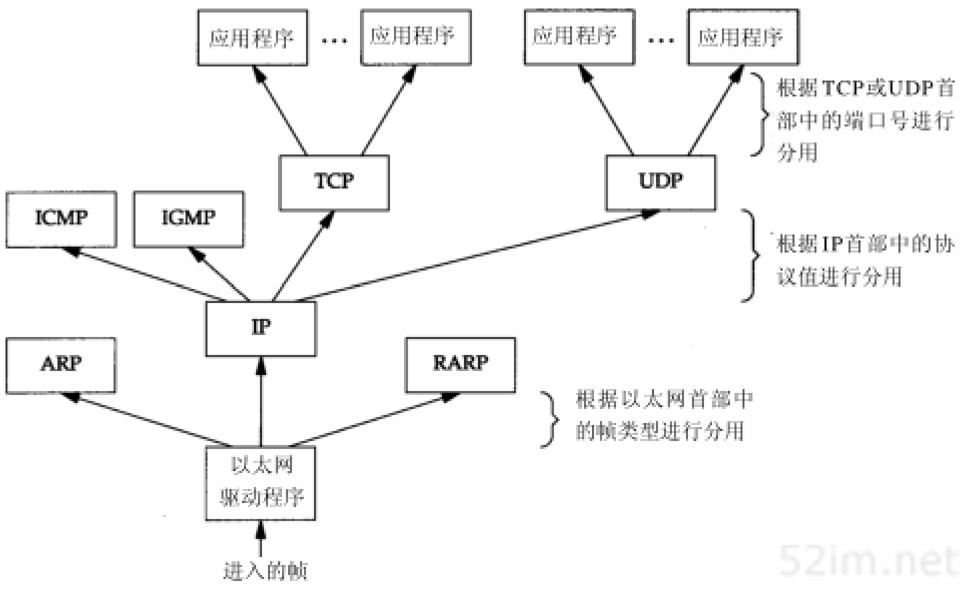
Nmap 使用指南

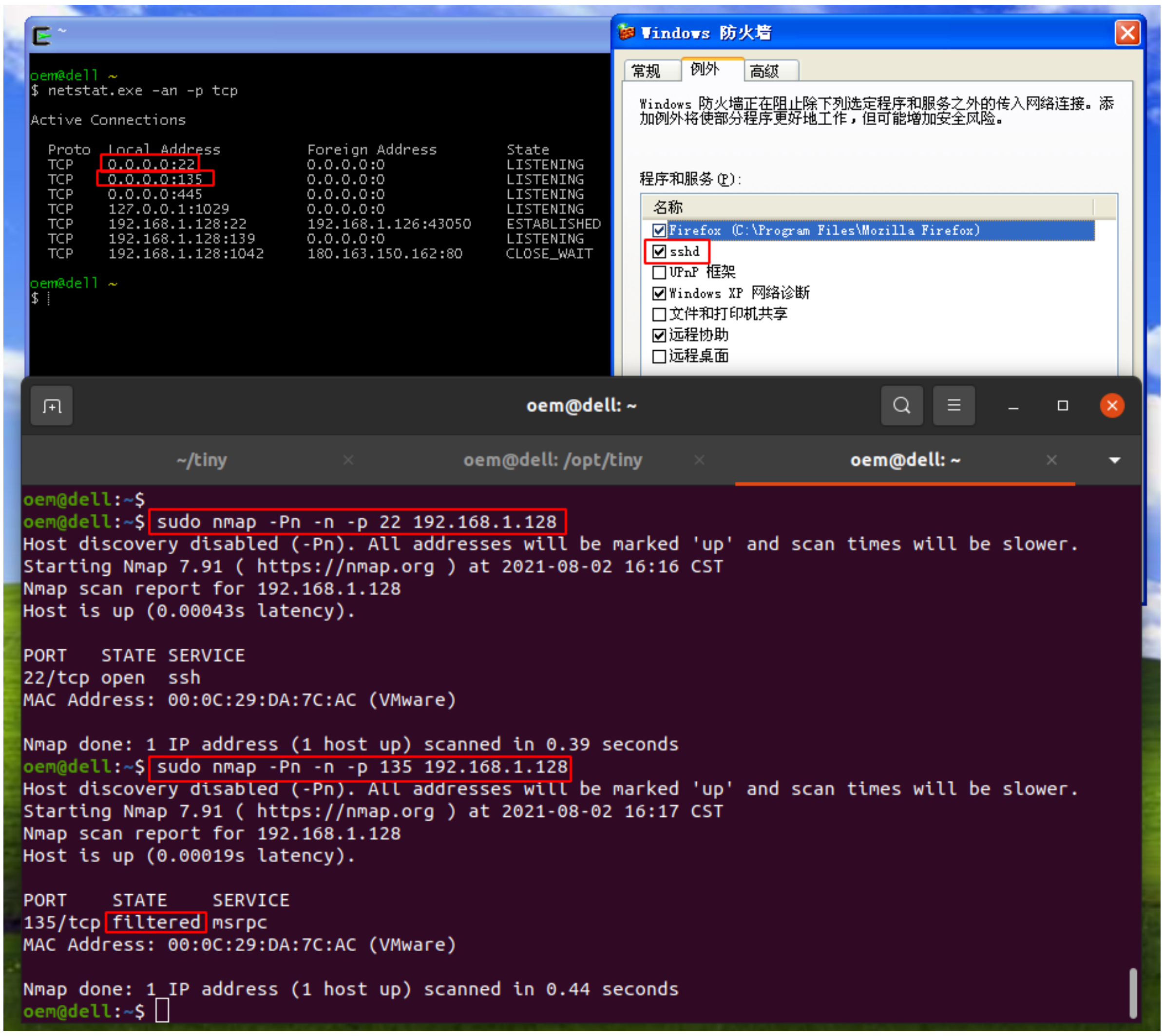
开局一张图



应用程序
…
应用程序
应用程序
…
应用程序
根据TCP或UDP首部中的端口号进行分用
TCP
UDP
ICMP
IGMP
根据IP首部中的协议值进行分用
IP
ARP
RARP
根据以太网首部中的帧类型进行分用
以太网驱动程序
进入的帧
52im.net

端口状态

> filtered: 被防火墙等过滤



```
oem@dell ~
$ netstat.exe -an -p tcp

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:22             0.0.0.0:0              LISTENING
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    127.0.0.1:1029         0.0.0.0:0              LISTENING
  TCP    192.168.1.128:22       192.168.1.126:43050    ESTABLISHED
  TCP    192.168.1.128:139      0.0.0.0:0              LISTENING
  TCP    192.168.1.128:1042     180.163.150.162:80     CLOSE_WAIT

oem@dell ~
$
```

Windows 防火墙

常规 | 例外 | 高级

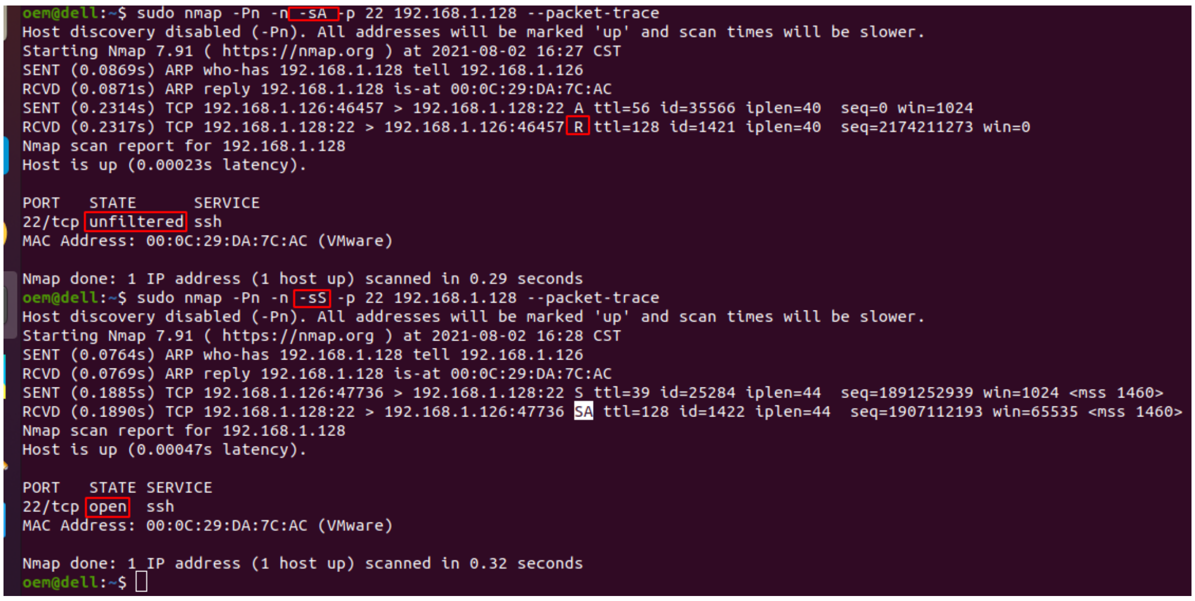
Windows 防火墙正在阻止除下列选定程序和服务之外的传入网络连接。添加例外将使部分程序更好地工作，但可能增加安全风险。

程序和服务(P):

名称
- [✓] Firefox (C:\Program Files\Mozilla Firefox)
- [✓] sshd
- [ ] UPnP 框架
- [✓] Windows XP 网络诊断
- [ ] 文件和打印机共享
- [✓] 远程协助
- [ ] 远程桌面

```
oem@dell:~$
oem@dell:~$ sudo nmap -Pn -n -p 22 192.168.1.128
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-08-02 16:16 CST
Nmap scan report for 192.168.1.128
Host is up (0.00043s latency).

PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 00:0C:29:DA:7C:AC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.39 seconds
oem@dell:~$ sudo nmap -Pn -n -p 135 192.168.1.128
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-08-02 16:17 CST
Nmap scan report for 192.168.1.128
Host is up (0.00019s latency).

PORT    STATE    SERVICE
135/tcp filtered msrpc
MAC Address: 00:0C:29:DA:7C:AC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
oem@dell:~$
```

> unfiltered: 使用 -sA 时, 有回复.



```
oem@dell:~$ sudo nmap -Pn -n -sA -p 22 192.168.1.128 --packet-trace
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-08-02 16:27 CST
SENT (0.0869s) ARP who-has 192.168.1.128 tell 192.168.1.126
RCVD (0.0871s) ARP reply 192.168.1.128 is-at 00:0C:29:DA:7C:AC
SENT (0.2314s) TCP 192.168.1.126:46457 > 192.168.1.128:22 A ttl=56 id=35566 iplen=40  seq=0 win=1024
RCVD (0.2317s) TCP 192.168.1.128:22 > 192.168.1.126:46457 R ttl=128 id=1421 iplen=40  seq=2174211273 win=0
Nmap scan report for 192.168.1.128
Host is up (0.00023s latency).

PORT   STATE      SERVICE
22/tcp unfiltered ssh
MAC Address: 00:0C:29:DA:7C:AC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
oem@dell:~$ sudo nmap -Pn -n -sS -p 22 192.168.1.128 --packet-trace
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-08-02 16:28 CST
SENT (0.0764s) ARP who-has 192.168.1.128 tell 192.168.1.126
RCVD (0.0769s) ARP reply 192.168.1.128 is-at 00:0C:29:DA:7C:AC
SENT (0.1885s) TCP 192.168.1.126:47736 > 192.168.1.128:22 S ttl=39 id=25284 iplen=44  seq=1891252939 win=1024 <mss 1460>
RCVD (0.1890s) TCP 192.168.1.128:22 > 192.168.1.126:47736 SA ttl=128 id=1422 iplen=44  seq=1907112193 win=65535 <mss 1460>
Nmap scan report for 192.168.1.128
Host is up (0.00047s latency).

PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: 00:0C:29:DA:7C:AC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds
oem@dell:~$
```

工作原理

主机发现

> -PE -PS443 -PA80 -PP
>
> sudo nmap -n -sn 114.114.114.114 --packet-trace

```
SENT (0.0872s) ICMP [192.168.1.126 > 114.114.114.114 Echo request (type=8/code=0)
...]
SENT (0.0872s) TCP 192.168.1.126:37488 > 114.114.114.114:443 S ...
SENT (0.0872s) TCP 192.168.1.126:37488 > 114.114.114.114:80 A ...
SENT (0.0872s) ICMP [192.168.1.126 > 114.114.114.114 Timestamp request
(type=13/code=0) ...]
RCVD (0.1120s) ICMP [114.114.114.114 > 192.168.1.126 Timestamp reply
(type=14/code=0) ...]
```

> sudo nmap -n -sn 192.168.1.1 `--disable-arp-ping` --packet-trace

```
SENT (0.0579s) ICMP [192.168.1.126 > 192.168.1.1 Echo request (type=8/code=0)
...]
SENT (0.0579s) TCP 192.168.1.126:38365 > 192.168.1.1:443 S ...
SENT (0.0580s) TCP 192.168.1.126:38365 > 192.168.1.1:80 A ...
SENT (0.0580s) ICMP [192.168.1.126 > 192.168.1.1 Timestamp request
(type=13/code=0) ...]
RCVD (0.0587s) ICMP [192.168.1.1 > 192.168.1.126 Echo reply (type=0/code=0) ...]
```

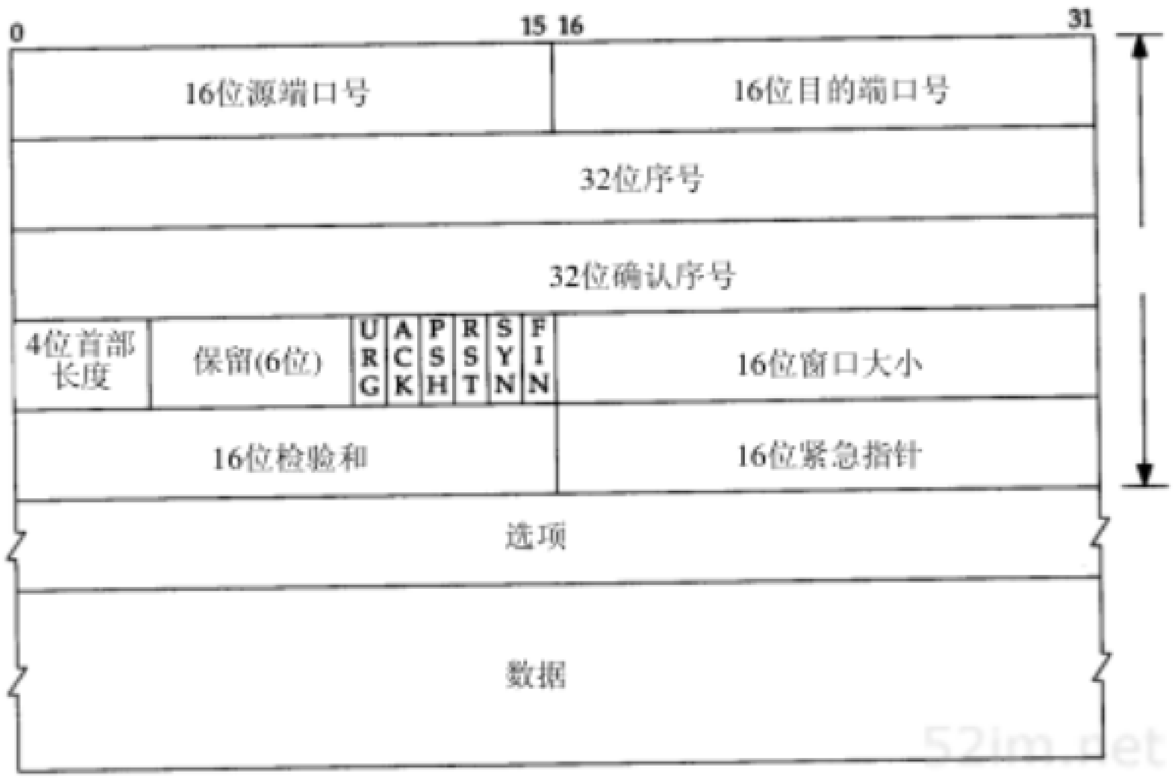
> 为什么要使用 --disable-arp-ping

```
SENT (0.0579s) ARP who-has 192.168.1.1 tell 192.168.1.126
RCVD (0.0585s) ARP reply 192.168.1.1 is-at CC:C2:E0:A2:B0:CC
```

> 我们比较关注的几个参数

1. -Pn
2. -sn
3. -n
4. -PO

端口扫描



> -sS, -sA, -sN, -sF, -sX(URG, RSH, FIN), -sW, --scanflags(可以跟数字和URGACKPSHRSTSYNFIN)
>
> -sT?

> -sI: https://blog.csdn.net/dong976209075/article/details/7771159

> 请注意，要同时扫描 UDP 和 TCP，您必须指定 -sU 和至少一种 TCP 扫描类型（例如 -sS、-sF 或 -sT）

> nmap 默认扫描多少个端口?
>
> > `-r` (Don't randomize ports)
>
> -F 扫描多少端口?

服务探测

https://www.cnblogs.com/liun1994/p/6985796.html

NSE

怎么调 nse

> --script `<filename>` | `<category>` | `<directory>` / | `<expression>` [,...]

nse 格式

```
description Field
categories Field
author Field
license Field
dependencies Field
Rules
Action
Environment Variables
```

nse 的两种分类依据

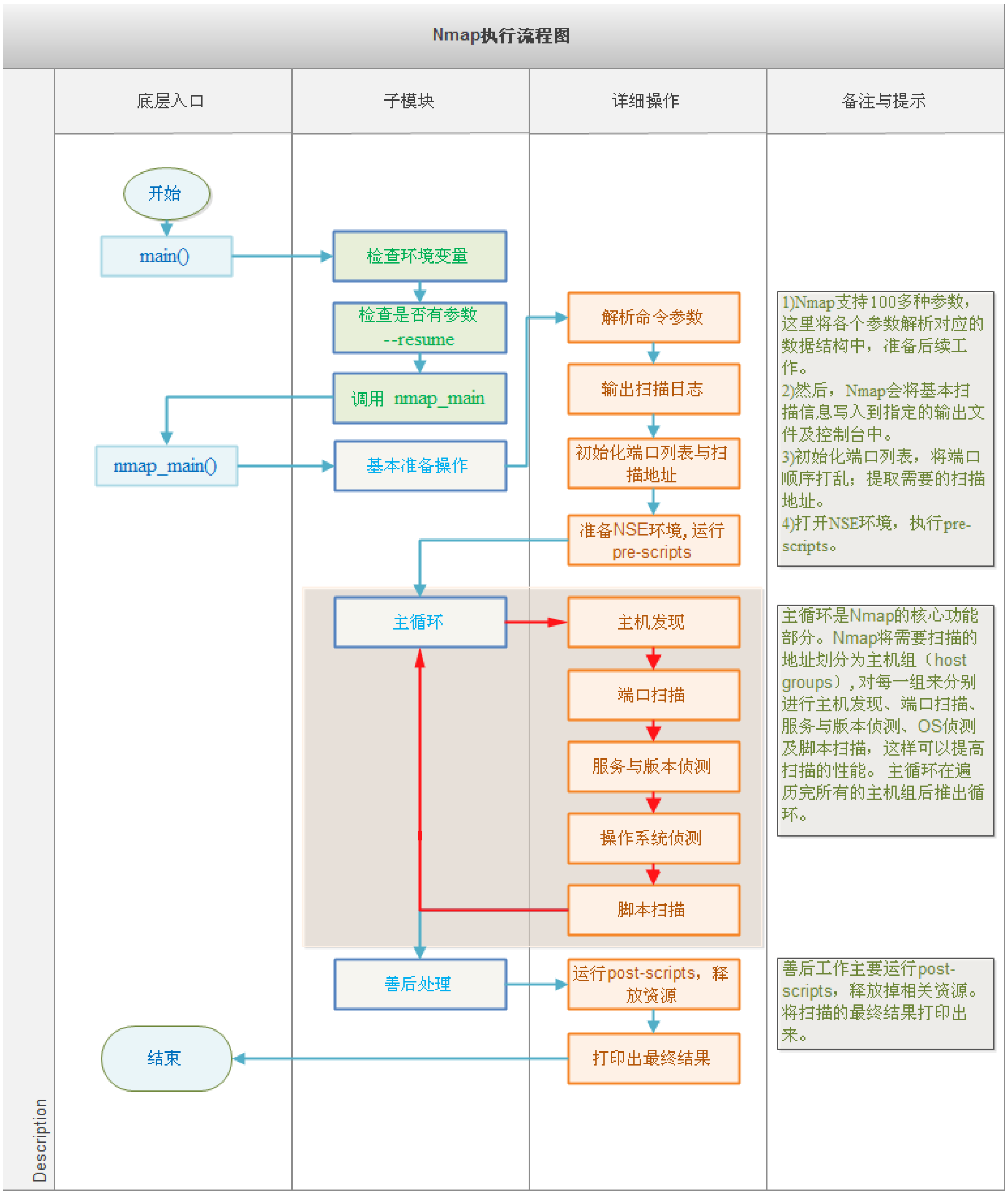
> 1. 种类: auth, broadcast, brute, default...
> 2. 运行时: prerule, host, service, postrule

```
portrule = function(host, port)
    return port.state == "open" and port.number == 80 and port.protocol == "tcp"
end

action = function(host, port)
    return {web=true, title="demo"}
end
```

Rules

```
prerule()
hostrule(host)
portrule(host, port)
postrule()
```

概览



1)Nmap支持100多种参数，这里将各个参数解析对应的数据结构中，准备后续工作。
2)然后，Nmap会将基本扫描信息写入到指定的输出文件及控制台中。
3)初始化端口列表，将端口顺序打乱，提取需要的扫描地址。
4)打开NSE环境，执行pre-scripts。

主循环是Nmap的核心功能部分。Nmap将需要扫描的地址划分为主机组（host groups），对每一组来分别进行主机发现、端口扫描、服务与版本侦测、OS侦测及脚本扫描，这样可以提高扫描的性能。主循环在遍历完所有的主机组后推出循环。

善后工作主要运行post-scripts，释放掉相关资源。将扫描的最终结果打印出来。

看一个真实的案例

```
description = "Fingerprints Red Lion HMI devices"
author = "Thought Leader"
email_address = "thoughtleader@internetofallthethings.com"
license = "TO-ILL"
categories = {"version", "discovery"}
```

```
stdnse = require "stdnse"

-- Perform discovery using Red Lion Crimson V3 Protocol

-- this method should expose a user configuration
portrule = function(host, port)
    return port.number == 789
end

action = function(host, port)
    local client = nmap.new_socket()
    local catch = function()
        client:close()
    end

    local try = nmap.new_try(catch)

    -- first fingerprint gets the manufacturer info
    try(client:connect(host.ip, 789))

    local localip, loaclport, remoteip, remoteport =
        try(client:get_info())

    local probe_manufacturer = string.char(0x00,0x04,0x01,0x2b,0x1b,0x00)
    try(client:send(probe_manufacturer))
    resp = try(client:receive())

    if string.len(resp) > 2 then
        -- return the result, skipping the CR3 header and omitting the trailing
null
        resp_string = "\nManufacturer: " .. string.sub(resp, 7, -2)
    end

    try(client:close())

    -- second fingerprint gets the model information
    try(client:connect(host.ip, 789))

    local localip, loaclport, remoteip, remoteport =
        try(client:get_info())

    local probe_manufacturer = string.char(0x00,0x04,0x01,0x2a,0x1a,0x00)
    try(client:send(probe_manufacturer))
    resp = try(client:receive())

    if string.len(resp) > 2 then
        -- return the result, skipping the CR3 header and omitting the trailing
null
        resp_string = resp_string .. "\nModel: " .. string.sub(resp, 7, -2) ..
"\n"
    end

    try(client:close())

    return resp_string

end
```

一个问题

为什么 udp 扫描比较慢