

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

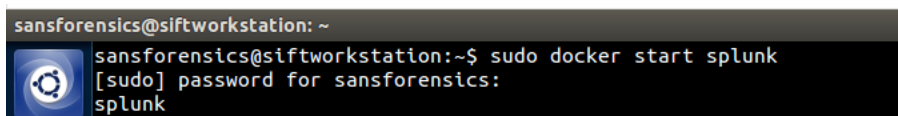
For this lab, we will be using a Linux machine, Splunk docker container, Wireshark, open source malware analysis tools like Virustotal

Analysis

1. Log Analysis

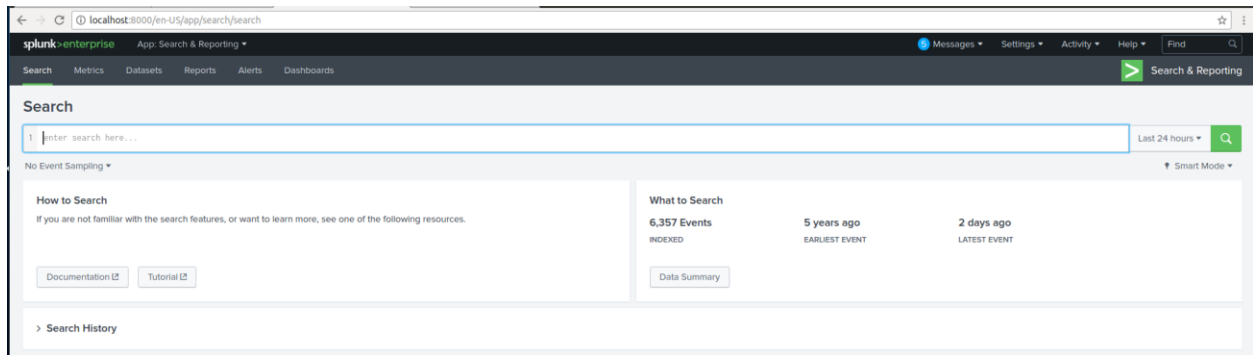
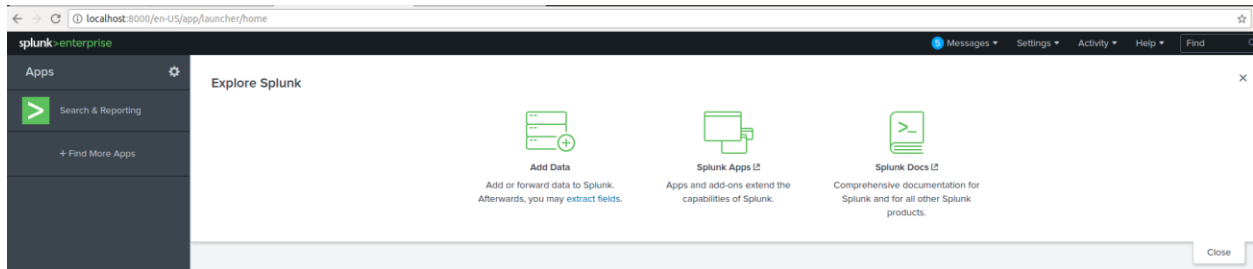
First, run the command – ‘sudo docker start splunk’. Once started, browse to Splunk using ‘localhost:8000’.

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.

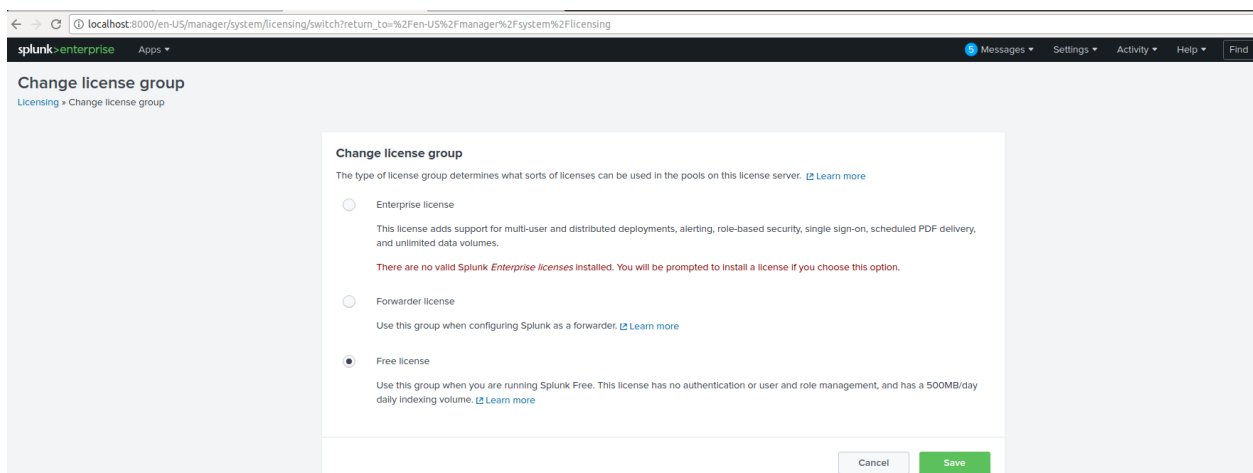


```
sansforensics@siftworkstation: ~  
sansforensics@siftworkstation:~$ sudo docker start splunk  
[sudo] password for sansforensics:  
splunk
```

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.log" 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.

The screenshot shows a Splunk search interface with the query `sourcetype=Auth.log "authentication failure"`. It displays 11 events. The table below represents the data shown in the 'List' view.

| Time | Event |
|-----------------------|--|
| 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=uid=0 euid=0 tty=/dev/pts/0 ruser=rhost= user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:04:05.000 AM | 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 euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:59.000 AM | 2011-04-26T19:03:59-06:00 baboon-srv sshd[6559]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:52.000 AM | 2011-04-26T19:03:52-06:00 baboon-srv sshd[6557]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:44.000 AM | 2011-04-26T19:03:44-06:00 baboon-srv sshd[6555]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:37.000 AM | 2011-04-26T19:03:37-06:00 baboon-srv sshd[6553]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:30.000 AM | 2011-04-26T19:03:30-06:00 baboon-srv sshd[6551]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:22.000 AM | 2011-04-26T19:03:22-06:00 baboon-srv sshd[6549]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob euid = 0 host = c5fb6d9ac6c2 index = main source = auth.log sourcetype = Auth.log |
| 8/5/23 1:03:15.000 AM | 2011-04-26T19:03:15-06:00 baboon-srv sshd[6547]: pam_unix(sshd:auth): authentication failure; logname=uid=0 euid=0 tty=ssh ruser=rhost=172.30.1.77 user=bob |

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

The screenshot shows a Splunk search interface with the query `sourcetype=Auth.log "authentication failure"`. It displays 11 events. The table below represents the data shown in the 'List' view.

| Time | Event |
|------------------------|---|
| 8/4/23 11:17:01.000 PM | 2011-04-26T17:17:01-06:00 baboon-srv CRON[6370]: pam_unix(cron:session): session opened for user root by (uid=0) |
| 8/4/23 11:17:01.000 PM | 2011-04-26T17:17:01-06:00 baboon-srv CRON[6370]: pam_unix(cron:session): session closed for user root |
| 8/4/23 11:17:01.000 PM | 2011-04-26T17:17:01-06:00 cheetah-srv CRON[1630]: pam_unix(cron:session): session opened for user root by (uid=0) |
| 8/4/23 11:17:01.000 PM | 2011-04-26T17:17:01-06:00 cheetah-srv CRON[1630]: pam_unix(cron:session): session closed for user root |
| 8/4/23 11:17:01.000 PM | 2011-04-26T18:17:01-06:00 baboon-srv CRON[6391]: pam_unix(cron:session): session opened for user root by (uid=0) |
| 8/4/23 11:17:01.000 PM | 2011-04-26T18:17:01-06:00 baboon-srv CRON[6391]: pam_unix(cron:session): session closed for user root |
| 8/4/23 11:17:01.000 PM | 2011-04-26T18:17:01-06:00 cheetah-srv CRON[8072]: pam_unix(cron:session): session opened for user root by (uid=0) |
| 8/4/23 11:17:01.000 PM | 2011-04-26T18:17:01-06:00 cheetah-srv CRON[8072]: pam_unix(cron:session): session closed for user root |
| 8/4/23 11:17:01.000 PM | 2011-04-26T19:17:01-06:00 cheetah-srv login[6427]: pam_unix(login:auth): check pass; user unknown |

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 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
1:05:10.000 AM 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 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
1:05:34.000 AM 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 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
1:07:15.000 AM COMMAND = /usr/bin/apt-get | PWD = /home/bob | USER = root | host = c5fb6d9ac6c2 | index = main | source = auth.log | sourcetype = Auth.log

> 8/5/23 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
1:07:03.000 AM 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.

| Time | Event |
|-----------------------|--|
| 8/5/23 1:04:47.000 AM | 2011-04-26T19:10:47-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(58217) -> inside/192.168.30.101(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:09:37.000 AM | 2011-04-26T19:09:37-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(58216) -> inside/192.168.30.101(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:09:37.000 AM | 2011-04-26T19:09:37-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(58215) -> inside/192.168.30.101(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(54477) -> inside/192.168.30.146(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(58324) -> inside/192.168.30.135(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(33647) -> inside/192.168.30.220(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(54474) -> inside/192.168.30.146(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(58321) -> inside/192.168.30.135(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(33644) -> inside/192.168.30.220(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |
| 8/5/23 1:08:59.000 AM | 2011-04-26T19:08:59-06:00 ant-fw : SASA-6-106100: access-list dmz permitted tcp dmz/10.30.20(43987) -> inside/192.168.30.130(3389) hit-cnt 1 first hit [0xda142b8f, 0x0] |

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 | ... 28 lines omitted ... |
| | 12:52:58.000 AM | <p>2011-04-26T18:56:04-06:00 fox-ws MSWinEventLog#011#001Security#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: {00000000-0000-0000-0000-000000000000} Account whose Credentials were Used: Account Name: bob Account Domain: fox-ws Logon GUID: {00000000-0000-0000-0000-000000000000} Target Server: localhost Target Server Name: localhost Additional Information: localhost Process Information: Process ID: 0xae4 Process Name: C:\Windows\System32\winlogon.exe Network Information: Network 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's credentials. This most commonly occurs in batch-type configurations such as scheduled tasks, or when using the RUNAS command.#011290</p> <p>2011-04-26T18:56:04-06:00 fox-ws MSWinEventLog#011#001Security#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 Logon: 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} Process 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 generated 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 commonly 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 logon 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 this logon request. - Package name indicates which sub-protocol was used among the NTLM protocols. - Key length in bytes.</p> <p>... 4 lines omitted ...</p> <p>2011-04-26T18:56:06-06:00 fox-ws MSWinEventLog#011#001Security#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</p> <p>2011-04-26T18:56:08-06:00 fox-ws MSWinEventLog#011#001Security#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</p> <p>2011-04-26T18:56:08-06:00 fox-ws MSWinEventLog#011#001Security#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: 0x66d96 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 the same computer.#011298</p> |

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.

fa5fc1ffad525688626c301372b37e101efcbbbd124f9781f5701648e6a02be3

30 / 59

30 security vendors and no sandboxes flagged this file as malicious

fa5fc1ffad525688626c301372b37e101efcbbbd124f9781f5701648e6a02be3
evidence-malware.pcap

Size: 2.26 MB | Last Analysis Date: a moment ago

cap trojan cve-2010-0249 cve-2010-0247 exploit shellcode cve-2002-1623 cve-2013-2566

Community Score

DETECTION DETAILS COMMUNITY 7

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to [automate checks](#).

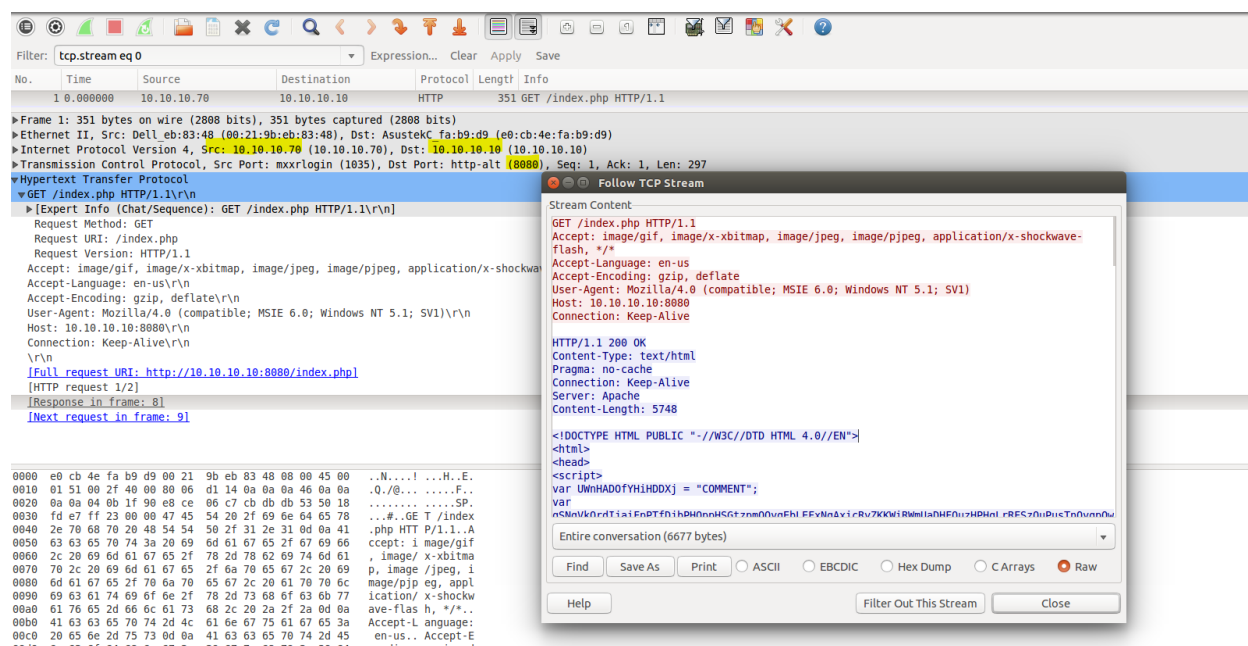
Popular threat label ① trojan.shellcode/aurora Threat categories trojan downloader Family labels shellcode aurora marte

Security vendors' analysis ① Do you want to automate checks?

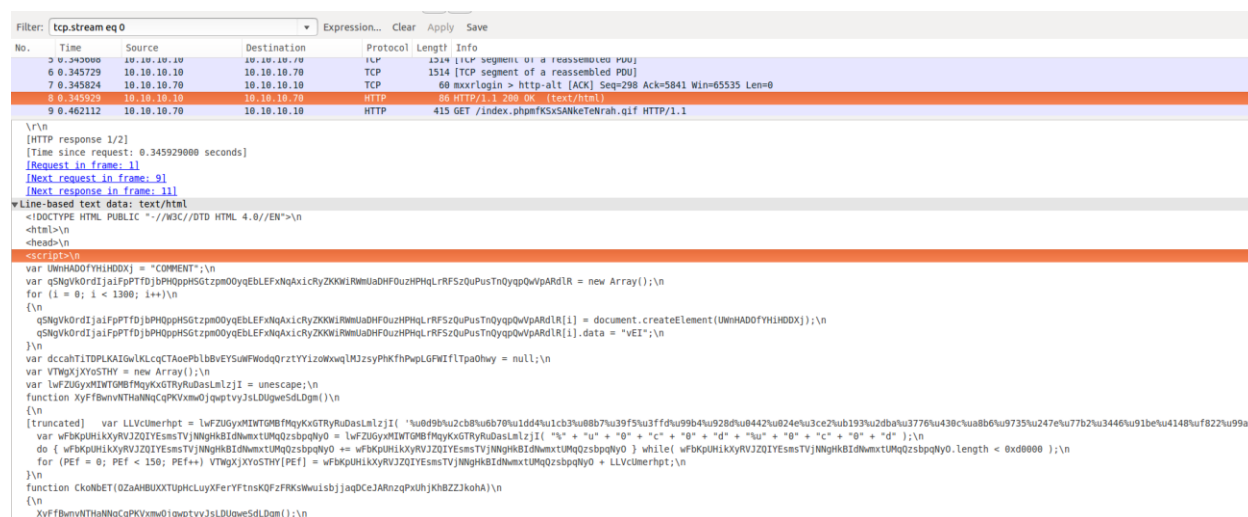
| | | | |
|------------------|--------------------------------------|----------|--------------------------------------|
| ALYac | ① Generic.ShellCode.Marte.1.C71DBBFC | Anty-AVL | ① Trojan(Exploit)JS.CVE-2010-0249 |
| Avast | ① JS.CVE-2010-0247-1 [Exploit] | AVG | ① JS.CVE-2010-0247-1 [Exploit] |
| Avira (no cloud) | ① EXPJS.Expack.GW | Baidu | ① JS.Exploit.CVE-2010-0249.d |
| ClamAV | ① Win.Exploit.CVE_2010_0249-1 | Cyren | ① Malicious (score: 99) |
| Cyren | ① JS/Comele.A | eScan | ① Generic.ShellCode.Marte.1.C71DBBFC |

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.

The image shows a Wireshark packet capture of a network conversation. The top pane displays a list of packets, with packet 9 (42112) highlighted, showing a GET request for index.phpmfKSxSANkeTeHrah.gif. The middle pane shows the details of this packet, including the request method (GET), URL, version (HTTP/1.1), and various headers like Accept, Accept-Encoding, User-Agent, Host, and Connection. The bottom pane shows the raw packet data in hexadecimal and ASCII. A 'Follow TCP Stream' window is open, showing the stream content, which includes the HTML response from the server, indicating a successful GET request for the specified GIF file.

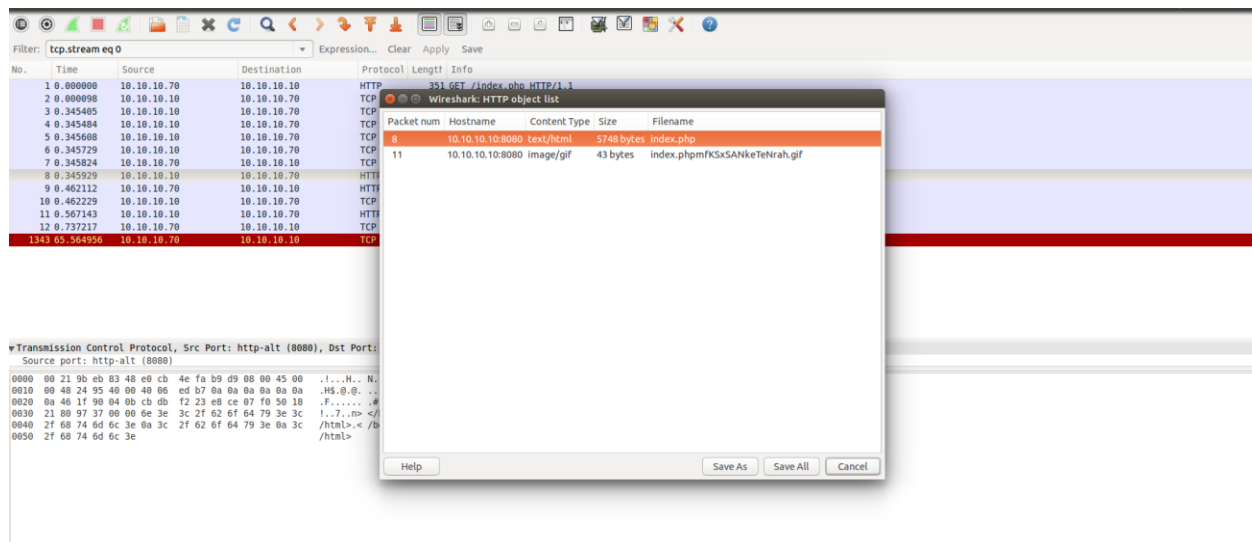
After receiving the .gif file, TCP traffic was seen between 10:10:10:10 and 10:10:10:70 with SYNs, SYN-ACK, and ACK. This is a full duplex connection. TCP uses a three-way handshake to establish a reliable connection. During this, Client requests connection by sending SYN (synchronize) message to the 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)

| | | | | | |
|----|----------|-------------|-------------|------|--|
| 6 | 0.345729 | 10.10.10.10 | 10.10.10.70 | TCP | 1514 [TCP segment of a reassembled PDU] |
| 7 | 0.345824 | 10.10.10.70 | 10.10.10.10 | TCP | 60 mxrlogin > http-alt [ACK] Seq=298 Ack=5841 Win=65535 Len=0 |
| 8 | 0.345929 | 10.10.10.10 | 10.10.10.70 | HTTP | 86 HTTP/1.1 200 OK (text/html) |
| 9 | 0.462112 | 10.10.10.70 | 10.10.10.10 | HTTP | 415 GET /index.phpmfKSxSANkeTeHrah.gif HTTP/1.1 |
| 10 | 0.462229 | 10.10.10.10 | 10.10.10.70 | TCP | 60 http-alt > mxrlogin [ACK] Seq=5873 Ack=659 Win=9648 Len=0 |
| 11 | 0.567143 | 10.10.10.10 | 10.10.10.70 | HTTP | 201 HTTP/1.1 200 OK (GIF89a) |
| 12 | 0.737217 | 10.10.10.70 | 10.10.10.10 | TCP | 60 mxrlogin > http-alt [ACK] Seq=659 Ack=6020 Win=65356 Len=0 |
| 13 | 1.265851 | 10.10.10.70 | 10.10.10.10 | TCP | 62 nsstp > krb524 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1 |
| 14 | 1.265922 | 10.10.10.10 | 10.10.10.70 | TCP | 62 krb524 > nsstp [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1 |
| 15 | 1.266218 | 10.10.10.70 | 10.10.10.10 | TCP | 60 nsstp > krb524 [ACK] Seq=1 Ack=1 Win=65535 Len=0 |
| 16 | 1.526339 | 10.10.10.10 | 10.10.10.70 | TCP | 60 krb524 > nsstp [PSH, ACK] Seq=1 Ack=1 Win=5840 Len=4 |
| 17 | 1.529777 | 10.10.10.10 | 10.10.10.70 | TCP | 1514 krb524 > nsstp [ACK] Seq=5 Ack=1 Win=5840 Len=1460 |
| 18 | 1.529856 | 10.10.10.10 | 10.10.10.70 | TCP | 1514 krb524 > nsstp [ACK] Seq=1465 Ack=1 Win=5840 Len=1460 |
| 19 | 1.530178 | 10.10.10.70 | 10.10.10.10 | TCP | 60 nsstp > krb524 [ACK] Seq=1 Ack=1 Win=65535 Len=0 |

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

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.

36

/ 57

Community Score

ⓘ

36 security vendors and no sandboxes flagged this file as malicious

Reanalyze Similar More

e8f3cf9492e2c7ef2716edebf20fdb9f24dde4d77239786173143bd6076737

index.php

Size: 5.61 KB | Last Analysis Date: 2 minutes ago

HTML

html exploit cve-2010-0247 contains-embedded-js cve-2010-0249

DETECTION

DETAILS

COMMUNITY 1

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to [automate checks](#).

Popular threat label ⓘ trojan.elecom/aurora

Threat categories trojan

Family labels elecom aurora expack

Security vendors' analysis ⓘ

Do you want to automate checks?

| | | | |
|------------------|--------------------------------|----------|--------------------------------|
| AhnLab-V3 | ⓘ JS/SARS.S120 | Arcabit | ⓘ Trojan.HTML.IFrame.BR |
| Avast | ⓘ JS.CVE-2010-0247-I [Exploit] | AVG | ⓘ JS.CVE-2010-0247-I [Exploit] |
| Avira (no cloud) | ⓘ EXP/JS.Expact.GW | Baidu | ⓘ JS.Exploit.CVE-2010-0249.d |
| BitDefender | ⓘ Trojan.HTML.IFrame.BR | ClamAV | ⓘ Win.Exploit.CVE_2010_0249-1 |
| Cynet | ⓘ Malicious (score: 99) | Emsisoft | ⓘ Trojan.HTML.IFrame.BR (B) |

0

/ 30

Community Score

✔

File distributed by Microsoft, Down10.Software and others

Reanalyze Similar More

548f2d6f4dd820c6c5ffbeffc710e73193e2932eeffe542acc84762deec87

wmsspace.gif

Size: 43 B | Last Analysis Date: 10 hours ago

GIF

gif nsrl known-distributor attachment trusted

DETECTION

DETAILS

RELATIONS

COMMUNITY 30 +

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to [automate checks](#).

Security vendors' analysis ⓘ

Do you want to automate checks?

| | | | |
|---------------------|--------------|------------------|--------------|
| Acronis (Static ML) | ✔ Undetected | AhnLab-V3 | ✔ Undetected |
| Antiy-AVL | ✔ Undetected | Avira (no cloud) | ✔ Undetected |
| Baidu | ✔ Undetected | Bkav Pro | ✔ Undetected |
| ClamAV | ✔ Undetected | CMC | ✔ Undetected |
| Cynet | ✔ Undetected | eScan | ✔ Undetected |
| F-Secure | ✔ Undetected | Fortinet | ✔ Undetected |

Citations –

- 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/>