF file header might start with the bytes representing "GIF87a" or "GIF89a". [3] (J., 2022)

The "MZ" signature at the beginning of the file indicates that it's in the MS-DOS executable file format.



You can export the two files (index.php, index.phpmfKSxSANkeTeNrah.gif) you have seen from the pcap file. For that, go to File → Export Objects → HTTP



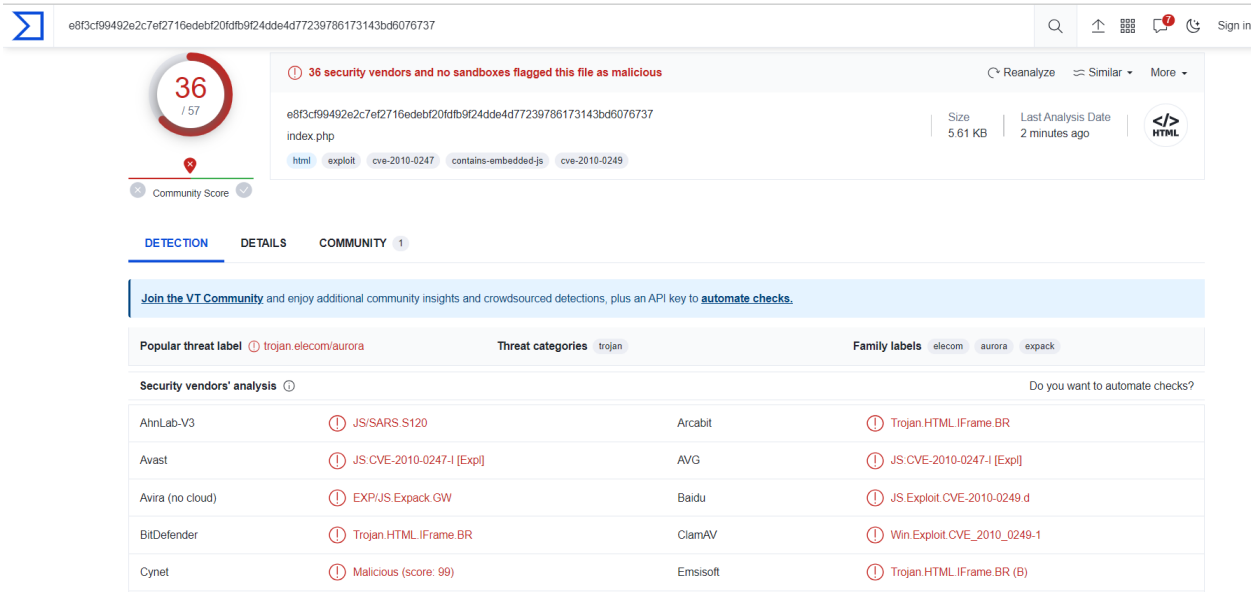


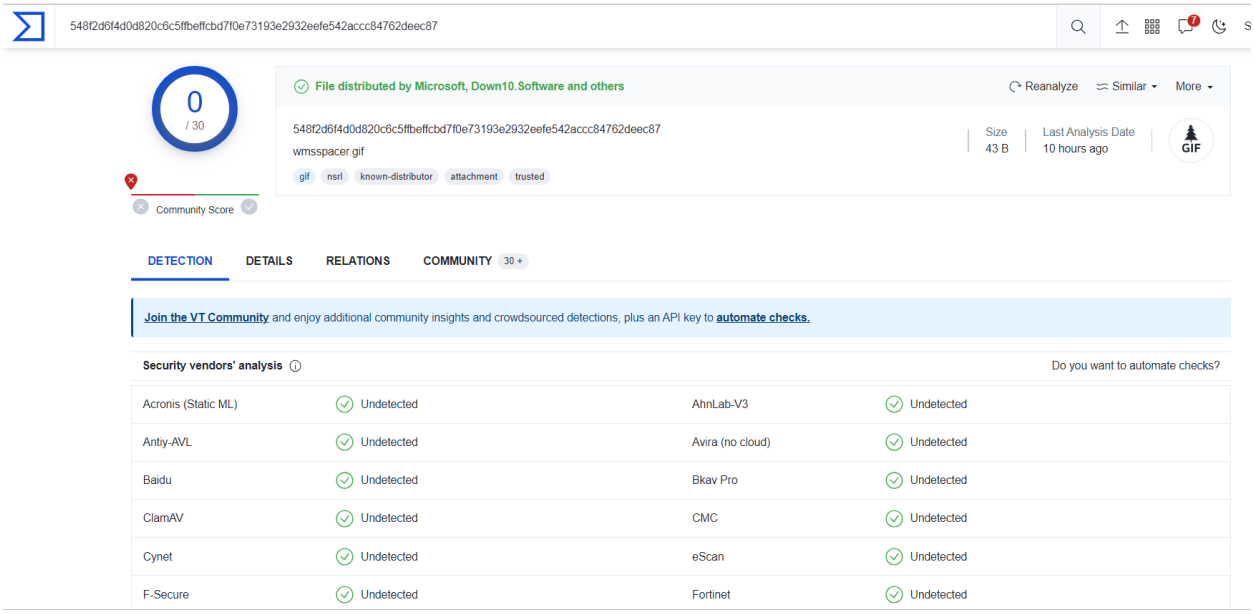
After checking the index.php on Virustotal it is detected as Trojan malware name ‘trojan.elecom/aurora’ and some of the threat vendors detected this Trojan as a ‘Win.Exploit.CVE_2010_0249-1’.

However, the second file (index.phpmfKSxSANkeTeNrah.gif) is not detected as malicious.

After searching more on this CVE we found that it is a security vulnerability identifier related to Microsoft Internet Explorer. This vulnerability allows remote attackers to execute arbitrary code on vulnerable installations of Microsoft Internet Explorer. [4] (Bureau, 2010)

It is concluded that Vick Timmes machine was vulnerable to this internet explorer vulnerability and when he clicked on the malicious link, the malicious javascript gets downloaded on his machine.





Citations –

1. Fortinet. (n.d.). Brute force attack. Retrieved August 5, 2023, from https://www.fortinet.com/resources/cyberglossary/brute-force-attack#:~:text=A%20brute%20force%20attack%20is,and%20organizations'%20systems%20and%20networks.

2. Williams, L. (2023, July 8). TCP 3-Way Handshake (SYN,SYN-ACK,ACK). Guru99. https://www.guru99.com/tcp-3-way-handshake.html
3. Alblas, J. (2022, June 21). TryHackMe: Upload Vulnerabilities Walkthrough. Medium. https://medium.com/@JAlblas/tryhackme-upload-vulnerabilities-walkthrough-32f7b2e555c3
4. Bureau, P.-M. (2010, January 25). Aurora exploit code: From targeted attacks to mass infection. WeLiveSecurity. https://www.welivesecurity.com/2010/01/25/aurora-exploit-code-from-targeted-attacks-to-mass-infection/
5. Olsen, A. (2023, February 3). The Fundamentals of HTTP for Hackers. TCM Security. https://tcm-sec.com/the-fundamentals-of-http-for-hackers/