بخش اول - ابزارنویسی

ابزار ping

برای نوشتن ابزار ping، از کتابخانهٔ scapy کمک می‌گیریم. کد این ابزار به ۳ تابع اصلی تقسیم شده است.

- تابع resolve_host: این تابع آدرس آی‌پی دامنهٔ ورودی برنامه را به دست می‌آورد.
- تابع ping: این تابع یک آدرس آی‌پی مقصد به عنوان ورودی دریافت کرده، و یک پکت ICMP به آن ارسال می‌کند. در صورتی که بسته با موفقیت پاسخ بگیرد، rtt آن را به عنوان خروجی برمی‌گرداند.
- تابع print_statistics: این تابع لیستی از rtt پکت‌های ارسالی به عنوان ورودی گرفته، و آمارهای مربوطه، اعم از کمینه،میانگین، بیشینه و انحراف معیار rttها، و همچنین نرخ پکت‌های از دست رفته را چاپ می‌کند.

```
(venv) ┌─ ~/PycharmProjects/info-sec-project/src ──────────────────────── DanialH danials-MacBook-Pro-2:s
003 ─┐
└─(18:57:15)──> sudo python3 ping.py                                                    ──(Fri,Nov04)─┘
WARNING: No IPv4 address found on en5 !
WARNING: No IPv4 address found on en1 !
WARNING: more No IPv4 address found on en2 !
Please Enter Your IP/Domain: 1.1.1.1
PING 1.1.1.1 (1.1.1.1): 56 data bytes
28 bytes from 1.1.1.1: icmp_seq=0 ttl=59 time=138.386ms
28 bytes from 1.1.1.1: icmp_seq=1 ttl=59 time=194.915ms
28 bytes from 1.1.1.1: icmp_seq=2 ttl=59 time=194.911ms
28 bytes from 1.1.1.1: icmp_seq=3 ttl=59 time=139.127ms
28 bytes from 1.1.1.1: icmp_seq=4 ttl=59 time=163.355ms
28 bytes from 1.1.1.1: icmp_seq=5 ttl=59 time=188.274ms
28 bytes from 1.1.1.1: icmp_seq=6 ttl=59 time=401.12ms
28 bytes from 1.1.1.1: icmp_seq=7 ttl=59 time=221.311ms
28 bytes from 1.1.1.1: icmp_seq=8 ttl=59 time=171.42ms
28 bytes from 1.1.1.1: icmp_seq=9 ttl=59 time=195.85ms
28 bytes from 1.1.1.1: icmp_seq=10 ttl=59 time=201.013ms
28 bytes from 1.1.1.1: icmp_seq=11 ttl=59 time=195.544ms
28 bytes from 1.1.1.1: icmp_seq=12 ttl=59 time=200.96ms
^C28 bytes from 1.1.1.1: icmp_seq=13 ttl=59 time=199.526ms
--- 1.1.1.1 ping statistics ---
14 transmitted, 14 packets received, 0% packet loss
round-trip min/avg/max/stddev = 138.386/200.408/401.12/62.506 ms
```

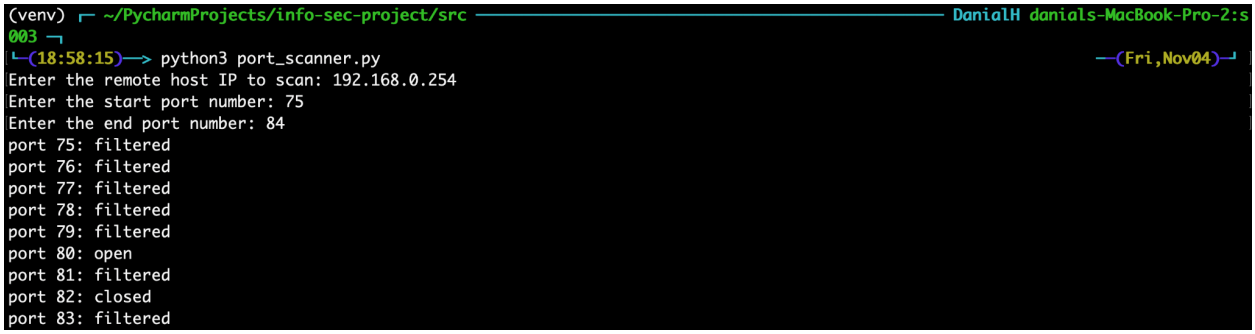
ابزار host_scanner

برای این ابزار، از کتابخانهٔ nmap استفاده می‌کنیم. ابتدا با تابع generate_ips_in_range لیستی از تمام آی‌پی‌های بازهٔ شروع و پایان ایجاد می‌کنیم. سپس با فراخوانی تابع nmap.Scanner.scan با آرگومان "-sP" هر یک از آی‌پی‌ها را بررسی می‌کنیم. آرگومان -sP یا برای چشم‌پوشی از اسکن کردن پورت‌ها پس از گام Host Discoveryـست.

```
(venv) ┌─ ~/PycharmProjects/info-sec-project/src ──────────────────────── DanialH danials-MacBook-Pro-2:s
003 ─┐
└─(22:25:16)──> python3 host_scanner.py                                                 ──(Fri,Nov04)─┘
Enter the start ip: 89.43.3.20
Enter the end ip: 89.43.3.30
89.43.3.20 --> up
89.43.3.21 --> up
89.43.3.22 --> up
89.43.3.23 --> up
89.43.3.24 --> up
89.43.3.25 --> up
89.43.3.26 --> up
89.43.3.27 --> up
89.43.3.28 --> up
89.43.3.29 --> up
```

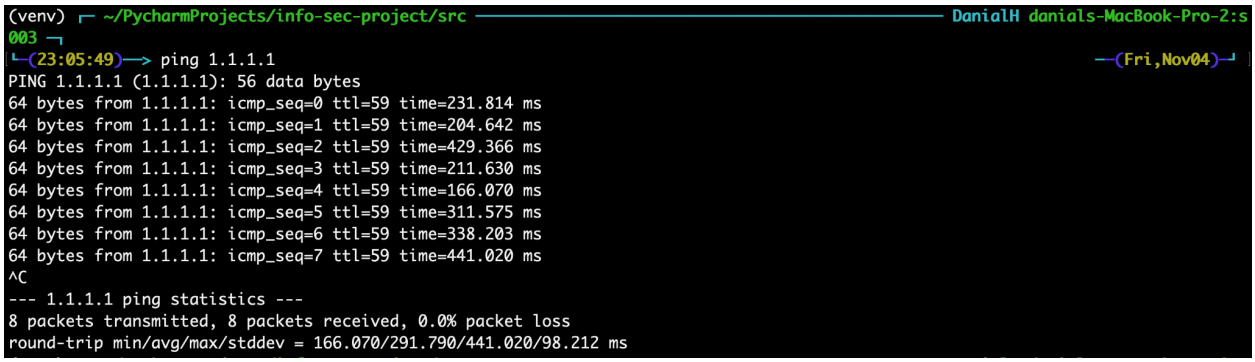
ابزار port_scanner

برای این ابزار از کتابخانهٔ nmap و به طور خاص کلاس nmap.PortScanner استفاده می‌کنیم. با فراخوانی تابع scan روی این کلاس، با آرگومان‌های آدرس آی‌پی مقصد و بازهٔ پورت‌ها، می‌توانیم وضعیت پورت‌های مورد نظر را چاپ کنیم.



بخش دوم - کار با ابزارهای آماده

ابزار ping (مقایسه با ابزار نوشته شده)



ابزار nmap

یافتن Hostهای فعال یک رنج آی‌پی:

- اسکن Ping:
  - آرگومان: -sP
  - شرح: این اسکن به طور پیش‌فرض از یک درخواست ICMP Echo، یک درخواست TCP SYN به پورت ۴۴۳ و TCP ACK به پورت ۸۰ و ICMP Timestamp تشکیل می‌شود.

```
└─(23:34:23)──> nmap -sP 89.43.3.20-30                                   ──(جمعه,نوامبر04)─┘
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-04 23:35 +0330
Nmap scan report for 20.mobinnet.net (89.43.3.20)
Host is up (0.26s latency).
Nmap scan report for 21.mobinnet.net (89.43.3.21)
Host is up (0.18s latency).
Nmap scan report for 22.mobinnet.net (89.43.3.22)
Host is up (0.31s latency).
Nmap scan report for 23.mobinnet.net (89.43.3.23)
Host is up (0.31s latency).
Nmap scan report for 24.mobinnet.net (89.43.3.24)
Host is up (0.31s latency).
Nmap scan report for 25.mobinnet.net (89.43.3.25)
Host is up (0.31s latency).
Nmap scan report for 26.mobinnet.net (89.43.3.26)
Host is up (0.31s latency).
Nmap scan report for 27.mobinnet.net (89.43.3.27)
Host is up (0.31s latency).
Nmap scan report for 28.mobinnet.net (89.43.3.28)
Host is up (0.31s latency).
Nmap scan report for 29.mobinnet.net (89.43.3.29)
Host is up (0.31s latency).
Nmap scan report for 30.mobinnet.net (89.43.3.30)
Host is up (0.31s latency).
Nmap done: 11 IP addresses (11 hosts up) scanned in 1.21 seconds
```

- [اسکن TCP full]:

آرگومان: ‎-sT

شرح: زمانی که کاربر اجراکننده دسترسی ارسال پکت‌های دلخواه را نداشته باشد، یا در حال اسکن شبکه‌ای با IPv6 باشد، از این اسکن استفاده می‌شود. در این نوع اسکن، به جای ساخت پکت‌ها مستقیم توسط nmap، سیستم‌کال connect سیستم عامل فراخوانی می‌شود.

```
┌─(~)──────────────────────────────────────────────────────(divar@divar-ThinkPad-E14:pts/0)─┐
└─(23:36:38)──> nmap -sT 89.43.3.20                            130 ↵ ──(جمعه,نوامبر04)─┘
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-04 23:36 +0330
Nmap scan report for 20.mobinnet.net (89.43.3.20)
Host is up (0.16s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 13.93 seconds
```

همان‌طور که مشاهده می‌شود، برای آدرس 89.43.3.20 پورت‌های ۸۰، ۴۴۳ و ۸۰۸۰ باز هستند.

- [اسکن Stealth]

آرگومان: ‎-sS

شرح: این اسکن ارتباط TCP نیمه‌باز با پورت‌های مختلف سیستم مقصد ایجاد می‌کند تا پورت‌های باز آن را شناسایی کند. به دلیل کامل نکردن ارتباط TCP (چشم‌پوشی از گام آخر - ACK) سریع است. به طور پیش‌فرض nmap این‌گونه اسکن می‌کند.

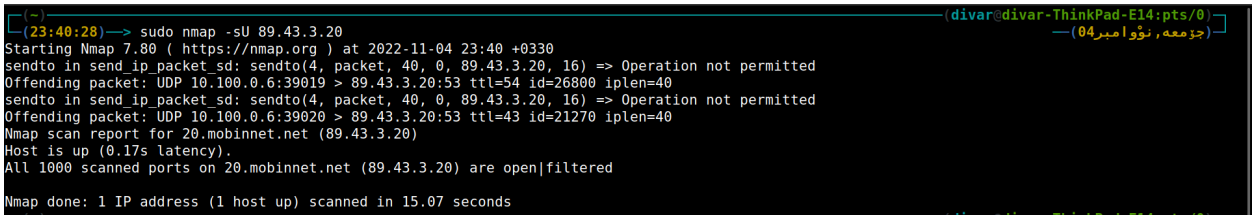
```
┌─(~)──────────────────────────────────────────────────────(divar@divar-ThinkPad-E14:pts/0)─┐
└─(23:38:06)──> sudo nmap -sS 89.43.3.20                         1 ↵ ──(جمعه,نوامبر04)─┘
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-04 23:38 +0330
sendto in send_ip_packet_sd: sendto(4, packet, 44, 0, 89.43.3.20, 16) => Operation not permitted
Offending packet: TCP 10.100.0.6:33832 > 89.43.3.20:53 S ttl=44 id=12210 iplen=44  seq=2777377991 win=1024 <mss 1460>
sendto in send_ip_packet_sd: sendto(4, packet, 44, 0, 89.43.3.20, 16) => Operation not permitted
Offending packet: TCP 10.100.0.6:33833 > 89.43.3.20:53 S ttl=42 id=48704 iplen=44  seq=2777312454 win=1024 <mss 1460>
Nmap scan report for 20.mobinnet.net (89.43.3.20)
Host is up (0.18s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 18.17 seconds
```

این نوع اسکن هم این ۳ پورت باز تشخیص داده و ۹۹۷ پورت بسته هم امتحان شده‌اند. این اسکن ۱۸ ثانیه طول کشیده است.

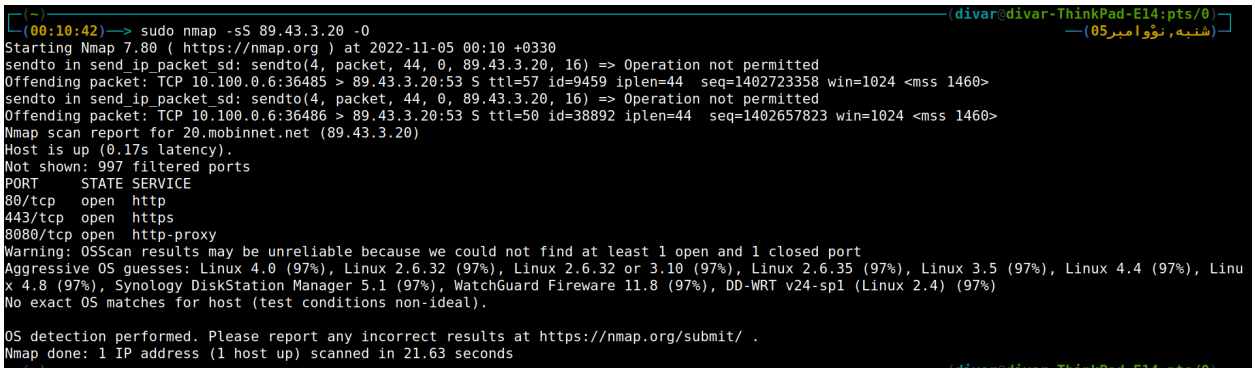
- اسکن UDP:

آرگومان: -sU

شرح: با این آرگومان، nmap برای ارتباط UDP با پورت‌های مختلف مقصد تلاش می‌کند. این اسکن برای شناسایی وجود یا عدم وجود سرویس‌های از نوع UDP مانند سرویس DNS استفاده می‌شود. از اسکن TCP کندتر است.



- اسکن Fingerprint:
  - آرگومان: -O
  - شرح: nmap می‌تواند با استفاده از TCP/IP Fingerprinting مواردی چون OS دستگاه مقصد را حدس بزند.



در مثال بالا، سیستم عامل‌های لینوکس با نسخه‌های مختلف با درصد اطمینان‌های متفاوت حدس زده شده‌اند.

- اسکن Idle:

آرگومان: -sI

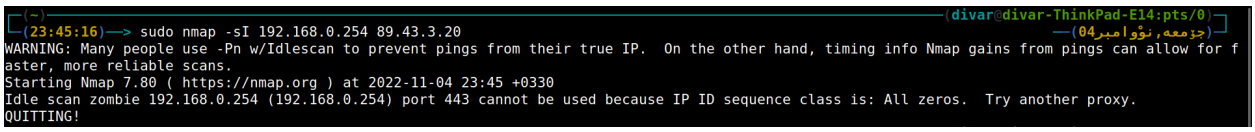
شرح: هر پکت IP در اینترنت یک Identification Number در هدر خود دارد. بسیاری از سیستم‌عامل‌ها برای هر پکت ارسالی خود، این ID را یک واحد زیاد می‌کنند. به این روش ساخت IPID Sequence، روش Incremental می‌گویند.

در اسکن Idle، مهاجم از یک دستگاه سوم، که Zombie نامیده می‌شود برای اسکن کردن سیستم مقصد (Target) استفاده می‌کند.

دستگاه Zombie باید حتماً بی‌استفاده باشد. مهاجم ابتدا یک ارتباط با Zombie برقرار کرده، و IP ID فعلی آن را ذخیره می‌کند. سپس یک درخواست SYN به پورتی خاص و با آی‌پی Zombie به Target ارسال می‌کند. در صورت باز بودن پورت مورد نظر Target به این SYN با SYN-ACK جواب می‌دهد.

مهاجم دوباره IP ID ـ Zombie را گرفته و ذخیره می‌کند. در صورتی که این مقدار ۲ واحد زیاد شده باشد، یعنی ارتباط با Target برقرار شده و آن پورت باز بوده است. در این روش مهاجم اصلاً آی‌پی خود را به Target بروز نمی‌دهد.

در تصویر بالا، IPIDـی که مهاجم از Zombie ابتدا گرفته، 31337 است. در مرحله‌ی دوم، مهاجم از طرف Zombie، یک درخواست SYN به هدف ارسال می‌کند. در صورتی که هدف پاسخ‌گو باشد، یک بسته‌ی SYN/ACK به Zombie ارسال می‌کند. Zombie چون از ابتدا این ارتباط را شروع نکرده، این درخواست را با RST (با IPID 31338) پاسخ می‌دهد. در این صورت، هنگامی که مهاجم دوباره ارتباط با Zombie برقرار کند، بسته‌ی RST با IPID 31339 دریافت می‌کند. از این که IPID به مقدار دو واحد از مقدار ذخیره شده بالاتر رفته، مهاجم می‌تواند نتیجه بگیرد که ارتباط با هدف موفقیت‌آمیز بوده، و پورت مورد نظر در هدف باز بوده است.



این اسکن نیاز به یک Zombie Host دارد، تا به کمک آن به آدرس Target درخواست بزند. در تصویر بالا، آدرس 192.168.0.254 را به عنوان Zombie معرفی کردیم، اما چون پورت آن بسته است، امکان تکمیل این روش نبود.

برای یافتن Zombie، ابزار NMAP دو راه در اختیار ما می‌گذارد. قابلیت OS Detection این ابزار که با آرگومان -O مشخص می‌شود، به همراه آرگومان -v (آرگومان verbose) می‌تواند الگوریتم Host هدف برای IPID Sequence را حدس بزند. اگر این الگوریتم Incremental یا Broken little-endian incremental باشد، Host مورد نظر یک Zombie بالقوه است.
راه دیگر استفاده از اسکریپت ipidseq.nse ابزار NMAP است. این ابزار هم اقدام به شناسایی الگوریتم IPID Sequence می‌کند.
در گام اول، از دو روش بالا برای اسکن شبکه‌ی 89.43.0.0 تا 89.43.7.255 استفاده کردیم. اما Zombieـی یافت نشد.
دستورهای

```
sudo nmap –script=ipidseq.nse -v 89.43.3.0/24
sudo nmap -sA -O -v 89.43.3.0/24
```

در ادامه به سراغ این رفتیم که خودمان Zombie مورد نظر را بسازیم. برای ساخت Zombie، باید از سیستم‌عاملی ساده استفاده کنیم که الگوریتم Incremental یا الگوریتم‌های آسیب‌پذیر دیگر را هنگام ساخت IPID استفاده می‌کند. روش اسکن Idle در سال 1998 مطرح شد، و از آن زمان به بعد رفته‌رفته سیستم‌عامل‌ها با به‌کارگیری الگوریتم‌های پیچیده‌تر، در برابر Zombie شدن ایمن شدند. مثلاً امروز برای ارتباط‌های مختلف، IPIDهای مختلف در نظر گرفته می‌شود. در نتیجه،

Zombie گذشته، حال با Target و Attacker به وسیله‌ی IPIDهای مستقل و بی‌ربط به همی ارتباط برقرار می‌کند و Attacker از تغییرات IPID ماشین Zombie هیچ نتیجه‌ای نمی‌تواند بگیرد (دیگر Zombie نیست!)
در این لینک، تعدادی از سیستم‌عامل‌های قدیمی که آسیب‌پذیر هستند، لیست شده است.

ابزار netdiscover

```
Currently scanning: 172.26.62.0/16   |   Screen View: Unique Hosts

47 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 2100
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.0.186   9c:9d:7e:ea:62:a0     38    1596  Unknown vendor
192.168.0.254   d8:d8:66:3f:09:3f      7     420  SHENZHEN TOZED TECHNOLOGIES CO.,LTD.
192.168.0.238   88:e9:fe:7a:28:be      1      42  Apple, Inc.
192.168.0.128   82:01:80:29:dc:b1      1      42  Unknown vendor
```

این ابزار با فرستادن درخواست‌های ARP، اقدام به شناسایی Hostهای فعال شبکه به همراه IP و MAC Addressشان می‌کند. ARP یا Address Resolution Protocol، پروتکلی برای نگاشت IP دستگاه‌ها به آدرس فیزیکی‌شان است. با داشتن این نگاشت، هر دستگاه می‌داند که برای ارسال به IP مثال X، به چه آدرس فیزیکی‌ای باید بسته را بفرستد. در تصویر بالا، خروجی این ابزار را برای شبکه‌ی داخلی می‌بینیم.

ابزار hping3

از این ابزار برای فرستادن بسته‌های دلخواه به یک آدرس آی‌پی مقصد استفاده می‌شود. با آرگومان -c تعداد پکت‌های ارسالی، با -S آن آدرس را با پکت‌های SYN (ارتباط TCP) اسکن می‌کند.

```
┌──(~)──────────────────────────────────────────────────────────────────────────────────────────────────────(divar@divar-ThinkPad-E14:pts/0)─┐
└─(23:15:14)──> sudo hping3 -S 192.168.0.254                                                                  1 ↵ ──(جزمعه,نوًوامبر04)─┘
HPING 192.168.0.254 (wlp0s20f3 192.168.0.254): S set, 40 headers + 0 data bytes
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=0 win=0 rtt=3.6 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=1 win=0 rtt=11.2 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=2 win=0 rtt=10.9 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=3 win=0 rtt=26.6 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=4 win=0 rtt=90.2 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=5 win=0 rtt=89.8 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=6 win=0 rtt=13.7 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=7 win=0 rtt=69.2 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=8 win=0 rtt=12.9 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=9 win=0 rtt=4.2 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=10 win=0 rtt=3.8 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=11 win=0 rtt=39.5 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=12 win=0 rtt=11.4 ms
len=46 ip=192.168.0.254 ttl=64 DF id=0 sport=0 flags=RA seq=13 win=0 rtt=11.1 ms
^C
--- 192.168.0.254 hping statistic ---
14 packets transmitted, 14 packets received, 0% packet loss
round-trip min/avg/max = 3.6/28.4/90.2 ms
```

```
┌──(~)──────────────────────────────────────────────────────────────────────────────────────────────────────(divar@divar-ThinkPad-E14:pts/0)─┐
└─(23:19:15)──> sudo hping3 -S 185.166.104.4 -c 4                                                             1 ↵ ──(جزمعه,نوًوامبر04)─┘
HPING 185.166.104.4 (tun0 185.166.104.4): S set, 40 headers + 0 data bytes

--- 185.166.104.4 hping statistic ---
4 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

اطلاعات بیش‌تر راجع به Hostهای فعال اسکن شده:

با استفاده از وب‌سایت www.infobyip.com راجع به آدرس آی‌پی 89.43.3.20 به اطلاعات زیر می‌رسیم.

- دامنه: https://20.mobinnet.net/
- شرکت خدمات‌دهنده اینترنت: Mobin Net Communication Company (Private Joint Stock)
- موقعیت جغرافیایی: 35.698 / 51.4115 - ایران، تهران
- ای‌اس‌ان (ASN): برابر است با 50810

این وب‌سایت از مواردی چون whois و اطلاعات مربوط به دامنه (به کمک dns)برای جمع‌آوری اطلاعات راجع به آدرس آی‌پی استفاده می‌کند.