Scan and network statistics !

Πίνακας περιεχομένων

1. Install docker (Home PC)

HowTo: See How to

NOTE

Assuming you’re already logged in

2. Find IP

```
# ifconfig

eth0:   flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.21.0.2  netmask 255.255.0.0  broadcast 172.21.255.255
        ether 02:42:ac:15:00:02  txqueuelen 0  (Ethernet)
        RX packets 61  bytes 9309 (9.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo:     flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 248  bytes 14260 (14.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 248  bytes 14260 (14.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

inet **172.21.0.2** netmask 255.255.0.0 broadcast 172.21.255.255

3. Find live hosts

```
nmap -sP 172.21.0.*

Nmap scan report for 172.21.0.1 (172.21.0.1)
Host is up (0.00028s latency).
MAC Address: 02:42:56:48:D0:61 (Unknown)
Nmap scan report for testnetwork2_worker_1.testnetwork2_net (172.21.0.3)
Host is up (0.00015s latency).
MAC Address: 02:42:AC:15:00:03 (Unknown)
Nmap scan report for testnetwork2_worker_3.testnetwork2_net (172.21.0.4)
Host is up (0.00018s latency).
MAC Address: 02:42:AC:15:00:04 (Unknown)
Nmap scan report for testnetwork2_worker_4.testnetwork2_net (172.21.0.5)
Host is up (0.00015s latency).
MAC Address: 02:42:AC:15:00:05 (Unknown)
Nmap scan report for testnetwork2_worker_2.testnetwork2_net (172.21.0.6)
Host is up (0.00017s latency).
MAC Address: 02:42:AC:15:00:06 (Unknown)
Nmap scan report for 9110d42e466b (172.21.0.2)


nmap -sP 172.21.0.* | grep Nmap | cut -d' ' -f5-6

172.21.0.1 (172.21.0.1)
testnetwork2_worker_1.testnetwork2_net (172.21.0.3)
testnetwork2_worker_3.testnetwork2_net (172.21.0.4)
testnetwork2_worker_4.testnetwork2_net (172.21.0.5)
testnetwork2_worker_2.testnetwork2_net (172.21.0.6)
9110d42e466b (172.21.0.2)
```

ℹ

What is Nmap?

Nmap, short for Network Mapper, is a free, open-source tool for vulnerability scanning and network discovery. Network administrators use Nmap to identify what devices are running on their systems, discovering hosts that are available and the services they offer, finding open ports and detecting security risks.

See https://en.wikipedia.org/wiki/Nmap

4. Scan Ports

4.1. Scan a Single Port, All Ports, or Series

```
Nmap commands can be used to scan a single port or a series of ports:
```

4.2. Scan port 80 on the target system:

```
nmap □p 80 172.21.0.3
```

4.3. Scan ports 1 through 200 on the target system:

```
nmap □p 1-200 172.21.0.3
```

4.4. Scan (Fast) the most common ports:

```
nmap □F 172.21.0.3
```

4.5. To scan all ports (1 – 65535):

```
nmap □p□ 172.21.0.3
```

4.6. Scan All TCP UDP Ports

Scan all UDP and TCP ports in a single command. We will use -sU for UDP and sT for TCP protocol.

```
nmap -sU -sT -p0-65535 IP
```

> *What Are Ports?*
>
> On modern operating systems, ports are numbered addresses for network traffic. Different kinds of services use different ports by default.
>
> For example, normal web traffic uses Port 80, while POP3 email uses Port 110. One of the ways that a firewall works is by allowing or restricting traffic over a particular port.
>
> Because the ports into your computer can cause a security risk, it’s critical to know which ports are open and which are blocked.

5. netstat

5.1. Listing (Almost all)

```
netstat -antlupe
```

5.2. Listing TCP Ports connections

```
netstat -at
```

> *Netstat*
>
> Netstat command displays various network related information such as network connections, routing tables, interface statistics, masquerade connections, multicast memberships etc.,

5.3. Listing UDP Ports connections

```
netstat -au
```

5.4. Listing all LISTENING Connections

```
netstat -l
```

5.5. Listing all TCP Listening Ports

```
netstat -lt
```

5.6. Listing all UDP Listening Ports

```
netstat -lu
```

5.7. Listing all UNIX Listening Ports

```
netstat -lx
```

5.8. Showing Statistics by Protocol

```
netstat -s
```

5.9. Showing Statistics by TCP Protocol

```
netstat -st
```

5.10. Showing Statistics by UDP Protocol

```
netstat -su
```

5.11. Displaying Service name with PID

```
netstat -tp
```

5.12. Displaying Promiscuous Mode

Displaying Promiscuous mode with -ac switch, netstat print the selected information or refresh screen every five second. Default screen refresh in every second.

```
 netstat -ac 5 | grep tcp
```

5.13. Setting Promiscuous Mode

```
 ifconfig eth0 promisc

OR

ip link set eth0 promisc on
```

5.14. Remove Promiscuous Mode

```
 ifconfig eth0 □promisc
```

Promiscuous Mode

Promiscuous mode is a mode for a wired network interface controller (NIC) or wireless network interface controller (WNIC) that causes the controller to pass all traffic it receives to the central processing unit (CPU) rather than passing only the frames that the controller is specifically programmed to receive.

When a capable NIC is placed in Promiscuous Mode, it allows the NIC to intercept and read each arriving network packet in its entirety.

If the NIC is not in Promiscuous Mode, it will only receive packets that are specifically addressed to the NIC. Promiscuous Mode must be supported by the NIC and by the operating system and any associated driver. Not all NICs support Promiscuous Mode, however it is pretty easy to determine if you have a NIC and OS capable of Promiscuous Mode.

5.15. check if promiscuous mode is enabled on network interface

```
netstat -i

Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0      1500     8352      0      0 0         12677      0      0      0 BMRU
lo       65536    14656      0      0 0         14656      0      0      0 LRU

ifconfig eth0 promisc

netstat -i

Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0      1500     8352      0      0 0         12677      0      0      0 BMPRU
lo       65536    14696      0      0 0         14696      0      0      0 LRU

ifconfig eth0 -promisc

netstat -i

Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0      1500     8353      0      0 0         12677      0      0      0 BMRU
lo       65536    15232      0      0 0         15232      0      0      0 LRU
```

Promiscuous Mode

Look under the last column “Flg” for value “P”. If it’s there, it means promiscuous mode is enabled for that network interface
B flag is for broadcast
M flag is for multicast
P flag is for promisc mode
R is for running
U is for up

Appendix A: How to use Nmap

While the basis of Nmap’s functionality is port scanning, it allows for a variety of related capabilities including:

- Network mapping: Nmap can identify the devices on a network (also called host discovery), including servers, routers and switches, and how they’re physically connected.
- OS detection: Nmap can detect the operating systems running on network devices (also called OS fingerprinting), providing the vendor name, the underlying operating system, the version of the software and even an estimate of devices' uptime.
- Service discovery: Nmap can not only identify hosts on the network, but whether they’re acting as mail, web or name servers, and the particular applications and versions of the related software they’re running.
- Security auditing: Figuring out what versions of operating systems and applications are running on network hosts lets network managers determine their vulnerability to specific flaws. If a network admin receives an alert about a vulnerability in a particular version of an application, for example, she can scan her network to identify whether that software version is running on the network and take steps to patch or update the relevant hosts. Scripts can also automate tasks such as detecting specific vulnerabilities.

Reminder

Caminante, no hay camino,
se hace camino al andar.

Wanderer, there is no path,
the path is made by walking.

Antonio Machado Campos de Castilla