# 三.nmap-操作文档

# 概述

Nmap（network mapper，网络映射器），一款开放源代码的网络探测和安全审核工具。

# 作用

被设计用来快速扫描大型网络，包括主机探测与发现、开放的端口情况，操作系统与应用服务指纹识别、WAF识别以及常见安全漏洞

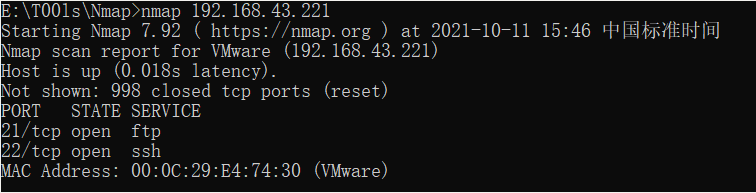
# 特点

主机探测：探测网络上的主机端口扫描：探测目标主机开放的端口版本检测：探测目标主机的网络服务，判断其服务名称和版本号系统检测：探测目标主机的操作系统及网络设备的硬件特性支持探测脚本的编写

# 常用方法

## 扫描单个目标地址

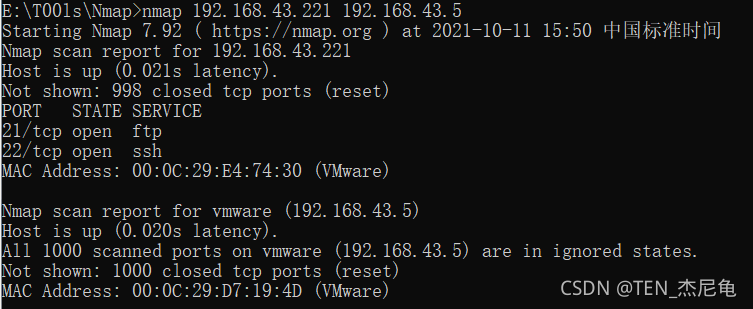
nmap 192.168.43.221

[MISSING IMAGE: , ]

## 扫描多个目标地址

如果目标地址不在同一网段，或者在同一网段但是不连续且数量不多

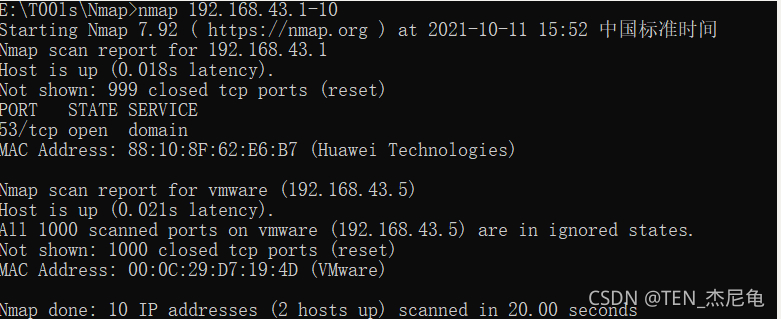
nmap 192.168.43.221 192.168.43.5

[MISSING IMAGE: , ]

## 扫描一个范围内的的目标地址

指定扫描一个连续的网段，使用“-” 连接

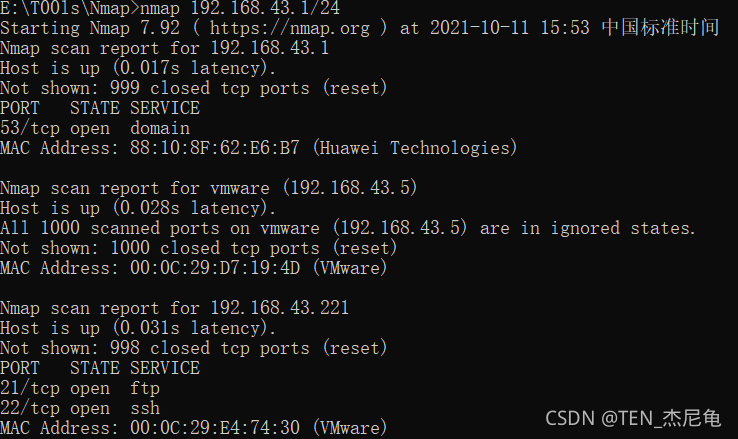
nmap 192.168.43.1-10

[MISSING IMAGE: , ]

## 扫描目标地址所在某个网段

如果目标是一个网段，则可以通过添加子网掩码的方式扫描

nmap 192.168.43.1/24

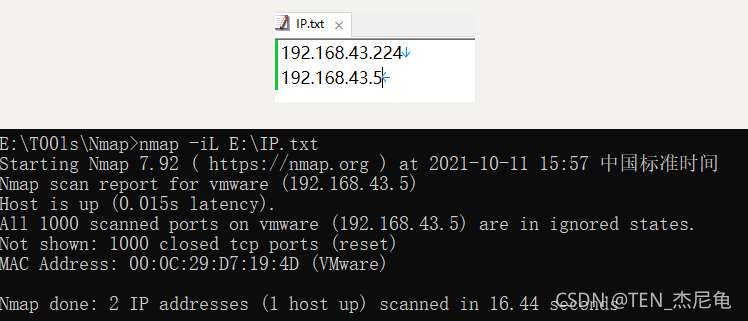
[MISSING IMAGE: , ]

## 扫描主机列表txt中的所有地址–iL

将众多IP地址写入到txt中，通过namp进行读取

注意：导入的txt地址为绝对路径

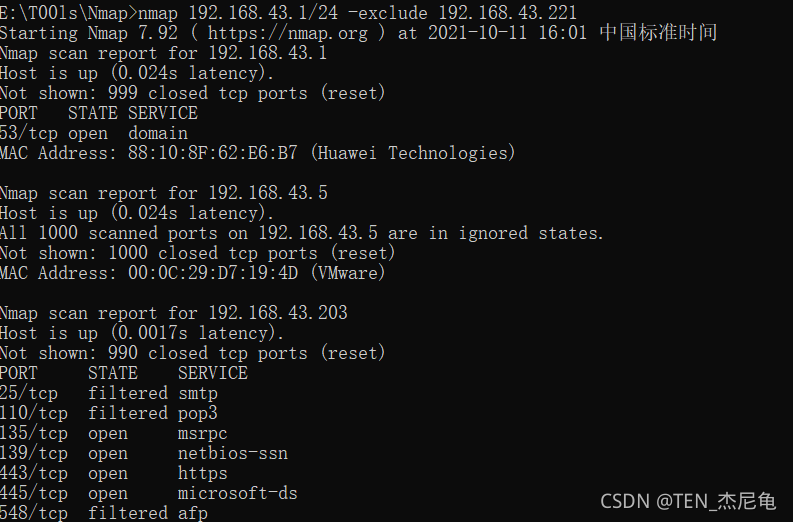
nmap -iL E:\IP.txt

[MISSING IMAGE: , ]

## 扫描除某一个目标地址之外的所有目标地址–exclude

如果扫描除了192.168.43.221之外的所有192.168.43.x地址

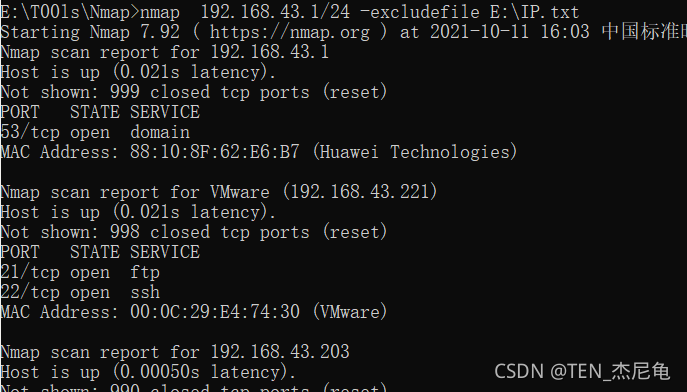
nmap 192.168.43.1/24 -exclude 192.168.43.221

[MISSING IMAGE: , ]

## 扫描除某一文件中的目标地址之外的目标地址–excludefile

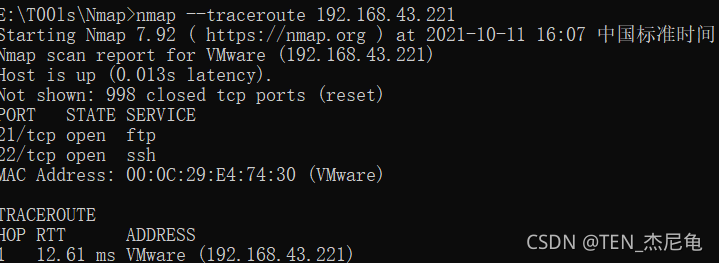
扫描除了文件txt中的地址之外的所有IP地址

nmap 192.168.43.1/24 -excludefile E:\IP.txt

[MISSING IMAGE: , ]

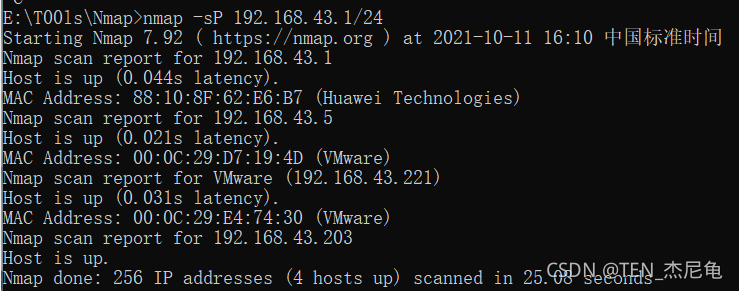
## 对目标地址进行路由跟踪 --traceroute

nmap --traceroute 192.168.43.221

[MISSING IMAGE: , ]

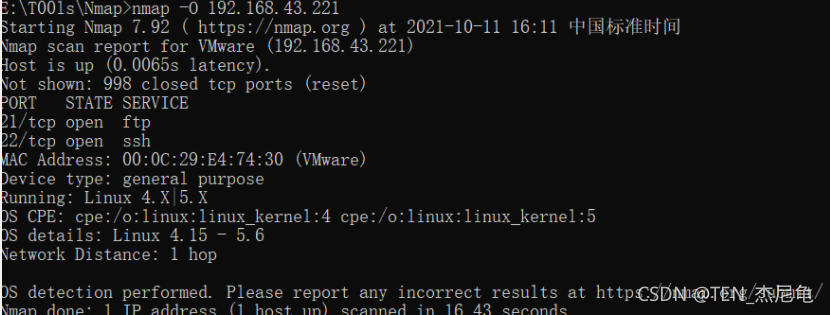
## 扫描目标地址所在C端的在线情况–sP

nmap -sP 192.168.43.1/24

[MISSING IMAGE: , ]

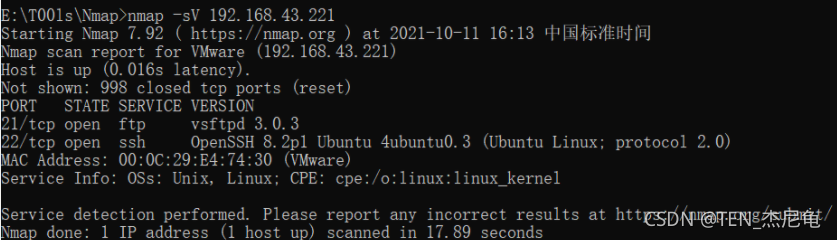
## 对目标地址进行操作系统指纹识别–O

nmap -O 192.168.43.221

[MISSING IMAGE: , ]

## 扫描目标地址提供的服务版本探测–sV

nmap -sV 192.168.43.221

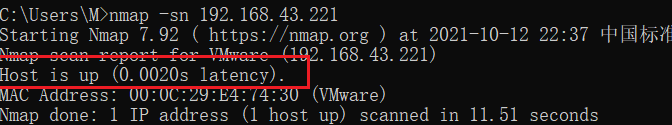
[MISSING IMAGE: , ]

# 设备发现

## 只进行设备发现，不扫描端口–sn

如果只想指定设备是否在线，不扫描端口，可以使用-sn

nmap -sn 192.168.43.221

[MISSING IMAGE: , ]

## ARP设备发现

按照ARP规定，当设备B收到来自设备A的ARP请求数据包时，设备B应当向设备A回应一个数据包。

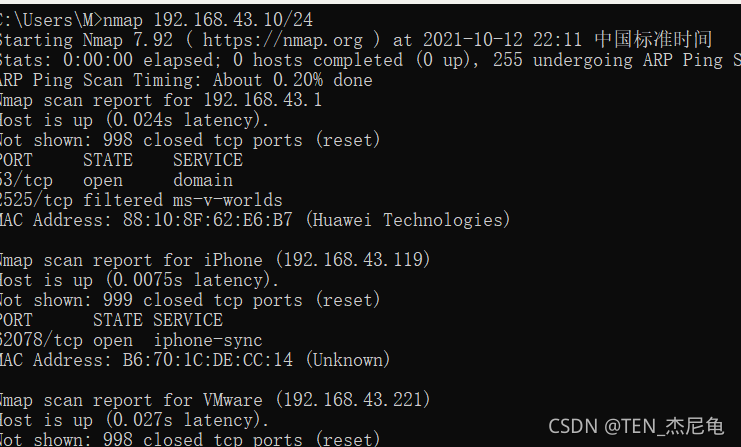
故ARP活跃设备发现技术的原理：构造一个ARP数据包，并广播出去，如果得到了回应，则说明该设备为活跃设备。

优点：准确度高，任何处于同一网段的设备都没有办法防御此技术，没有任何的安全措施会阻止正常的ARP请求。

缺点：不能对处于不同网段的目标设备进行扫描。

使用方法：在Nmap7.8以上，ARP扫描技术会在可能使用的地方默认启动

nmap 192.168.43.10/24

[MISSING IMAGE: , ]

## ICMP设备发现

ICMP–因特网控制消息协议，位于网络层，目的是在IP设备、路由器之间传递控制消息

ICMP的报文分为两类：差错和查询

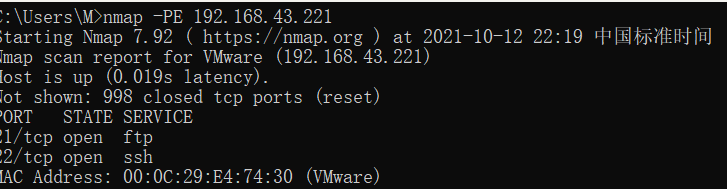
ICMP活跃设备发现技术使用的是查询报文，其中查询报文又分为三类：回送请求和回答、时间戳请求和回答、地址掩码请求和回答

## -PE

这个过程实际上和ping命令一样，设备A向目标设备B发送一个ICMP echo请求数据包，注意包中的type字段的值为8

如果目标设备在线，而且没有防火墙隔离通信，将会收到目标设备发回ICMP echo回应包

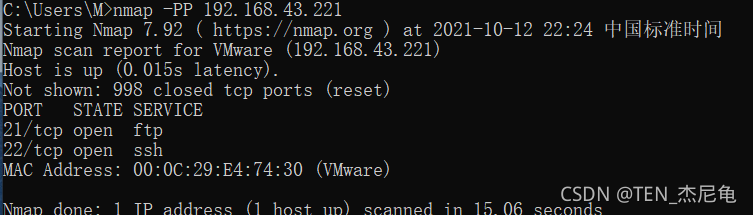
nmap -PE 192.168.43.221

[MISSING IMAGE: , ]

## PP

时间戳设备发现，这种扫描方式和使用- PE 作为参数很相似，但是Type的值为13。当目标设备接收到了这个数据包之后，会给出一个Type字段值为14的回应数据包

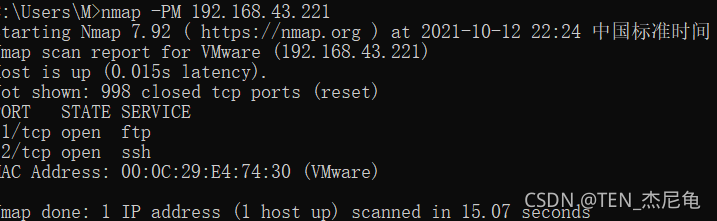
nmap -PP 192.168.43.221

[MISSING IMAGE: , ]

## PM

地址掩码设备发现，该扫描方式仅将Type字段的值换为17

nmap -PM 192.168.43.221

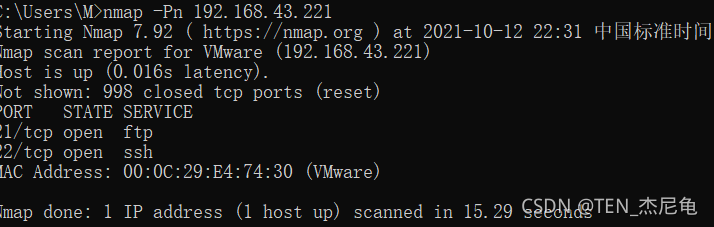
[MISSING IMAGE: , ]

## 注意(- Pn)

通常，Nmap在进行其他扫描之前，都会对目标设备进行ping扫描。如果目标设备对ping扫描没反应，就会直接结束整个扫描过程，这种扫描方式可以跳过没有响应的设备，从而节省大量的时间。但是 如果目标设备在线，只是采用了某种手段屏蔽了ping扫描，那么就会躲过其他扫描操作。因此，可以指定无论目标设备是否响应ping扫描，都会执行整个扫描过程。

参数：-Pn

nmap -Pn 192.168.43.221

[MISSING IMAGE: , ]

## TCP 设备发现

TCP的主要过程由“三次握手”构成，

第一次：客户端发送同步序列编号（SYN Seq=x）的数据包给服务端，进入SYN\_SEND状态第二次：服务端收到客户端的SYN数据包，回应一个SYN（Seq=y）+ ACK（Ack=x+1）数据包，进入SYN\_RECV状态第三次：客户端收到服务端的SYN数据包，回应一个ACK（Ack=y+1）数据包，进入ESTABLISHED状态利用这个过程，Nmap向目标设备发送SYN数据包，如果对方回应了SYN+ACK，则说明目标设备在线

因为三次握手中的第三次握手意义不大，所以在扫描的时候，第三次握手可以完成也可以不完成，

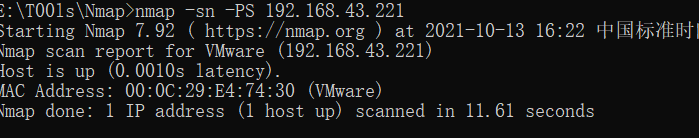
如果完成，一般称之为全开扫描；如果不完成，一般称之为半开扫描

### TCP SYN扫描 --PS

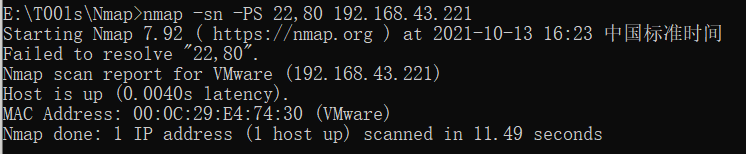
Nmap使用-PS参数向目标设备发送一个设置了SYN标志的数据包，这个数据包的内容部分为空，通常默认的目标端口是80端口，也可以使用参数来改变目标端口，当指定多个端口时，Nmap将会并行地对这些端口进行测试

并不在乎端口是否处于开放状态，只在乎目标设备是否处于活跃状态，发出SYN数据包之后，只要收到了数据包，无论是SYN+ACK还是RST数据包，都意味着目标设备是活跃状态。如果没有收到任何数据包，就意味着目标设备不在线（或者带端口被屏蔽，像石沉大海一样）

nmap -sn -PS 192.168.43.221

[MISSING IMAGE: , ]

nmap -sn -PS 22,80 192.168.43.221

[MISSING IMAGE: , ]

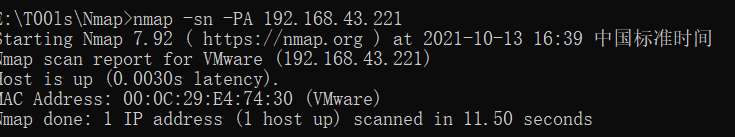
### TCP ACK扫描 --PA

ACK扫描与SYN扫描很相似，不同之处只在于Nmap发送的数据包中使用了SYN+ACK标志位

Nmap向目标设备发送一个SYN+ACK数据包，目标设备显然不清楚这是这么回事，因此，只会向Nmap所在的设备回应一个RST数据包，表示无法建立这个TCP连接，同时也表示目标设备处于开放状态

注意：ACK扫描和SYN扫描同样默认使用80端口，也可以进行指定

nmap -sn -PA 192.168.43.221

[MISSING IMAGE: , ]

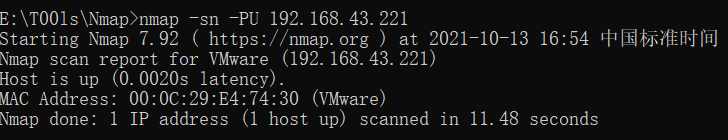
在Nmap所在设备发出数据包之后，并没有收到任何回应，可能由两种情况：

数据包被目标设备上的安全机制过滤了目标设备并不是活跃设备

## UDP设备发现 --PU

UDP 用户数据包协议，位于传输层的协议，当一个端口接收到一个UDP数据包时，如果它是关闭的，就会给源端口发回一个ICMP端口不可达的数据包；如果它是开放的，就会忽略这个数据包，也就是将它丢弃而不返回任何的信息

优点：可以完成对UDP端口的探测缺点：扫描结果可靠性不高、扫描速度慢nmap -sn -PU 192.168.43.221

[MISSING IMAGE: , ]

# 端口扫描

程序在发送的信息中加入端口编号，而操作系统在接收到信息后会按照端口编号将信息分流到当前内存中使用该端口的程序

## 端口分类

工认端口：0–1024，绑定一些特殊服务，通常这些端口的通信明确表明了某种服务的协议，不可重新定义它的作用对象注册端口：1025–49151，松散的绑定于一些服务，这些端口多数没有明确的地定义服务对象，不同程序可根据实际需要自己定义动态/私有端口：49152–65535，理论上，不应把常用服务分配到这些端口，但是有些较为特殊的程序，特别是木马程序非常循环这些端口

## Nmap对端口的定义

|  |  |
| --- | --- |
| 端口状态 | 说明 |
| open | 应用程序在该端口接收TCP连接或者UDP报文 |
| closed | 关闭的端口对应Nmap也可访问，该端口接收Nmap的探测报文并作出响应，但是没有应用程序在其上监听 |
| filtered | 由于包过滤阻止探测报文到达端口，Nmap无法确定该端口是否开放。过滤原因可能时来自专业的防火墙设备、路由规则或者是设备上的防火墙 |
| unfiltered | 未被过滤状态，意味着端口可访问，但是Nmap无法确定该端口是否开放。只有用于映射防火墙规则集的ACK扫描才会把端口分类到 这个状态 |
| open|filtered | 无法确定该端口是开放的还是被过滤的，例如：开放的端口不响应 |
| closed|filtered | 无法确定该端口是关闭的还是被过滤的 |

## SYN扫描–sS

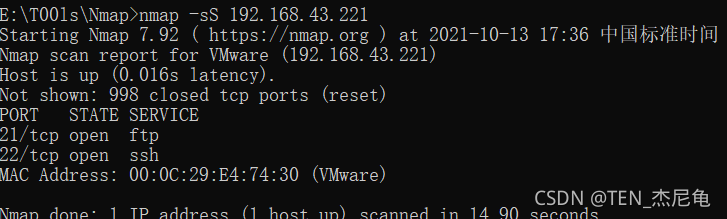
Nmap默认的扫描方式，速度极快，可以在一秒内扫描上千个端口，并且不容易被网络中的安全设备所发现

Nmap会向目标设备的一个端口发送请求连接的SYN数据包，而目标设备在接收到数据包之后，扫描器不是发送ACK数据包，而是发送RST数据包请求断开连接。这样，三次握手没有完成，无法建立正常的TCP连接。因此，扫描就不会被记录到系统日志中，这种扫描方式一般不会在目标设备上留下扫描痕迹

### 目标设备的回应与目标设备的端口状态（TCP部分）

|  |  |
| --- | --- |
| 目标设备的回应 | 目标设备的端口状态 |
| 如果目标设备给出了一个SYN+ACK回应 | open |
| 如果目标设备给出了一个RST回应 | closed |
| 如果目标设备没有给出回应 | filtered |
| ICMP端口无法抵达错误 | filtered |

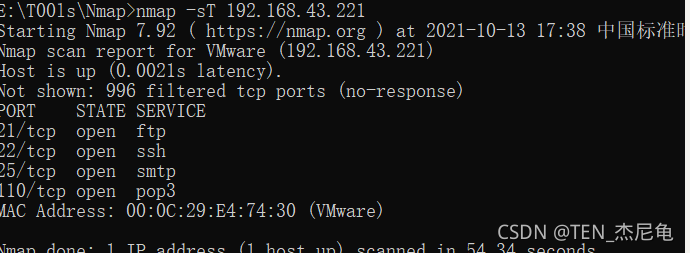
nmap -sS 192.168.43.221

[MISSING IMAGE: , ]

## Connect扫描 --sT

该扫描方式和SYN扫描很相似，但是该扫描完成了TCP的三次握手

nmap -sT 192.168.43.221

[MISSING IMAGE: , ]

## UDP扫描 --sU

当我们对一个端口进行UDP扫描时，端口的状态将是open、close、filtered中的一个，UDP扫描速度很慢

### 目标设备的回应与目标设备的端口状态（UDP部分）

|  |  |
| --- | --- |
| 目标设备的回应 | 目标设备的端口状态 |
| 从目标设备得到任意的UDP回应 | open |
| 如果目标设备没有给出回应 | open |

nmap -sU 192.168.43.221

## 扫描全部端口 --p “\*”

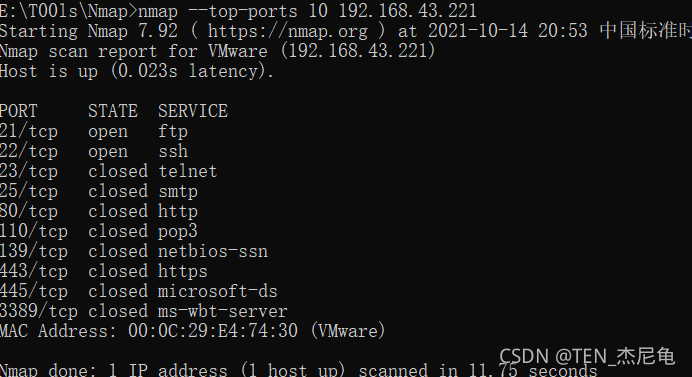
如果对65535个端口进行扫描，则使用参数 -p “\*”

nmap -p “\*” 192.168.43.221

## 扫描使用频率最高的N个端口 --top-ports n

如果只想扫描使用频率最高的n个端口，使用参数 --top-ports n

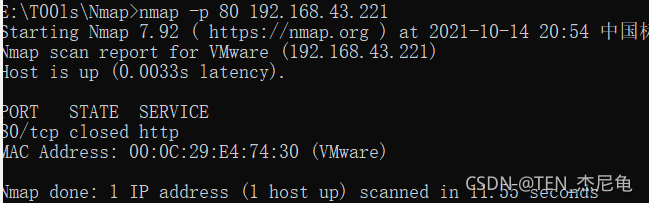
nmap --top-ports 10 192.168.43.221

[MISSING IMAGE: , ]

## 指定扫描端口 --p

如果只想对指定的端口进行扫描，使用参数-p

nmap -p 80 192.168.43.221

[MISSING IMAGE: , ]

## 扫描目标设备的系统

参数

|  |  |
| --- | --- |
| 参数 | 意义 |
| –osscan-limit | 只对满足具有打开和关闭的端口条件的设备进行操作系统扫描 |
| –osscan-guess | 猜测认为最接近目标设备的操作系统类型 |

## 扫描结果保存为XML文件

Nmap支持多种保存格式

将扫描结果保存为XML文件

语法：

nmap -oX [文件名] [IP地址]