nmap

贵州大学

hnzhang1@gzu.edu.cn

December 23, 2018

Overview

Introduction

Nmap ("Network Mapper") is a free and open source (license) utility for network discovery and security auditing.
Nmap uses IP packets to determine

- what hosts are available on the network,
- what services (application name and version) those hosts are offering,
- what operating systems (and OS versions) they are running,
- what type of packet filters/firewalls are in use,
- and dozens of other characteristics.

Host Discovery

基于不同协议的活跃主机发现技术

ARP

ARP 协议的作用是完成逻辑地址和物理地址的转换。几乎所有的网络设备都实现了 ARP 协议。这使得 nmap 在扫描同一网段的设备时，若使用基于 ARP 协议的方法，一般是无法防御的。

Example (命令语法)

```
nmap -PR target
```

执行扫描时，nmap 会构造一个 arp 询问包“who has target tell localhost”，若能收到回应，则说明目标主机 target 是开机状态，否则就是不活跃主机。

ICMP

ICMP 是互联网控制报文协议，用来在 ip 主机和路由器之间传递控制消息，比如差错报告报文和查询报文。基于 ICMP 的活跃主机发现技术使用的是查询报文，可使用的有 3 类：

1. 响应请求和应答（很多主机会禁 ping, 即阻止此类型的 ICMP 报文）用来测试链路及目标主机的 TCP/IP 协议是否正常。
2. 时间戳请求和应答
   ICMP 时间戳请求允许系统向另一个系统查询当前的时间。
3. 地址掩码请求和应答（较少用）
   ICMP 地址掩码请求由源主机发送，用于无盘系统引导时获取自己的子网掩码。虽然 RFC 规定除非是地址掩码的代理，否则不能发送应答，但是有些主机在收到请求时会发送一个应答。

Example (命令语法)

```
nmap -PE target
nmap -PP target
nmap -PM target
```

TCP

TCP(Transmission Control Protocol) 是位于传输层的协议。TCP 是一个面向连接的协议，其特点就是在建立连接的时候会有一个三次握手的过程：

1. 客户端发送 SYN(SEQ=x) 报文给服务器端，进入 SYN_SEND 状态
2. 服务器端收到 SYN 报文后回应一个 SYN(SEQ=y) 和 ACK(SEQ=x+1) 报文，进入 SYN_RECV 状态
3. 客户端收到服务器端的 SYN 报文，回应一个 ACK(SEQ=y+1) 报文，进入 Established 状态。三次握手完成，连接建立。

TCP-SYN/ACK

基于三次握手的过程，nmap 有两种扫描方式：

1. TCP SYN 扫描
   执行 SYN 扫描时，目标主机会认为 nmap 所在主是想与自己建立连接，若此端口是开放的，则会按三次握手规则进行回应，否则会回应一个 RST 数据包拒绝连接。
2. TCP ACK 扫描
   执行 ACK 扫描时，nmap 主机直接发送 ACK 数据包到目标主机，这违反了三次握手机制，目标主机无法接受此数据包，只能回应 RST 数据包来拒绝连接。通常情况下，目标主机上的安全机制会直接过滤掉这个无中生有的 ACK 数据包，这会导致 nmap 误以为目标主机是非活跃的。

Example (命令语法)

```
nmap -PS [port默认80] target
nmap -PA [port默认80] target
```

UDP

UDP(User Datagram Protocol) 是位于传输层的协议。UDP 是一个非面向连接的协议，不能使用连接建立的过程构建扫描。当目标主机上的 UDP 端口收到 UDP 数据包时，若此端口是关闭的，则会给源端发回一个 ICMP 端口不可达的 ICMP 报文，否则就丢弃这个数据包而不做回应。基于 UDP 的主机发现技术缺点如下：

1. 不可靠
   如果目标端口是开放的，nmap 主机不会收到任何回应，会误以为目标主机是非活跃的。(当然，也有可能收不到回应是因为数据包在传输过程中丢失了)
2. 速度慢
   RFC1812 对 ICMP 错误报文的生成速度做出了限制。

Example (命令语法)

```
nmap -PU  target
```

SCTP[1]

SCTP 是位于传输层的协议，其与 TCP 功能类似，其通过 4 次握手建立连接。

1. 客户端发起 INIT 报文
2. 服务器端回应 INIT-ACK（包含 cookie）
3. 客户端回应 COOKIE-ECHO 报文
4. 服务器端回应 COOKIE-ACK 报文

Example (命令语法)

```
nmap -PY  target（发送SCTP INIT数据包）
```

[1] 目前支持此协议的主机并不多

IP

IP 协议是 TCP/IP 协议族中的核心协议，也是 TCP/IP 协议的载体。所有的 TCP、UDP、ICMP、IGMP 数据都是以 IP 数据包的格式进行传输的。nmap 可以向目标主机发送 IP 数据包来检测目标主机是否活跃，可以指定所要使用的协议，若不指定则默认使用 ICMP(1)、IGMP(2)、IP-in-IP(4)。需要注意的是 nmap 发送的这种 IP 数据包是空包，很容易被过滤，可以使用–data-length 参数来发送添加了随机数据的数据包。

Example (命令语法)

```
nmap -PO  target
nmap -PO 1,2,4 target (TCP6 UDP17)
```

主机发现技术的分析

NMAP 到底发送了什么样的数据包

上面的命令执行之后 nmap 到底向目标主机发送了什么样的内容？可以使用-|,-packet-trace 参数来查看。

Example (命令语法)

```
nmap -PS -|,-packet-trace www.gzu.edu.cn
```

Port Scanning

Port Scanning

Port

计算机通过端口来对外提供服务。每个网络设备的端口有 2^{16} 个。端口的分类：

1. 公认端口 (well known port)
   0～1024 就是公认端口，或者是说“常用端口”。这些端口上运行的服务一般都是约定俗成的。比如：web 服务运行在 80 端口，ftp 服务运行在 21 端口，ssh 服务运行在 22 端口。一个正常的应用程序不应该使用公认端口。
2. 注册端口 (registered port)
   这部分端口号的范围是 1025～49151, 实际运行的程序中，大部分在这些端口上注册监听对外提供服务。
3. 动态/私有端口 (dynamic/private port)
   范围是 49152～65535 。一般地，常用服务不会运行在这些端口上，但是由于这些端口不太容易引起人们的注意，一些病毒或木马程序喜欢使用这些端口。

nmap 对端口状态的定义

状态	说明
open	开放的，有服务在此端口上监听
closed	是可访问的，但没有服务在此端口上监听
filtered	无法确定是否开放，数据包可能被安全设备过滤掉了
unfiltered	目标端口是可以访问的，但是无法确定是 open 还是 close
open|filtered	无法确定端口是可以访问的还是被过滤了
closed|filtered	无法确定端口是关闭的还是被过滤了，通常只有在使用 id

基于 TCP 的 SYN 扫描

SYN 扫描是最为流行的一种扫描方式，同时也是 nmap 的默认扫描方式，需要 root 权限。这种扫描方式速度快，通常可以在 1s 内扫描上千个端口，同时也不容易被安全设备发现。
nmap 使用这种方式进行扫描时，在发送了 SYN 包，并收到 SYN/ACK 回应后，直接发送 RST 数据包中断连接。这样的话，一般不会在目标主机上留下痕迹。这次扫描也不会被记录到系统日志中。

目标主机应答	端口状态
ACK/SYN	open
RST	closed
没有应答	filtered
ICMP 无法抵达	filtered

Example (命令语法)

```
nmap -sS target
```

基于 TCP 的 Connect 扫描

这种方式和基于 SYN 的扫描很像，只是本方式完成了三次握手，并且不需要 root 权限。

命令语法

```
nmap -sT target
```

基于 UDP 扫描

UDP 扫描的速度很慢。UDP 程序通常不会对 nmap 发送的空数据包产生回应，这就需要 nmap 在使用不同协议时，构造不同的数据包，比如使用 SNMP 和 DHCP 请求，所使用的数据包就完全不同。nmap 将这些数据包格式存储在 Nmap-service-probes 中。

目标主机应答	端口状态
任意 UDP 应答	open
ICMP 端口无法抵达（类型 3, 代码 3）	closed
目标主机没有应答	open|filtered
ICMP 无法抵达 (类型 3, 代码 1,2,9,10,13)	filtered

Example (命令语法)

```
nmap -sU target
```

基于 TCP 的 FIN 扫描

nmap 向目标主机发送一个 TCP FIN 数据包。根据 RFC793 的规定，对于关闭的端口，目标系统应该返回 RST 标志。

命令语法

```
nmap -sF target
```

基于 TCP 的 NULL 扫描

nmap 向目标主机发送一个不包含任何标志的 TCP 数据包。根据 RFC793 的规定，对于关闭的端口，目标系统应该返回 RST 标志。

命令语法

```
nmap -sN target
```

基于 TCP 的 Xmas Tree 扫描

nmap 向目标主机发送一个含有 FIN、URG、PUSH 标志的数据包。根据 RFC793 的规定，对于关闭的端口，目标系统应该返回 RST 标志。

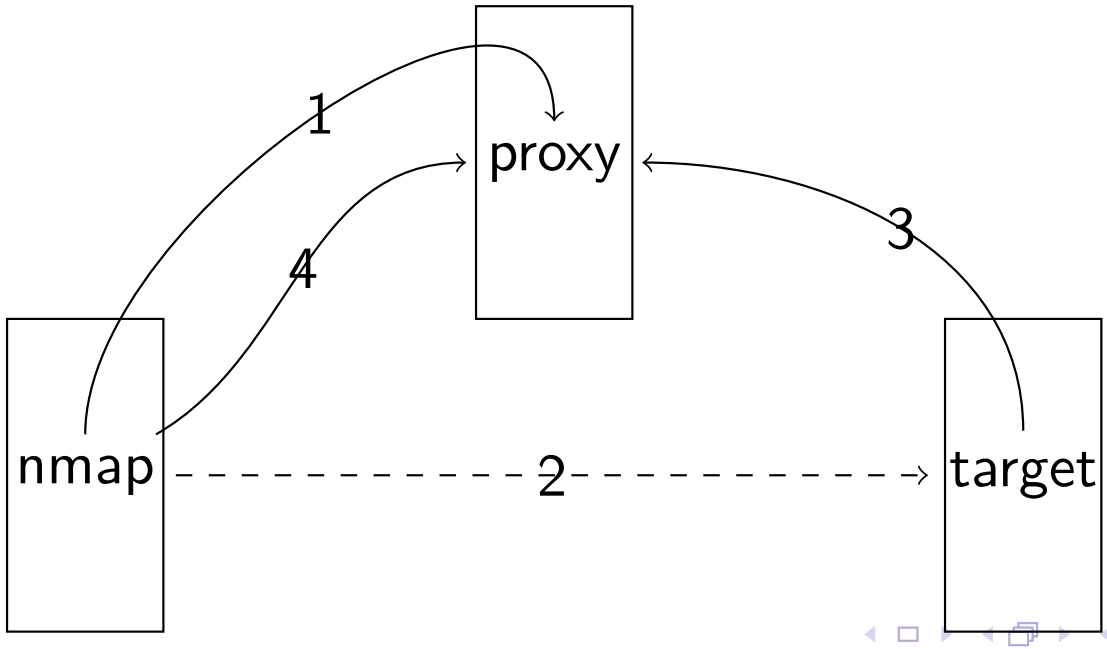
命令语法

```
nmap -sX target
```

idle 扫描

这种方式下，相当于存在一个代理，其流程如下所示。

1. 检测第三方（代理）的 IP ID 值，并记录
2. 构造一个源地址为第三方的数据包，并发送给目标主机
3. 目标主机与第三方通信
4. 检测第三方的 IP ID 值



idle 扫描 II

将第 4 步与第一步记录的 IP ID 值进行比较，应该增加了 1 或 2. 如果是增加了 1, 说明第三方主机在此期间并未向外发送数据包，此时认为目标主机的端口是关闭的；若增加了 2, 说明对外发送过数据包，说明目标主机的端口是开放的。

idle 扫描的不足

- 耗时长
  远长于 SYN 扫描
- 运营商可能会禁止发送伪造的数据包
- 不好寻找理想的第三方主机
  - IP ID 的增长方式是 Incremental
  - 对外通信量不大

idle 扫描 III

命令语法

```
$ sudo nmap -Pn -p- -sI  bbs.nga.cn www.gzu.edu.cn
Starting Nmap 7.70 ( https://nmap.org )
at 2018-12-21 20:06 CST
Idle scan zombie bbs.nga.cn (42.123.103.80) port 80
cannot be used because IP ID sequence class is: All zeros.
Try another proxy.
QUITTING!
```

指定扫描的端口

指定扫描端口

指定端口	参数	语法
扫描常见的100端口	-F target	nmap -F target
扫描某个端口	-p port/name	nmap -p port(s)/name(s)
扫描指定协议的指定端口	-p U:[UDP ports],T:[TCP ports]	nmap -sU -p U:80
扫描所有端口	-p * target	nmap -p * target
扫描常用端口	-|,-top-ports[numbers]	nmap -|,-top-numbers target

OS Detecting and Services discovery

OS Detecting

操作系统检测

Nmap 通过向目标主机发送探针（TCP 或 UDP 数据包），然后检查回应信息中的 IP 标识符（ID）数字时间戳、显示拥塞通知（ECN）、窗口大小等信息来猜测系统信息。Nmap 进行识别的探针和响应对应的关系保存在 Nmap-os-db 中。

远程判断操作系统的方式

- 被动
  不向目标主机发送数据，通过抓包工具来收集网络上的数据包，进而判断目标主机的操作系统
- 主动
  主动向目标主机发送数据，根据远程主机的回应信息进行判断

常用命令

常用命令

```
nmap -F -O target
nmap -O target
nmap -sV -F -|,-fuzzy -|,-osscan-guess target
```

Services discovery

Services discovery

服务发现的意义在于发现目标主机上运行的有哪些服务及相应的版本。

如何进行服务发现

nmap target（此种方式只会判断相关端口是否开放，
然后根据Nmap-services数据库自行把服务关联起来）
nmap -sV target(这种方式会先进行端口扫描，一般是SYN方式；
然后进行服务识别，发送探针报文进行服务识别；
最后进行版本识别，发送探针报文进行服务版本识别。）

Advanced Technical

伪装技术

伪装技术

Nmap 不提供检测和破坏防火墙或 IDS 的专门工具，但却有相关的技术可以实现此目的。

- nmap -f xxx
  将 Nmap 发送的数据包分段 (一个数据包分成若干个，加大检测难度)
- nmap -mtu 16 xxx
  重新指定最大传输单元，必须为 8 的整数倍
- nmap -D <xx1,xx2,...>
  使用诱饵主机隐蔽扫描 (感觉和 idle 扫描差不多)
- nmap -g 20 xxx
  改变源端口，有些防护策略会允许来自某些端口的数据连接，这就导致了危险
- nmap -|,-data-length 30 xxx
  nmap 发送的探测数据包默认是空的，只有包头，这样就很容易被检测并过滤，使用此选项会增加随机的数据内容
- nmap -|,-spoof-mac 0 xxx
  发送以太网包进行探测

格式化输出

格式化输出

Nmap 支持将扫描结果格式化输出为 txt、xml、grep 格式的文件。

- 普通文本文件
  nmap -oN *.txt xxx
- XML 文件
  nmap -oX *.xml xxx
- grep 文件
  nmap -oG *.grep xxx

NSE

NSE 简介

NSE 简介

NSE 脚本最初设计目的是改善服务和主机的侦测工作，但其现在提供越来越多和越来越强大的功能。如今正式版的 NSE 已经包含了 14 个大类的脚本，总数达 500 多个。功能包括：对各种网络口令强度的审计、对各种服务器安全性配置的审计和对服务器漏洞的审计等。

NSE 脚本分类

类别	说明
auth	处理鉴权证书（绕开鉴权）
broadcast	在局域网内探查更多服务开启状态
brute	针对常见应用，如 HTTP/FTP 等进行暴力破解密码
default	提供基本脚本扫描能力，使用-sC 或-A 参数调用
discovery	对网络进行更多的信息收集，如 SMB、SNMP
dos	发起拒绝服务攻击
exploit	对目标系统安全漏洞进行渗透
external	针对第三方服务的脚本
fuzzer	进行模糊测试，发送异常的包到目标主机，探测出潜在漏洞
intrusive	可能会引起目标主机系统崩溃或网络拥塞，容易被 IDS 发现
malware	用来检测恶意软件的脚本
safe	在任何情况下都是安全无害的脚本
version	增强服务与版本扫描功能
vuln	检查目标主机是否有常见漏洞

NSE 脚本的选择

1. -sC/-A
   选择 default 脚本
2. -|,-script
   选择某个脚本，某个脚本类里的所有脚本，某个路径下的一个或所有脚本
   1. nmap -p 80,443 -|,-script http-methods www.gzu.edu.cn
   2. nmap -|,-script safe www.gzu.edu.cn
   3. nmap -|,-script discovery,intrusive www.gzu.edu.cn
   4. nmap -|,-script /Nmap/scripts/banner.NSE www.gzu.edu.cn

向脚本传递参数

Nmap 使--script-args 来传递参数。

示例（改变 http 的 useragent 参数）

nmap --script http-methods -p 80 --script-args http.useragent="Mozilla 42 "www.gzu.edu.cn[a]

[a] nmap 发送数据包所使用的默认客户端一般会被目标主机的安全机制过滤，因此需要传递新的客户端参数。

常用脚本

常用脚本

1. 信息收集类
   eg: http-methods
2. 高级主机发现类
   eg: broadcast-ping, target-sniffer
3. 密码审计类
   eg: mysql-brute, smtp-brute
4. 漏洞扫描类
   eg: http-slowloris, ssl-poodle

信息收集类脚本

http-methods

http-methods 脚本是用来查看目标主机所支持的 HTTP 方法。WEB 服务器需要支持 HTTP 方法，这样才能对外正常提供 HTTP 服务，常见的服务有：

- GET
  请求指定页
- HEAD
  类似 GET，用于获取报头
- POST
  向指定资源提交数据并要求处理（如提交表单或上传文件）
- PUT
  从客户端传送数据到服务器并取代指定文档的内容
- DELETE
  请求服务器删除指定的页面
- OPTIONS
  允许客户端查看服务器的性能
- TRACE
  回显服务器收到的请求，主要用于诊断测试

使用 http-methods 脚本进行审计

审计代码

```
$ nmap -p80 --script http-methods www.gzu.edu.cn
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-23 19:19 CS
Nmap scan report for www.gzu.edu.cn (210.40.12.58)
Host is up (0.0035s latency).

PORT   STATE SERVICE
80/tcp open  http
| http-methods:
|_  Supported Methods: OPTIONS HEAD GET POST

Nmap done: 1 IP address (1 host up) scanned in 0.66 seconds
```

http-methods 审计结果说明

对 www.gzu.edu.cn 网站的审计结果表明，服务器支持如下方法：OPTIONS HEAD GET POST。
通常来说我们认为存在风险的方法有：TRACE[2]、CONNECT、PUT 和 DELETE。

[2]TRACE 方法，可能会导致 Cross Site Tracing(XST) 攻击。攻击者将恶意代码嵌入到支持此方法的 WEB 服务器上的 web 页面中，当用户去浏览此网页时，恶意代码在用户浏览器中执行，然后就会把用户的 Cookie 和 http 基本验证信息发送到服务器，同时传送 Trace 请求给目标主机，导致 cookie 欺骗或者是中间人攻击。

高级主机发现类脚本

broadcast-ping

Nmap 中也有发现本地网中活跃主机的功能，nmap 是向本地网络中每一个 IP 地址发送单播探针数据包来实现的，而 broadcast-ping 脚本是通过发送广播探针数据包来实现的。

使用 broadcast-ping 脚本进行审计

审计代码

```
$ nmap --script broadcast-ping 10.11.38.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-23 19:41 CS
NSE: [broadcast-ping] not running for lack of privileges.
Nmap scan report for 10.11.38.4
Host is up (0.00036s latency).

Nmap scan report for 10.11.38.10
Host is up (0.016s latency).

Nmap scan report for 10.11.38.254
Host is up (0.011s latency).

Nmap done: 256 IP addresses (3 hosts up) scanned in 19.23 seco
```

targets-sniffer

targets-sniffer 脚本是通过对所在网络进行嗅探，从而发现网络中的主机的。

使用 targets-sniffer 脚本进行审计

审计代码

```
nmap --script targets-sniffer 10.11.38.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-23 19:51 CS
Nmap scan report for 10.11.38.4
Host is up (0.00031s latency).

Nmap scan report for 10.11.38.10
Host is up (0.015s latency).

Nmap scan report for 10.11.38.11
Host is up (0.031s latency).

Nmap scan report for 10.11.38.254
Host is up (0.012s latency).

Nmap done: 256 IP addresses (4 hosts up) scanned in 19.38 seco
```

密码审计类脚本

mysql-brute

网络上的服务一般都会使用一些认证措施，使用最广泛的还是用户名和密码这种方式，但其存在的缺点就是有些用户安全意识不到位，选择的密码强度不够，或干脆直接使用默认密码，这就带来了安全风险。
mysql-brute 是针对 mysql 数据库的密码强度审计脚本。
smtp-brute 是针对邮件服务器的 SMTP 服务的密码强度审计脚本。
注意脚本中的 brute，这是暴力破解的意思，nmap 会使用这此脚本提供的用户名和密码去一一尝试登陆。

使用 mysql-brute 脚本进行审计

```
$ nmap --script mysql-brute localhost
```

漏洞扫描类脚本

http-slowloris

http-slowloris 以极低的速度向目标服务器发送 http 请求，由于 web server 对于并发的连接数都有一定的上限，因此，如果恶意占用这些连接不释放，就会导致服务器无法处理新的请求，从而导致拒绝服务攻击。

使用 http-slowloris 脚本进行审计

```
$ nmap --script http-slowloris --max-parallelism 300 -p80
    xxx
```

ssl-poodle

SSL 3.0 中的 POODLE 漏洞[3]可以被攻击者利用来窃取 SSL 3.0 加密通信过程中的通信内容。Nmap 中的 ssl-poodle 脚本可以用来检测目标服务器上是否存在 POODLE 漏洞。

使用 ssl-poodle 脚本进行审计

```
$ nmap -sV --version-all --script ssl-poodle -p443 xxx
```

[3] CVE-2014-3566, 谷歌团队发现。

Q&A