Linux 系统中 nmap 网络扫描工具的命令用法详解

1、描述

nmap 是一个网络探测和安全扫描程序 ,系统管理者和个人可以使用这个软件扫描大型的网络 , 获取那台主机正在运行以及提供什么服务等信息。nmap 支持很多扫描技术，例如：UDP、TCP connect()、TCP SYN(半开扫描)、ftp 代理(bounce 攻击)、反向标志、ICMP、FIN、ACK 扫描、圣诞树(Xmas Tree)、SYN 扫描和 null 扫描。从扫描类型一节可以得到细节。nmap 还提供了一些高级的特征，例如：通过 TCP/IP 协议栈特征探测操作系统类型，秘密扫描，动态延时和重传计算，并行扫描，通过并行 ping 扫描探测关闭的主机，诱饵扫描，避开端口过滤检测 , 直接 RPC 扫描(无须端口影射) , 碎片扫描 , 以及灵活的目标和端口设定。

为了提高 nmap 在 non-root 状态下的性能 ,软件的设计者付出了很大的努力。很不幸，一些内核界面(例如 raw socket)需要在 root 状态下使用。所以应该尽可能在 root 使用 nmap。

nmap 运行通常会得到被扫描主机端口的列表。nmap 总会给出 well known 端口的服务名(如果可能)、端口号、状态和协议等信息。每个端口的状态有：open、filtered、unfiltered。open 状态意味着目标主机能够在这个端口使用 accept()系统调用接受连接。filtered 状态表示：防火墙、包过滤和其它

的网络安全软件掩盖了这个端口，禁止 nmap 探测其是否打开。unfiltered 表示：这个端口关闭，并且没有防火墙/包过滤软件来隔离 nmap 的探测企图。通常情况下，端口的状态基本都是 unfiltered 状态，只有在大多数被扫描的端口处于 filtered 状态下，才会显示处于 unfiltered 状态的端口。

根据使用的功能选项，nmap 也可以报告远程主机的下列特征：使用的操作系统、TCP 序列、运行绑定到每个端口上的应用程序的用户名、DNS 名、主机地址是否是欺骗地址、以及其它一些东西。

2、功能选项

功能选项可以组合使用。一些功能选项只能够在某种扫描模式下使用。nmap 会自动识别无效或者不支持的功能选项组合，并向用户发出警告信息。

如果你是有经验的用户，可以略过结尾的示例一节。可以使用 nmap -h 快速列出功能选项的列表。

2.1 扫描类型

-sT

TCP connect()扫描：这是最基本的 TCP 扫描方式。connect()是一种系统调用，由操作系统提供，用来打开一个连接。如果目标端口有程序监听，connect()就会成功返回，否则这个端口是不可达的。这项技术最大的优点是，你勿需 root 权限。任何 UNIX 用户都可以自由使用这个系统调用。这种扫描很容易被检测到，在目标主机的日志中会记录大批的连接请求以及错误信息。

-sS

TCP 同步扫描(TCP SYN)：因为不必全部打开一个 TCP 连接，所以这项技术通常称为半开扫描(half-open)。你可以发出一个 TCP 同步包(SYN)，然后等待回应。如果对方返回 SYN|ACK(响应)包就表示目标端口正在监听；如果返回 RST 数据包，就表示目标端口没有监听程序；如果收到一个 SYN|ACK 包，源主机就会马上发出一个 RST(复位)数据包断开和目标主机的连接，这实际上有我们的操作系统内核自动完成的。这项技术最大的好处是，很少有系统能够把这记入系统日志。不过，你需要 root 权限来定制 SYN 数据包。

-sF -sF -sN

秘密 FIN 数据包扫描、圣诞树(Xmas Tree)、空(Null)扫描模式：即使 SYN 扫描都无法确定的情况下使用。一些防火墙和包过滤软件能够对发送到被限制端口的 SYN 数据包进行监视，而且有些程序比如 synlogger 和 courtney 能够检测那些扫描。这些高级的扫描方式可以逃过这些干扰。
这些扫描方式的理论依据是：关闭的端口需要对你的探测包回应 RST 包，而打开的端口必需忽略有问题的包(参考 RFC 793 第 64 页)。FIN 扫描使用暴露的 FIN 数据包来探测，而圣诞树扫描打开数据包的 FIN、URG 和 PUSH 标志。不幸的是，微软决定完全忽略这个标准，另起炉灶。所以这种扫描方式对 Windows95/NT 无效。不过，从另外的角度讲，可以使用这种方式来分别两种不同的平台。如果使用这种扫描方式可以发现打开的端口，你就可以确定目标注意运行的不是 Windows 系统。如果使用-sF、-sX 或者-sN 扫描显示所有的端口都是关闭的，而使用 SYN 扫描显示有打开的端口，你可以确定目标主机可能运

行的是 Windwos 系统。现在这种方式没有什么太大的用处，因为 nmap 有内嵌的操作系统检测功能。还有其它几个系统使用和 windows 同样的处理方式，包括 Cisco、BSDI、HP/UX、MYS、IRIX。在应该抛弃数据包时，以上这些系统都会从打开的端口发出复位数据包。

-sP

ping 扫描：有时你只是想知道此时网络上哪些主机正在运行。通过向你指定的网络内的每个 IP 地址发送 ICMP echo 请求数据包 ,nmap 就可以完成这项任务。如果主机正在运行就会作出响应。不幸的是，一些站点例如 :microsoft.com 阻塞 ICMP echo 请求数据包。然而，在默认的情况下 nmap 也能够向 80 端口发送 TCP ack 包，如果你收到一个 RST 包，就表示主机正在运行。nmap 使用的第三种技术是：发送一个 SYN 包，然后等待一个 RST 或者 SYN/ACK 包。对于非 root 用户，nmap 使用 connect()方法。

在默认的情况下(root 用户)，nmap 并行使用 ICMP 和 ACK 技术。

注意，nmap 在任何情况下都会进行 ping 扫描，只有目标主机处于运行状态，才会进行后续的扫描。如果你只是想知道目标主机是否运行，而不想进行其它扫描，才会用到这个选项。

-sU

UDP 扫描：如果你想知道在某台主机上提供哪些 UDP(用户数据报协议,RFC768)服务，可以使用这种扫描方法。nmap 首先向目标主机的每个端口

发出一个 0 字节的 UDP 包，如果我们收到端口不可达的 ICMP 消息，端口就是关闭的，否则我们就假设它是打开的。

有些人可能会想 UDP 扫描是没有什么意思的。但是，我经常会想到最近出现的 solaris rpcbind 缺陷。rpcbind 隐藏在一个未公开的 UDP 端口上，这个端口号大于 32770。所以即使端口 111(portmap 的众所周知端口号)被防火墙阻塞有关系。但是你能发现大于 30000 的哪个端口上有程序正在监听吗?使用 UDP 扫描就能！cDc Back Orifice 的后门程序就隐藏在 Windows 主机的一个可配置的 UDP 端口中。不考虑一些通常的安全缺陷，一些服务例如:snmp、tftp、NFS 使用 UDP 协议。不幸的是，UDP 扫描有时非常缓慢，因为大多数主机限制 ICMP 错误信息的比例(在 RFC1812 中的建议)。例如，在 Linux 内核中(在 net/ipv4/icmp.h 文件中)限制每 4 秒钟只能出现 80 条目标豢纱锏肾 CMP 消息，如果超过这个比例，就会给 1/4 秒钟的处罚。solaris 的限制更加严格，每秒钟只允许出现大约 2 条 ICMP 不可达消息，这样，使扫描更加缓慢。nmap 会检测这个限制的比例，减缓发送速度，而不是发送大量的将被目标主机丢弃的无用数据包。

不过 Micro$oft 忽略了 RFC1812 的这个建议，不对这个比例做任何的限制。所以我们可以能够快速扫描运行 Win95/NT 的主机上的所有 65K 个端口。

-sA

ACK 扫描：这项高级的扫描方法通常用来穿过防火墙的规则集。通常情况下，这有助于确定一个防火墙是功能比较完善的或者是一个简单的包过滤程序，只是阻塞进入的 SYN 包。

这种扫描是向特定的端口发送 ACK 包(使用随机的应答/序列号)。如果返回一个 RST 包，这个端口就标记为 unfiltered 状态。如果什么都没有返回，或者返回一个不可达 ICMP 消息，这个端口就归入 filtered 类。注意，nmap 通常不输出 unfiltered 的端口，所以在输出中通常不显示所有被探测的端口。显然，这种扫描方式不能找出处于打开状态的端口。

-sW

对滑动窗口的扫描：这项高级扫描技术非常类似于 ACK 扫描，除了它有时可以检测到处于打开状态的端口，因为滑动窗口的大小是不规则的，有些操作系统可以报告其大小。这些系统至少包括：某些版本的 AIX、Amiga、BeOS、BSDI、Cray、Tru64 UNIX、DG/UX、OpenVMS、Digital UNIX、OpenBSD、OpenStep、QNX、Rhapsody、SunOS 4.x、Ultrix、VAX、VXWORKS。从 nmap-hackers 邮件 3 列表的文档中可以得到完整的列表。

-sR

RPC 扫描。这种方法和 nmap 的其它不同的端口扫描方法结合使用。选择所有处于打开状态的端口向它们发出 SunRPC 程序的 NULL 命令，以确定它们是否是 RPC 端口，如果是，就确定是哪种软件及其版本号。
因此你能够获得防火墙的一些信息。诱饵扫描现在还不能和 RPC 扫描结合使用。

-b

FTP 反弹攻击(bounce attack):FTP 协议(RFC 959)有一个很有意思的特征，它支持代理 FTP 连接。也就是说，我能够从 evil.com 连接到 FTP 服务器 target.com，并且可以要求这台 FTP 服务器为自己发送 Internet 上任何地方的文件！1985 年，RFC959 完成时，这个特征就能很好地工作了。然而，在今天的 Internet 中，我们不能让人们劫持 FTP 服务器，让它向 Internet 上的任意节点发送数据。如同 Hobbit 在 1995 年写的文章中所说的，这个协议"能够用来做投递虚拟的不可达邮件和新闻，进入各种站点的服务器,填满硬盘，跳过防火墙，以及其它的骚扰活动，而且很难进行追踪"。我们可以使用这个特征，在一台代理 FTP 服务器扫描 TCP 端口。因此，你需要连接到防火墙后面的一台 FTP 服务器，接着进行端口扫描。如果在这台 FTP 服务器中有可读写的目录，你还可以向目标端口任意发送数据(不过 nmap 不能为你做这些)。

传递给-b 功能选项的参数是你要作为代理的 FTP 服务器。语法格式为：

-b username:password@server:port。

除了 server 以外，其余都是可选的。如果你想知道什么服务器有这种缺陷，可以参考我在 Phrack 51 发表的文章。还可以在 nmap 的站点得到这篇文章的最新版本。

2.2 通用选项

这些内容不是必需的，但是很有用。

-P0

在扫描之前，不必 ping 主机。有些网络的防火墙不允许 ICMP echo 请求穿过，使用这个选项可以对这些网络进行扫描。microsoft.com 就是一个例子，因此在扫描这个站点时，你应该一直使用-P0 或者-PT 80 选项。

-PT

扫描之前，使用 TCP ping 确定哪些主机正在运行。nmap 不是通过发送 ICMP echo 请求包然后等待响应来实现这种功能，而是向目标网络(或者单一主机)发出 TCP ACK 包然后等待回应。如果主机正在运行就会返回 RST 包。只有在目标网络/主机阻塞了 ping 包，而仍旧允许你对其进行扫描时，这个选项才有效。对于非 root 用户，我们使用 connect()系统调用来实现这项功能。使用-PT 来设定目标端口。默认的端口号是 80，因为这个端口通常不会被过滤。

-PS

对于 root 用户，这个选项让 nmap 使用 SYN 包而不是 ACK 包来对目标主机进行扫描。如果主机正在运行就返回一个 RST 包(或者一个 SYN/ACK 包)。

-PI

设置这个选项，让 nmap 使用真正的 ping(ICMP echo 请求)来扫描目标主机是否正在运行。使用这个选项让 nmap 发现正在运行的主机的同时，nmap 也会对你的直接子网广播地址进行观察。直接子网广播地址一些外部可达的 IP 地

址，把外部的包转换为一个内向的 IP 广播包，向一个计算机子网发送。这些 IP 广播包应该删除，因为会造成拒绝服务攻击(例如 smurf)。

-PB

这是默认的 ping 扫描选项。它使用 ACK(-PT)和 ICMP(-PI)两种扫描类型并行扫描。如果防火墙能够过滤其中一种包，使用这种方法，你就能够穿过防火墙。

-O

这个选项激活对 TCP/IP 指纹特征(fingerprinting)的扫描 ,获得远程主机的标志。换句话说，nmap 使用一些技术检测目标主机操作系统网络协议栈的特征。nmap 使用这些信息建立远程主机的指纹特征 ,把它和已知的操作系统指纹特征数据库做比较，就可以知道目标主机操作系统的类型。

-I

这个选项打开 nmap 的反向标志扫描功能。Dave Goldsmith 1996 年向 bugtap 发出的邮件注意到这个协议，ident 协议(rfc 1413)允许使用 TCP 连接给出任何进程拥有者的用户名，即使这个进程并没有初始化连接。例如，你可以连接到 HTTP 端口，接着使用 identd 确定这个服务器是否由 root 用户运行。这种扫描只能在同目标端口建立完全的 TCP 连接时(例如：-sT 扫描选项)才能成功。使用-I 选项是，远程主机的 identd 精灵进程就会查询在每个打开的端口上监听的进程的拥有者。显然，如果远程主机没有运行 identd 程序，这种扫描方法无效。

-f

这个选项使 nmap 使用碎片 IP 数据包发送 SYN、FIN、XMAS、NULL。使用碎片数据包增加包过滤、入侵检测系统的难度，使其无法知道你的企图。不过，要慎重使用这个选项！有些程序在处理这些碎片包时会有麻烦，我最喜欢的嗅探器在接受到碎片包的头 36 个字节时，就会发生 segmentation faulted。因此，在 nmap 中使用了 24 个字节的碎片数据包。虽然包过滤器和防火墙不能防这种方法，但是有很多网络出于性能上的考虑，禁止数据包的分片。
注意这个选项不能在所有的平台上使用。它在 Linux、FreeBSD、OpenBSD 以及其它一些 UNIX 系统能够很好工作。

-v

冗余模式。强烈推荐使用这个选项，它会给出扫描过程中的详细信息。使用这个选项，你可以得到事半功倍的效果。使用-d 选项可以得到更加详细的信息。

-h

快速参考选项。

-oN

把扫描结果重定向到一个可读的文件 logfilename 中。

-oM

把扫描结果重定向到 logfilename 文件中，这个文件使用主机可以解析的语法。你可以使用-oM -来代替 logfilename，这样输出就被重定向到标准输出 stdout。在这种情况下，正常的输出将被覆盖，错误信息茬苒可以输出到标准错误 stderr。要注意，如果同时使用了-v 选项，在屏幕上会打印出其它的信息。

-oS thIs l0gz th3 r3suLtS of YouR ScanZ iN a s| THe fiL3 U sPecfy 4s an arGuMEnT! U kAn gIv3 the 4rgument -

(wItHOUt qUOteZ) to sh00t output iNT0 stDouT!@!! 莫名其妙，下面是我猜着翻译的，相形字？

把扫描结果重定向到一个文件 logfilename 中，这个文件使用一种"黑客方言"的语法形式(作者开的玩笑?)。同样，使用-oS -就会把结果重定向到标准输出上。

-resume

某个网络扫描可能由于 control-C 或者网络损失等原因被中断 ,使用这个选项可以使扫描接着以前的扫描进行。logfilename 是被取消扫描的日志文件，它必须是可读形式或者机器可以解析的形式。而且接着进行的扫描不能增加新的选项，只能使用与被中断的扫描相同的选项。nmap 会接着日志文件中的最后一次成功扫描进行新的扫描。

-iL

从 inputfilename 文件中读取扫描的目标。在这个文件中要有一个主机或者网络的列表，由空格键、制表键或者回车键作为分割符。如果使用-iL -，nmap 就会从标准输入 stdin 读取主机名字。你可以从指定目标一节得到更加详细的信息。

-iR

让 nmap 自己随机挑选主机进行扫描。

-p

这个选项让你选择要进行扫描的端口号的范围。例如，-p 23 表示：只扫描目标主机的 23 号端口。-p 20-30,139,60000-表示：扫描 20 到 30 号端口，139 号端口以及所有大于 60000 的端口。在默认情况下，nmap 扫描从 1 到 1024 号以及 nmap-services 文件(如果使用 RPM 软件包，一般在/usr/share/nmap/目录中)中定义的端口列表。

-F

快速扫描模式，只扫描在 nmap-services 文件中列出的端口。显然比扫描所有 65535 个端口要快。

-D

使用诱饵扫描方法对目标网络/主机进行扫描。如果 nmap 使用这种方法对目标网络进行扫描，那么从目标主机/网络的角度来看，扫描就象从其它主机

(decoy1,等)发出的。从而，即使目标主机的 IDS(入侵检测系统)对端口扫描发出报警，它们也不可能知道哪个是真正发起扫描的地址，哪个是无辜的。这种扫描方法可以有效地对付例如路由跟踪、response-dropping 等积极的防御机制，能够很好地隐藏你的 IP 地址。

每个诱饵主机名使用逗号分割开，你也可以使用 ME 选项，它代表你自己的主机，和诱饵主机名混杂在一起。如果你把 ME 放在第六或者更靠后的位置，一些端口扫描检测软件几乎根本不会显示你的 IP 地址。如果你不使用 ME 选项，nmap 会把你的 IP 地址随机夹杂在诱饵主机之中。

注意:你用来作为诱饵的主机应该正在运行或者你只是偶尔向目标发送 SYN 数据包。很显然，如果在网络上只有一台主机运行，目标将很轻松就会确定是哪台主机进行的扫描。或许，你还要直接使用诱饵的 IP 地址而不是其域名，这样诱饵网络的域名服务器的日志上就不会留下关于你的记录。

还要注意：一些愚蠢的端口扫描检测软件会拒绝路由试图进行端口扫描的主机。因而，你需要让目标主机和一些诱饵断开连接。如果诱饵是目标主机的网关或者就是其自己时 ,会给目标主机造成很大问题。所以你需要慎重使用这个选项。

诱饵扫描既可以在起始的 ping 扫描也可以在真正的扫描状态下使用。它也可以和-O 选项组合使用。

使用太多的诱饵扫描能够减缓你的扫描速度甚至可能造成扫描结果不正确。同时 ,有些 ISP 会把你的欺骗包过滤掉。虽然现在大多数的 ISP 不会对此进行限制。

-S

在一些情况下，nmap 可能无法确定你的源地址(nmap 会告诉你)。在这种情况下，可以使用这个选项给出你的 IP 地址。

在欺骗扫描时，也使用这个选项。使用这个选项可以让目标认为是其它的主机对自己进行扫描。

-e

告诉 nmap 使用哪个接口发送和接受数据包。nmap 能够自动对此接口进行检测，如果无效就会告诉你。

-g

设置扫描的源端口。一些天真的防火墙和包过滤器的规则集允许源端口为 DNS(53)或者 FTP-DATA(20)的包通过和实现连接。显然，如果攻击者把源端口修改为 20 或者 53，就可以摧毁防火墙的防护。在使用 UDP 扫描时，先使用 53 号端口；使用 TCP 扫描时，先使用 20 号端口。注意只有在能够使用这个端口进行扫描时，nmap 才会使用这个端口。例如，如果你无法进行 TCP 扫描，nmap 会自动改变源端口，即使你使用了-g 选项。

对于一些扫描，使用这个选项会造成性能上的微小损失，因为我有时会保存关于特定源端口的一些有用的信息。

-r

告诉 nmap 不要打乱被扫描端口的顺序。

--randomize_hosts

使 nmap 在扫描之前，打乱每组扫描中的主机顺序，nmap 每组可以扫描最多 2048 台主机。这样，可以使扫描更不容易被网络监视器发现，尤其和 --scan_delay 选项组合使用，更能有效避免被发现。

-M

设置进行 TCP connect()扫描时，最多使用多少个套接字进行并行的扫描。使用这个选项可以降低扫描速度，避免远程目标宕机。

2.3 适时选项

通常，nmap 在运行时，能够很好地根据网络特点进行调整。扫描时，nmap 会尽量减少被目标检测到的机会，同时尽可能加快扫描速度。然而，nmap 默认的适时策略有时候不太适合你的目标。使用下面这些选项，可以控制 nmap 的扫描 timing：

-T

设置 nmap 的适时策略。Paranoid:为了避开 IDS 的检测使扫描速度极慢，nmap 串行所有的扫描，每隔至少 5 分钟发送一个包；Sneaky：也差不多，只是数据包的发送间隔是 15 秒；Polite：不增加太大的网络负载，避免宕掉目标主机，串行每个探测，并且使每个探测有 0.4 秒种的间隔；Normal:nmap 默认

的选项，在不是网络过载或者主机/端口丢失的情况下尽可能快速地扫描；Aggressive:设置 5 分钟的超时限制，使对每台主机的扫描时间不超过 5 分钟，并且使对每次探测回应的等待时间不超过 1.5 秒钟；b>Insane:只适合快速的网络或者你不在意丢失某些信息，每台主机的超时限制是 75 秒，对每次探测只等待 0.3 秒钟。你也可是使用数字来代替这些模式，例如：-T 0 等于-T Paranoid，-T 5 等于-T Insane。

这些适时模式不能下面的适时选项组合使用。

--host_timeout

设置扫描一台主机的时间，以毫秒为单位。默认的情况下，没有超时限制。

--max_rtt_timeout

设置对每次探测的等待时间，以毫秒为单位。如果超过这个时间限制就重传或者超时。默认值是大约 9000 毫秒。

--min_rtt_timeout

当目标主机的响应很快时，nmap 就缩短每次探测的超时时间。这样会提高扫描的速度，但是可能丢失某些响应时间比较长的包。使用这个选项，可以让 nmap 对每次探测至少等待你指定的时间，以毫秒为单位。

--initial_rtt_timeout

设置初始探测的超时值。一般这个选项只在使用-P0 选项扫描有防火墙保护的主机才有用。默认值是 6000 毫秒。

--max_parallelism

设置最大的并行扫描数量。--max_parallelism 1 表示同时只扫描一个端口。这个选项对其它的并行扫描也有效，例如 ping sweep, RPC scan。

--scan_delay

设置在两次探测之间，nmap 必须等待的时间。这个选项主要用于降低网络的负载。

2.4 目标设定

在 nmap 的所有参数中，只有目标参数是必须给出的。其最简单的形式是在命令行直接输入一个主机名或者一个 IP 地址。如果你希望扫描某个 IP 地址的一个子网，你可以在主机名或者 IP 地址的后面加上/掩码。掩码在 0(扫描整个网络)到 32(只扫描这个主机)。使用/24 扫描 C 类地址，/16 扫描 B 类地址。

除此之外，nmap 还有更加强大的表示方式让你更加灵活地指定 IP 地址。例如，如果要扫描这个 B 类网络 128.210.*.*，你可以使用下面三种方式来指定这些地址:128.210.*.*、128.21-.0-255.0-255 或者 128.210.0.0/16 这三种形式是等价的。

3.示例

nmap -sP 192.168.1.0/24

#进行 ping 扫描 ,打印出对扫描做出响应的主机,不做进一步测试(如端口扫描或者操作系统探测)

nmap -sL 192.168.1.0/24

#仅列出指定网络上的每台主机，不发送任何报文到目标主机

nmap -PS 192.168.1.234

#探测目标主机开放的端口 ,可以指定一个以逗号分隔的端口列表(如-PS22 ,23，25，80)

nmap -PU 192.168.1.0/24

#使用 UDP ping 探测主机

nmap -sS 192.168.1.0/24

#使用频率最高的扫描选项：SYN 扫描,又称为半开放扫描，它不打开一个完全的 TCP 连接，执行得很快

nmap -sT 192.168.1.0/24

#当 SYN 扫描不能用时，TCP Connect()扫描就是默认的 TCP 扫描

nmap -sU 192.168.1.0/24

#UDP 扫描用-sU 选项,UDP 扫描发送空的(没有数据)UDP 报头到每个目标端口

nmap -sO 192.168.1.19

#确定目标机支持哪些 IP 协议 (TCP，ICMP，IGMP 等)

nmap -O 192.168.1.19

#探测目标主机的操作系统

nmap -A 192.168.1.19

#探测目标主机的操作系统

nmap -v scanme.nmap.org

#这个选项扫描主机 scanme.nmap.org 中 所有的保留 TCP 端口。选项-v 启用细节模式。

nmap -sS -O scanme.nmap.org/24

#进行秘密 SYN 扫描 ,对象为主机 Saznme 所在的 “C 类” 网段 的 255 台主机。同时尝试确定每台工作主机的操作系统类型。因为进行 SYN 扫描 和操作系统检测，这个扫描需要有根权限。

nmap -sV -p 22，53，110，143，4564 198.116.0-255.1-127

#进行主机列举和 TCP 扫描，对象为 B 类 188.116 网段中 255 个 8 位子网。这个测试用于确定系统是否运行了 sshd、DNS、imapd 或 4564 端口。如果这些端口 打开，将使用版本检测来确定哪种应用在运行。

nmap -v -iR 100000 -P0 -p 80

#随机选择 100000 台主机扫描是否运行 Web 服务器(80 端口)。由起始阶段 发送探测报文来确定主机是否工作非常浪费时间，而且只需探测主机的一个端口，因 此使用-P0 禁止对主机列表。

nmap -P0 -p80 -oX logs/pb-port80scan.xml -oG logs/pb-port80scan.gnmap 216.163.128.20/20

#扫描 4096 个 IP 地址，查找 Web 服务器(不 ping)，将结果以 Grep 和 XML 格式保存。

host -l company.com | cut -d -f 4 | nmap -v -iL -

#进行 DNS 区域传输，以发现 company.com 中的主机，然后将 IP 地址提供给 Nmap。上述命令用于 GNU/Linux -- 其它系统进行区域传输时有不同的命令