参考:

网络与安全工具：*Hping*

目录

类别：

CS-CNS：计算机网络安全

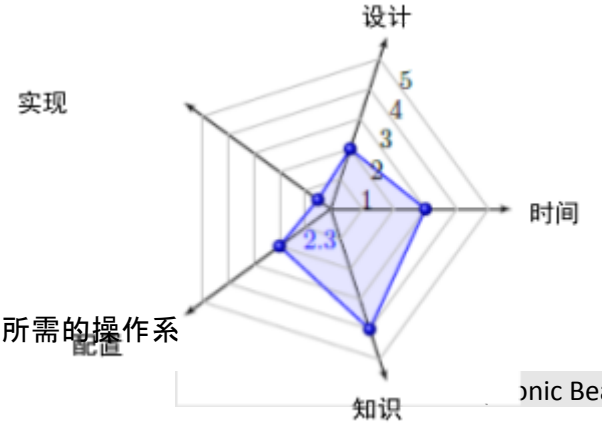
目标：

1. 了解网络测试工具 hping3
2. 使用 hping3 进行网络扫描和模拟 DoS 攻击

实验室的预计时长：

1. 行家：120 分钟
2. 新手：480 分钟

难度示意图：



衡量指标	数值 (0-5)
时间	3
设计	2
实现	0.5
配置	2
知识	4
分数（平均）	2.3

难度表格。

所需的操作系

nic Beaver)

实验室运行环境：

3. 网关：Linux (Ubuntu 18.04 LTS)

4. 网络设置：
通过 Net 3 连接互联网：172.16.0.0/12
客户端 Net 1：192.168.0.0/24
服务器端 Net 2：10.0.0.0/8

实验室概述

Hping（当前版本为 hping3）是一个面向命令行的 TCP/IP 数据包汇编器/分析器。该接口受 ping Unix 命令启发，但 hping 不仅能够发送ICMP 回显请求。它还支持 TCP、UDP、ICMP 和 RAW-IP 协议，具有路由跟踪模式，能够在覆盖通道之间发送文件，以及许多其他功能。

总之，学生将执行以下操作：

- 使用 hping3 发送自定义数据包以进行网络诊断和测试
- 使用 hping3 发送自定义数据包以模拟攻击

1. hping3 的背景

hping3 通常称为“数据包制作工具”，这意味着它可以创建你能想到的任何类型的数据包。这在进行侦察时非常有用，因为不同的数据包将从操作系统 TCP/IP 堆栈中生成不同的响应，从而为我们提供有关基础操作系统、端口和服务的线索。此外，我们还可测试各种 IDS 和防火墙规避技术，例如分片数据包、慢扫描等。你可以使用 hping3 完成的一部分工作包括：

- 防火墙和 IDS 测试
- 高级端口扫描
- 使用不同的协议、TOS、分片进行网络测试
- 手动路径 MTU 发现
- 所有受支持协议下的高级路由跟踪
- 远程操作系统指纹识别
- 远程正常运行时间猜测
- TCP/IP 堆栈审计
- hping 对学习 TCP/IP 的学生也很有用。

若要检查你的系统上是否已安装 hping3，请发出以下命令：

```
$       hping3 -v
```

你可能会看到以下反馈，表明你的系统已安装 hping3：

```
hping3 version 3.0.0-alpha-2 ($ID: release.h,v 1.4 2004/04/09 23:38:56 antirez Exp $)
此二进制文件支持 TCL 脚本编写
```

如果未安装，你可以发出以下命令，以在你的系统上安装 hping3：

```
$       sudo apt install hping3
```

你可以键入以下命令以显示 hping3 选项：

```
$       hping3 -h
```

输出将如下所示：

```
用法: hping3 主机 [选项]
      -h    --help      显示此帮助
      -v    --version   显示版本
      -c    --count     数据包计数
```

```
    -I    --interval wait (uX 为 X 微妙，例如 -i u1000)
          --fast      -i u10000 的别名 (每秒 10 个数据包)
          --faster    -i u1000 的别名 (每秒 100 个数据包)
          --flood     尽快发送数据包。不显示回复。
    -n    --numeric   数字输出
    -q    --quiet     静默
    -I    --interface 接口名称 (否则为默认路由接口)
    -V    --verbose   详细模式
    -D    --debug     调试信息
    -z    --bind      将 ctrl+z 绑定到 ttl              (默认为 dst 端口)
    -Z    --unbind    取消绑定 ctrl+z
          --beep      收到每个匹配的数据包时都发出蜂鸣声
模式
    默认模式          TCP
    -0    --rawip     RAW IP 模式
    -1    --icmp      ICMP 模式
    -2    --udp       UDP 模式
    -8    --scan      SCAN 模式。
                      示例：hping --scan 1-30,70-90 -S www.target.host
    -9    --listen    侦听模式
IP
    -a    --spoof           伪造源地址
          --rand-dest       随机目标地址模式。查看手册。
          --rand-source     随机源地址模式。查看手册。
    -t    --ttl             ttl (默认为 64)
    -N    --id              id (默认为随机值)
    -W    --winid           使用 win* id 字节顺序
    -r    --rel             使 id 字段相对化        (用于评估主机流量)
    -f    --frag            将数据包拆分为更多分片 (可能传递弱 acl)
    -x    --morefrag        设置更多分片标志
    -y    --dontfrag        设置不分片标志
    -g    --fragoff         设置分片偏移
    -m    --mtu             设置虚拟MTU值，当大于mtu的时候分段
    -o    --tos             服务类型 (默认为 0x00)，尝试 --tos 帮助
    -G    --rroute          包括 RECORD_ROUTE 选项并显示路由缓冲区
    --lsrr                  松散源路由和记录路由
    --ssrr                  严格源路由和记录路由
    -H    --ipproto         设置 IP 协议字段，仅在 RAW IP 模式中
ICMP
    -C    --icmptype        icmp 类型 (默认为回显请求)
    -K    --icmpcode        icmp 模式 (默认为 0)
    --force-icmp 发送所有 icmp 类型 (默认为仅发送支持的类型)
    --icmp-gw 设置 ICMP 重定向的网关地址 (默认为 0.0.0.0)
    --icmp-ts --icmp 的别名 --icmptype 13 (ICMP 时间戳)
    --icmp-addr --icmp 的别名 --icmptype 17 (ICMP 地址子网掩码)
    --icmp-help 显示其他 icmp 选项的帮助
UDP/TCP
    -s    --baseport 基源端口       (默认为随机值)
```

```
    -p    --destport  [+][+]<端口> 目标端口(默认为 0) ctrl+z inc/dec
    -k    --keep      保持源端口不变
    -w    --win       窗口大小 (默认为 64)
    -O    --tcpoff    设置虚假 tcp 数据偏移 (而不是 tcphdrlen / 4)
    -Q    --seqnum    仅显示 tcp 序列号
    -b    --badcksum        (尝试)发送带有错误 IP 校验和的数据包
                            许多系统将修复发送数据包的 IP 校验和，因此你
                            将获得错误 UDP/TCP 校验和。
    -M    --setseq          设置 TCP 序列号
    -L    --setack          设置 TCP ack
    -F    --fin             设置 FIN 标志
    -S    --syn             设置 SYN 标志
    -R    --rst             设置 RST 标志
    -P    --push            设置 PUSH 标志
    -A    --ack             设置 ACK 标志
    -U    --urg             设置 URG 标志
    -X    --xmas            设置 X 未用标志 (0x40)
    -Y    --ymas            设置 Y 未用标志 (0x80)
    --tcpexitcode 使用最后一个 tcp->th_flags 作为退出代码
    --tcp-mss 启用具有给定值的 TCP MSS 选项
    --tcp-timestamp 启用 TCP 时间戳选项以猜测 HZ/正常运行时间
常见
    -d    --data            数据大小 (默认值为 0)
    -E    --file            文件中的数据
    -e    --sign            添加“签名”
    -j    --dump            以十六进制转储数据包
    -J    --print           转储可打印的字符
    -B    --safe            启用“安全”协议
    -u    --end             告诉你 --file 何时到达 EOF 并防止倒带
    -T    --traceroute      路由跟踪模式 (暗指 --bind 和 --ttl 1)
    --tr-stop               当在路由跟踪模式中收到第一个非 ICMP 时退出
    --tr-keep-ttl           保持源 TTL 固定，对只监视一个跳步很有用
    --tr-no-rtt             不要在路由跟踪模式下计算/显示 RTT 信息

ARS 数据包描述 (全新、不稳定)
    --apd-send              发送使用 APD 描述的数据包 (请参阅 docs/APD.txt)
```

2. 使用 **hping3** 测试网络

在以下 hping3 示例中，我们使用域 demo-and-test.com。所有 URL 都可以使用 DNS 服务进行设置，并且你可以使用 URL 或 IP 地址执行测试。

2.1 测试已安装的软件和服务

第一步是确保在给定的虚拟机上正确安装项目所需的软件包。请注意，你可能需要调整 *bind9*、*apache2*、*ssh* 和 *vsftp* 的配置，以使其符合下面提出的要求。

1. 在服务器上，测试 Web 服务器，确保其正常工作：

   (a) 测试运行以下命令的 Apaches 服务器：

   ```
   $ service apache2 status
   ```

   (b) 通过编辑文件 */var/www/html/index.html* 建立一个演示网站，并添加一条语句，例如“欢迎进行演示和测试！”。有关如何设置 Web 服务的详细信息，请参考系统实验室 CS-SYS-00003（Linux 上的基本 Web 服务 (Apache) 设置）。

2. 在服务器上，测试 FTP 服务器，确保 vsFTP 服务器以被动模式运行，即数据通道将由客户端启动。因此，需要在网关上建立端口转发，以将 FTP 数据通道请求转发到服务器端上的指定数据接收端口。有关如何将 vsFTP 设置为被动模式的更多详细信息，请参考系统实验室 CS-SYS-00006（Linux 上的 FTP (vsFTP) 设置）。需要在 FTP 服务器上强制执行以下配置：
   - 启用匿名访问（即无需提供用户帐户和密码）。
   - 启用被动模式，客户端可以通过端口范围 [30000,30099] 启动数据通道。

   你可以在本地测试 FTP 服务。对于本项目，你可能不需要设置 FTP 身份验证和安全性。但是，你需要确保已启用被动模式。

3. 在服务器上，配置 DNS 服务器 (*bind9*) 以设置域名到 IP 地址的映射。DNS 服务需要配置的 URL 包括：

   ```
   1 demo-and-test.com
   2 www.demo-and-test.com % 这是 Apache 的默认 Web 链接
   3 ftp.demo-and-test.com
   4 ssh.demo-and-test.com
   ```

提示：URL 1-4 应解析为服务器的相同 IP 地址。请使用 *nslookup* 和 dig 测试你的配置。有关如何建立 DNS 服务的更多详细信息，请参考系统实验室 CS-SYS-00005（Linux 上的基本域名服务 (Bind9) 设置）。如果你想要跳过 DNS 设置（即不想从远程 DNS 服务器获取 URL 转换），则可以通过编辑 */etc/hosts* 添加本地域名解析程序，并在主机文件中添加以下映射：

```
192.168.0.3 demo-and-test.com
192.168.0.3 www.demo-and-test.com
192.168.0.3 ftp.demo-and-test.com
192.168.0.3 ssh.demo-and-test.com
```

在这里，IP 地址 192.168.0.3 是你的网关在网络客户端上的 IP 地址。请注意，将主机用于域名服务会使你的整个项目不完整，但允许你继续操作。你需要根据你的系统设置选择正确的地址。

4. 最后，在服务器上测试 SSH 服务器是否正在运行。

   ```
   $ sshd -v % 显示 ssh 服务器版本
   $ ssh ubuntu@localhost % 设置与服务器本身的 ssh 连接。SSH 服务器不允许 root 用户远程访问服务器，因此你应该使用用户帐户“ubuntu”来访问 ssh 服务。
   ```

2.2 使用 hping3 生成 TCP 测试流量

关于 hping3 要了解的一个最重要特性是，它的默认数据包是 TCP。这意味着，当路由器或防火墙等网络设备阻止 ICMP (ping) 时，我们仍可以使用 hping3 进行主机发现和侦察。

你可以设置 SYN 标志（这基本上与 nmap -sS 扫描相同）并检查端口 80 是否已打开 (-p 80)。

```
$ hping3 -S www.demo-and-test.com -p 80
```

请注意，在上面的屏幕截图中，返回的数据包设置了 SA 标志，这意味着端口已打开。

如果端口已关闭，端口将通过 RA 响应。

如果我们要扫描从 1 开始的所有端口，只需在端口 (p) 开关和要开始扫描的端口号（在此示例中为 1）后添加增量开关 (++)，如下所示：

```
$ hping3 -S www.demo-and-test.com -p ++1
```

这将扫描从 1 开始的每个端口，然后使端口 2 增加 1，再使端口 3 增加 1，依此类推。

2.3 使用 hping3 发送分片数据包

TCP 被设计成一种稳健的协议，即使在不利或困难的情况下也能继续通信。确保此稳健性的一个功能是，能够处理已分片或已分解成多个片段的数据包。TCP 将在这些数据包到达目标系统时对其进行重组。

通过使用诸如 hping3 的工具，可以使用 TCP 的这一功能对自身进行攻击，以将攻击分片成多个数据包，从而避开 IDS 和防火墙，然后在目标位置重组恶意软件。

虽然大多数 IDS 现在都尝试捕获分片攻击（在 Snort 中，有一个 frag3 预处理器，可尝试检测到分片），但旧的 IDS 做不到这一点。即使是较新的 IDS，也只能捕获设计用于检测的分片。hping3 的美妙之处在于，它允许我们设计 IDS 尚未发现的新攻击。

```
$ hping3 -f www.demo-and-test.com -p 80
```

2.4 使用 hping3 发送数据

hping3 还允许你将所需的任何数据放入传输的数据包中。通过使用 -E 开关实现此操作，以允许你表示要用于填充数据包的有效负荷的文件。

假设你有一个名为恶意软件的文件，其中包含一个漏洞。此外，你还担心 IDS 可能会检测到此恶意软件。你可以使用分片开关，并跨多个数据包加载恶意软件，该恶意软件将在避开 IDS 时由目标进行重组。

```
$ hping3 -f www.demo-and-test.com -p 80 -d 10 -E malware
```

在此命令中，-d 是数据有效负荷大小（在这里，我们将它指定为 10 字节），-E 指示 hping3 从以下文件中获取数据。然后，此命令一次性将大小为 10 字节的文件 malware 的内容发送到端口 80 上的目标。

2.5 使用 hping3 进行网络诊断

路由跟踪是一种工具，允许你通过操纵 ICMP 数据包的 TTL（生存时间），跟踪数据包在互联网上从客户端到目标的路由。

它可能是用于诊断网络问题的一种非常有用的工具，也可能被黑客用来查找网络上的设备和防火墙、路由器等的位置。因此，大多数网络管理员会阻止或删除 ICMP (ping)。

若要解决此问题，hping3 向我们提供与 ping 完全相同的功能，但要使用 TCP。以下命令使用 hping3 运行路由跟踪，并且 SYN 标志设置为 google.com。

```
$ hping3 -z -t 1 -S google.com -p 80
```

在此命令中，-z 将命令连接到键盘上的 ctrl z，以便每次按该键时，TTL 都增加 1，-t 将设置初始 TTL（在此案例中，我们使用 1）。

在此示例中，TTL 仍然是 1，hping3 告诉我们设备未知。然后，我们可以点击 ctrl z，使 TTL 增加 1，并找到我们和目标之间的每台设备。

此屏幕截图向我们显示我和 google.com 之间的两台设备。继续点击 ctrl z 将增加 TTL，查找每台设备，直到我到达 Google 的服务器。

2.6 使用 hping3 预测序列号

TCP 是一种面向连接的协议，它通过在目标位置对数据包进行重新排序来确保其稳健性，即使数据包在到达时没有秩序。为此，TCP 对数据包加序列号，以便它可以按到达时的顺序返回。

黑客使用此功能通过监视或猜测 TCP 序列号来执行 TCP 会话劫持攻击。为了防止猜测 TCP 序列号，操作系统制造商已调整其 TCP/IP 堆栈，以便序列号不再连续编号。相反，为了更难进行序列号猜测，操作系统使用一种算法来生成序列号。

hping3 可用于预测 TCP 序列号。我们可以让目标系统使用其序列号作出响应，然后从序列号中，我们可以破译操作系统正在使用的算法。我们可以通过以下命令实现此操作：

```
$ hping3 -Q -S google.com -p 80
```

在此命令中，-Q 显示序列号。

2.7 使用 hping3 测试服务器的正常运行时间

我们可以使用 hping3 来判断服务器已经运行了多长时间。这对黑客来说是非常有用的信息，因为通常必须重新启动服务器才能应用更新和补丁。通过了解系统已经运行了多长时间，我们可以预测应用了哪些补丁，以及系统容易受到哪些黑客攻击。

例如，如果我们发现一个系统在三年内都没有重新启动，我们可以非常确定，在这段时间内发布的任何安全补丁都没有被应用。这意味着，当时已知的所有漏洞在该系统上仍然存在。

Hping3 使用 TCP 时间戳数据包来预测系统已经运行了多长时间。

```
$ hping3 —TCP-timestamp -S google.com -p 80
```

3. 使用 hping3 进行侦察

hping3 可用于标识网络组件、服务和运行状态。在这里，你可以练习几个示例。在这些示例中，你可以根据你的系统设置更改扫描 IP 地址。

你可以使用 hping3 作为端口扫描仪。由于制作 TCP 数据包是 hping3 的默认行为，因此通过指定 TCP 标志、目标端口和目标 IP 地址，可以轻松地构建 TCP 数据包。

```
-F --fin      设置 FIN 标志
-S --syn      设置 SYN 标志
-R --rst      设置 RST 标志
-P –push      设置 PUSH 标志
-A --ack      设置 ACK 标志
-U –urg       设置 URG 标志
-X –xmas      设置 X 未用标志 (0x40)
-Y –ymas      设置 Y 未用标志 (0x80)
```

开放端口由 SA 返回数据包指示，封闭端口由 RA 数据包指示。请记住 TCP 3 次握手！这类似于一种众所周知的扫描方式，称为 SYN 扫描或隐蔽扫描。

3.1 将 ACK 数据包发送到目标

```
$ hping3 -A -c 5 192.168.0.3
```

你可以使用 -c 选项决定要发送多少数据包，在此示例中，我将该计数选项设置为 5。响应如下：

```
HPING 192.168.0.3 (eth0 192.168.0.3):A 集合，40 个标头 + 0 数据字节
len=46 ip=192.168.0.3 ttl=128 id=30010 sport=0 flags=R seq=0 win=32767 rtt=7.9 ms
len=46 ip=192.168.0.3 ttl=128 id=30011 sport=0 flags=R seq=1 win=32767 rtt=7.0 ms
len=46 ip=192.168.0.3 ttl=128 id=30012 sport=0 flags=R seq=2 win=32767 rtt=7.6 ms
len=46 ip=192.168.0.3 ttl=128 id=30013 sport=0 flags=R seq=3 win=32767 rtt=5.1 ms
len=46 ip=192.168.0.3 ttl=128 id=30014 sport=0 flags=R seq=4 win=32767 rtt=4.0 ms

--- 192.168.0.3 hping 统计信息 ---
已传输 5 个数据包，已接收 5 个数据包
```

3.2 将 SYN 数据包发送到目标

你可以创建 SYN 数据包，并使用扫描模式扫描目标上的端口 1-1000。

```
$ hping3 -S -8 1-1000 192.168.0.3
```

```
扫描 192.168.0.3 (192.168.0.3)，端口 1-1000
要扫描 1000 个端口，使用 -V 查看所有回复
端口  服务名称  标志  ttl  id  win  len

  53 domain: .S..A...128 55677 64240 46
  88 kerberos: .S..A...128 55933 64240 46
```

```
    135 epmap: .S..A...128 56189 64240 46
    139 netbios-ssn: .S..A...128 56445 64240 46
    389 ldap: .S..A...128 56701 64240 46
    445 microsoft-d: .S..A...128 56957 64240 46
    464 kpasswd : .S..A...128 57213 64240 46
    593 : .S..A...128 52863 64240 46
    636 ldaps : .S..A...128 53375 64240 46
已收到所有回复。已完成。
没有响应端口 : (199 smux) (202 at-nbp) (203 ) (204 at-echo) (299 ) (300 ) (301) (306 )
    (307 ) (308 ) (309 ) (312 ) (313 ) (407 ) (500 isakmp) (514 shell) (723) (729 ) (743 )
    (761 ) (763 ) (764 ) (766 ) (767 ) (768 ) (769 ) (772 ) (782 ) (783 spamd) (784 ) (790 )
    (791 ) (793 ) (794 ) (798 ) (799 ) (802 ) (803 ) (804 ) (805 ) (808 omirr) (809 ) (810 )
    (811 ) (812 ) (813 ) (817 ) (818 ) (819 ) (820 ) (821 ) (822 ) (823 ) (824 ) (825 ) (827 )
    (828 ) (829 ) (831 ) (832 ) (833 ) (834) (836 ) (837 ) (838 ) (839 ) (840 ) (841 ) (842 )
    (843 ) (844 ) (845 ) (846 ) (847 ) (848 ) (849 ) (854 ) (855 ) (858 ) (878 ) (879 ) (880 )
    (881 ) (911 ) (912) (913 )
root@ubuntu:~#
```

3.3 将 UDP 数据包发送到目标

你可以发送 UDP 扫描模式以将端口 80 上的 UDP 请求发送到目标，如果 UDP 端口已打开，你将获得一个响应，这在目标已阻止 ICMP ping 时非常有用。

```
$ hping3 -2 192.168.0.3 -c 2 -p 80
```

3.4 将 ping 数据包发送到目标

你可以创建 ping 数据包并使用 ICMP 模式。

```
$ hping3 -1 -c 4 192.168.0.3
```

```
HPING 192.168.0.3 (eth0 192.168.0.3): icmp 模式设置，28 个标头 + 0 数据字节
len=46 ip=192.168.0.3 ttl=128 id=34163 icmp_seq=0 rtt=8.1 ms
len=46 ip=192.168.0.3 ttl=128 id=34164 icmp_seq=1 rtt=5.9 ms
len=46 ip=192.168.0.3 ttl=128 id=34167 icmp_seq=2 rtt=4.0 ms
len=46 ip=192.168.0.3 ttl=128 id=34168 icmp_seq=3 rtt=3.0 ms

--- 192.168.0.3 hping 统计信息 ---
已传输 4 个数据包，已接收 4 个数据包，数据包损失 0%
round-trip min/avg/max = 3.0/5.2/8.1 ms
root@ubuntu:~#
```

3.5 将 traceroute/ping 数据包发送到目标

你可以使用 hping3 通过使用 ICM 模式将路由跟踪 prob 发送到目标并显示详细信息。

```
$ hping3 —traceroute -V -1 192.168.0.3
```

```
使用 eth0，地址:172.168.200.110, MTU:1500
HPING google.com (eth0 216.58.211.142): icmp 模式设置，28 个标头 + 0 数据字节
```

```
hop=1 TTL 0 during transit from ip=172.168.200.2 name=_gateway
hop=1 hoprtt=3.9 ms
hop=2 TTL 0 during transit from ip=192.168.10.1 name=UNKNOWN
hop=2 hoprtt=2.0 ms
hop=3 TTL 0 during transit from ip=10.33.221.74 name=UNKNOWN
hop=3 hoprtt=8.9 ms
hop=4 TTL 0 during transit from ip=88.129.174.18 name=gbg1.dr8.a3network.se
hop=4 hoprtt=8.9 ms
hop=5 TTL 0 during transit from ip=88.129.128.62 name=gbg1.a7network.se
hop=5 hoprtt=8.0 ms
hop=6 TTL 0 during transit from ip=85.8.9.16 name=gbg1.cr1.a3network.se
hop=6 hoprtt=6.9 ms
hop=7 TTL 0 during transit from ip=85.8.10.20 name=sto2.cr1.a3network.se
```

你还可以使用 ping3 发送路由跟踪 prob，以确定端口 443 是否已打开，并设置从源端口 8080 生成本地流量。

```
$ hping3 —traceroute -V -S -p 443 -s 8080 google.com
```

```
使用 eth0，地址:172.168.200.110, MTU:1500
HPING google.com (eth0 216.58.211.142):S 集合，40 个标头 + 0 数据字节
hop=1 TTL 0 during transit from ip=172.168.200.2 name=_gateway
hop=1 hoprtt=8.9 ms
len=46 ip=216.58.211.142 ttl=128 id=34374 tos=0 iplen=44
sport=443 flags=SA seq=8 win=64240 rtt=13.8 ms
seq=905581660 ack=1390210946 sum=3cce urp=0

len=46 ip=216.58.211.142 ttl=128 id=34376 tos=0 iplen=44
sport=443 flags=SA seq=9 win=64240 rtt=13.9 ms
seq=277232268 ack=486133387 sum=5a24 urp=0

len=46 ip=216.58.211.142 ttl=128 id=34377 tos=0 iplen=44
sport=443 flags=SA seq=10 win=64240 rtt=13.0 ms
seq=1939483389 ack=2029365982 sum=8498 urp=0

len=46 ip=216.58.211.142 ttl=128 id=34378 tos=0 iplen=44
sport=443 flags=SA seq=11 win=64240 rtt=12.9 ms
seq=90127368 ack=1561834414 sum=c208 urp=0
```

你可以在路由跟踪中使用 TTL，以检查负载平衡设备的 IP 地址。

```
$ hping3 -S 192.168.100.100 -p 80 -T --ttl 13 --tr-keep-ttl -n
```

对子网执行 ping 操作，不按顺序扫描，而是随机扫描。使用 --rand-dest 和 interface -I eth0 运算符。

```
hping3 -1 192.168.100.x —rand-dest -I eth0
```

发送 ICMP 数据包以从目标请求时间戳，如果目标已阻止 ICMP 响应，它将不会响应 ICMP 数据包，但是它可能会允许响应时间戳请求。

```
hping3 -1 192.168.0.3 —icmp-ts -c 3
```

3.6 其他类型的端口扫描

- 我们将尝试的第一种类型是 FIN 扫描。在 TCP 连接中，FIN 标志用于启动连接关闭例程。如果我们没有收到回复，这意味着端口已打开。通常，防火墙发送回 RST+ACK 数据包，以指示端口已关闭。

```
$ hping3 -c 1 -V -p 80 -s 5050 -F demo-and-test.com
```

- Ack 扫描：此扫描可用于查看主机是否处于活动状态（例如，当 Ping 被阻止时）。如果端口已打开，这应该发送回 RST 响应。

```
$ hping3 -c 1 -V -p 80 -s 5050 -A demo-and-test.com
```

- Xmas 扫描：此扫描将序列号设置为零，并在数据包中设置 URG + PSH + FIN 标志。如果目标设备的 TCP 端口已关闭，目标设备将发送 TCP RST 数据包作为回复。如果目标设备的 TCP 端口已打开，目标将放弃 TCP Xmas 扫描，不发送回复。

```
$ hping3 -c 1 -V -p 80 -s 5050 -M 0 -UPF demo-and-test.com
```

- Null 扫描：此扫描将序列号设置为零，并且在数据包中未设置任何标志。如果目标设备的 TCP 端口已关闭，目标设备将发送 TCP RST 数据包作为回复。如果目标设备的 TCP 端口已打开，目标将放弃 TCP NULL 扫描，不发送回复。

```
$ hping3 -c 1 -V -p 80 -s 5050 -Y demo-and-test.com
```

请注意，当使用 -Y 选项时，hping3 将发送一个设置了 ECN/CWR 位的 TCP 数据包（TCP- 转储标志 [W]）。因此，你可以通过使用 nmap 来使用更干净的 Null 扫描：

```
$ nmap -sN demo-and-test.com
```

4. 使用 hping3 部署 DOS 攻击

4.1 什么是 DOS 攻击？

什么是 *DOS？*

拒绝服务 (DOS) 攻击是一种非常简单的技术，用于拒绝对服务的访问（这就是它被称为“拒绝服务”攻击的原因）。此攻击由使用超大数据包或大量数据包使目标过载组成。虽然此攻击很容易执行，但它不会损害目标的信息或隐私，它不是一种渗透性攻击，只是旨在阻止对目标的访问。通过发送目标无法处理的大量数据包，攻击者阻止服务器为合法用户提供服务。

DOS 攻击从单个设备执行，因此通过阻止攻击者 IP 可以轻松阻止它们，然而攻击者可以更改甚至伪造（克隆）目标 IP 地址，但与 DDOS 攻击相反，防火墙处理此类攻击并不困难。

为什么使用 *hping3*？

工具 hping3 允许你发送操作的数据包。此工具允许你控制数据包的大小、数量和分片，以便使目标过载，以及绕过或攻击防火墙。Hping3 可以用于安全或功能测试，使用它，你可以测试防火墙有效性，以及服务器是否可以处理大量数据包。在下面，你可以找到有关如何使用 hping3 进行安全测试的说明。

使用 hping3，一个简单的 DOS 攻击将为：

```
$ sudo hping3 -S --flood -V -p 80 www.demo-and-test.com
```

在此命令中，--flood 让 hping3 随意射击，将忽略回复（这就是不显示回复的原因）并将尽快发送数据包，-V 选项是指详细信息。接下来，我们将枚举几个攻击者可以使用 hping3 部署的 DoS 攻击。请注意，我们将尝试从一个节点模拟攻击者的行为。当我们使用伪造的源 IP 地址部署多个攻击源时，我们可以模拟分布式 DoS (DDoS) 攻击。请始终记住，你的目标是研究攻击并设计一个如何防御攻击的解决方案。因此，在小范围内以可控的方式部署攻击，不要让它“失控”，也不要让它“无人看管”。一旦完成测试，请停止你的攻击行为。

4.2 ICMP 洪水

用于检查特定系统提供的服务的常用命令是“ping”。使用 ping 命令，计算机通过“ICMP ping 请求”向另一个台计算机发送信号。

ICMP 洪水

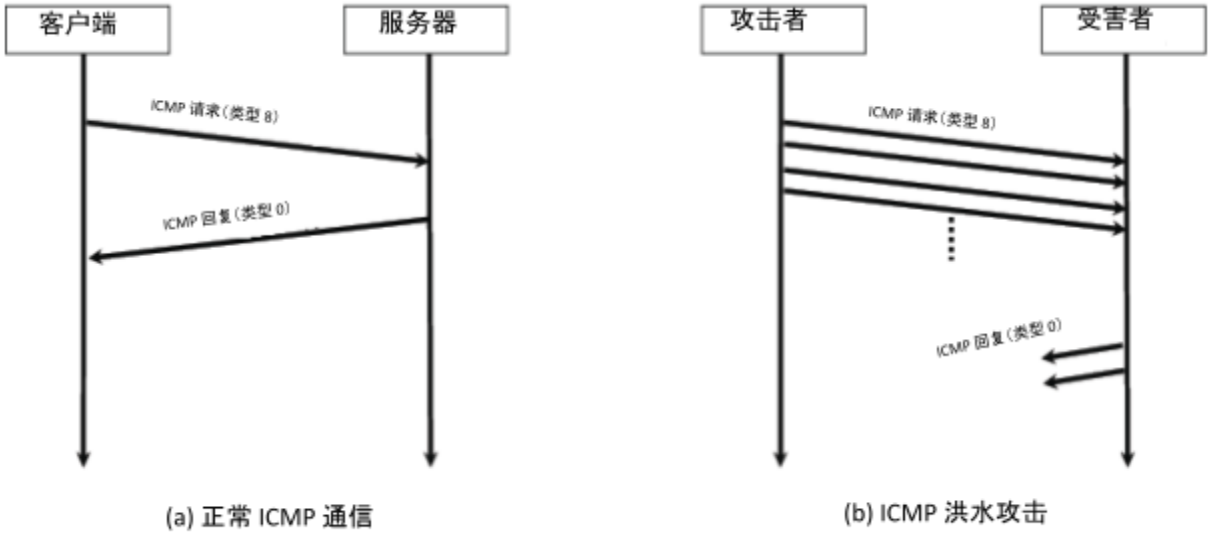


图 **CS-CNS-100031**
ICMP 洪水攻击

这是一种拒绝服务攻击，通过大量伪造的 ping 消息淹没目标系统。然后，使处理 ICMP 消息和响应伪造 IP 的受害者不堪重负。

```
$ hping3 -1 --flood -a SPOOFED_IP VICTIM_IP
```

4.3 Smurf 洪水攻击

Smurf 一种拒绝服务攻击，通过伪造的广播 ping 消息淹没目标系统。网络上的所有节点都将回复受害者 IP 地址。

如图 CS-CNS-10003.2 中所示，使用受害者的返回 IP 地址将“ICMP 回显请求”发送到广播地址。中间网络中的所有计算机都通过“ICMP 回显回复”响应受害者。因此，受害者会收到成百上千个 ping 回复。

```
$ hping3 -1 --flood -a VICTIM_IP BROADCAST_ADDRESS
```

Smurf 洪水

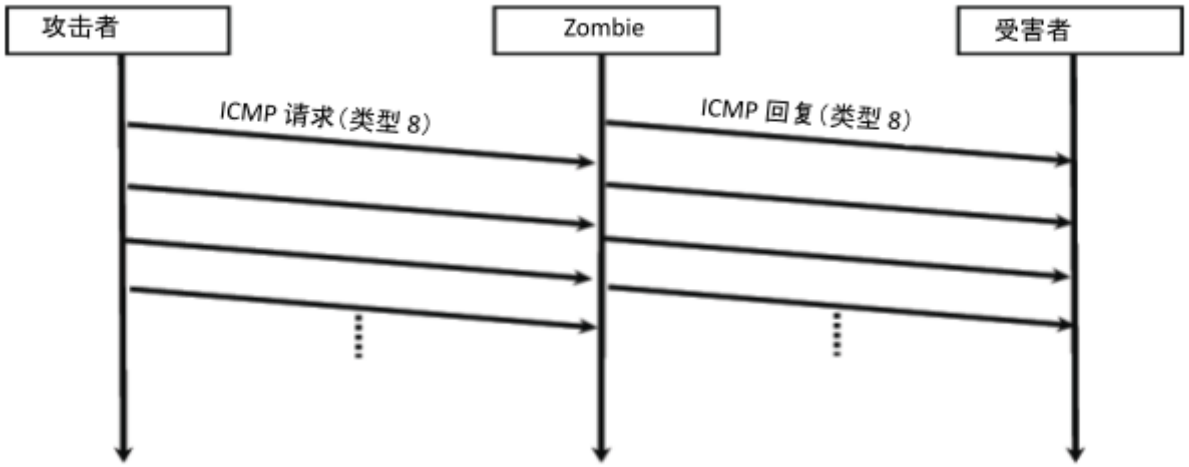


图 CS-CNS-10003.2

ICMP 洪水攻击。

使用受害者的返回 IP 地址将“ICMP 回显请求”发送到广播地址。中间网络中的所有计算机都通过“ICMP 回显回复”响应受害者。因此，受害者会收到成百上千个 ping 回复。

4.4 Fraggle 洪水攻击

Fraggle 洪水攻击是 UDP 版本的 Smurf 洪水攻击。

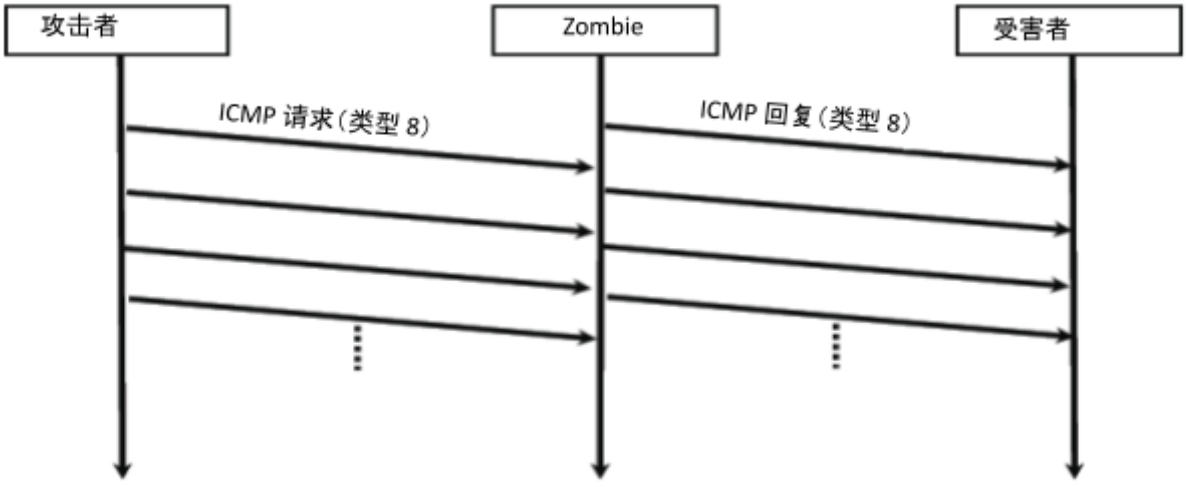


图 CS-CNS-10003.3

Fraggle 洪水攻击。

如图 CS-CNS-10003.2 中所示，Fraggle 攻击者向 IP 广播地址发送大量 UDP 回显流量，所有流量都有一个虚假的源地址作为受害者的 IP 地址。Zombies 向受害者发送 UDP 重放数据包。受害者收到大量 UDP 重放数据包。这是对 Smurf 攻击的简单重写。

```
$ hping3 -2 --flood -a VICTIM_IP BROADCAST_ADDRESS
```

4.5 UDP 洪水攻击

在 UDP 洪水攻击中，如图 CS-CNS-10003.4 中所示，攻击者向受害者系统上的一个随机端口发送 UDP 数据包。当受害者系统收到 UDP 数据包时，它将确定哪个应用程序正在目标端口上等待。当它意识到没有应用程序正在端口上等待时，它将生成一个对于伪造的源 IP 地址目的地不可达的 ICMP 数据包。

如图 CS-CNS-10003.2 中所示，Fraggle 攻击者向 IP 广播地址发送大量 UDP 回显流量，所有流量都有一个虚假的源地址作为受害者的 IP 地址。Zombies 向受害者发送 UDP 重放数据包。受害者收到大量 UDP 重放数据包。这是对 Smurf 攻击的简单重写。

UDP 洪水

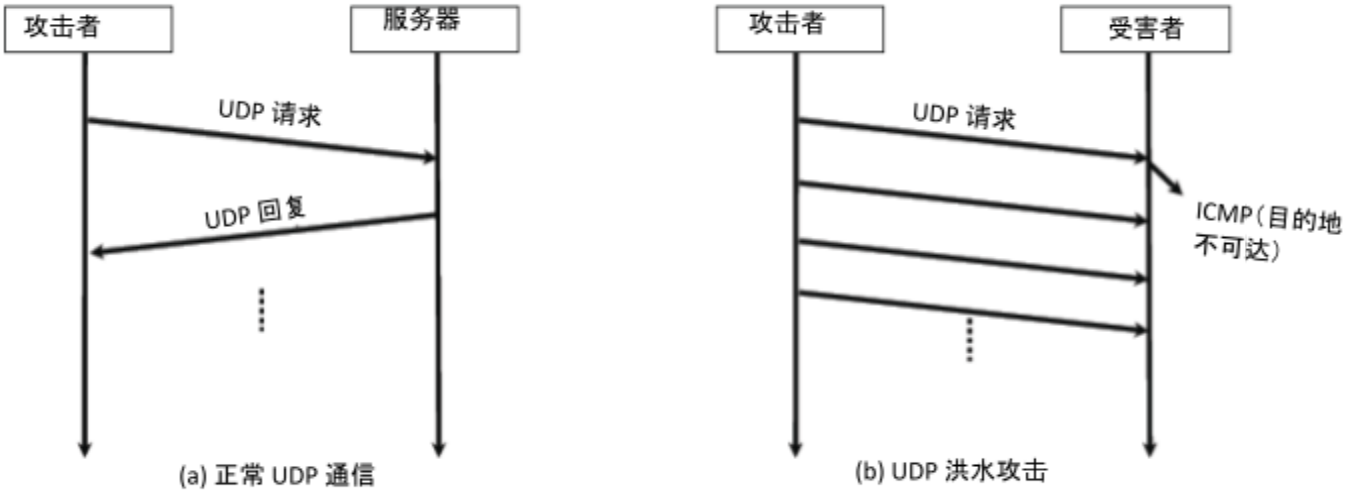


图 **CS-CNS 100034**
UDP 洪水攻击

攻击者向受害者系统上的一个随机端口发送 UDP 数据包。当受害者系统收到 UDP 数据包时，它将确定哪个应用程序正在目标端口上等待。当它意识到没有应用程序正在端口上等待时，它将生成一个对于伪造的源 IP 地址目的地不可达的 ICMP 数据包。

```
$ hping3 -2 --flood VICTIM_IP
```

4.6 Land 攻击

Land 攻击是指，当你发送一个数据包时，源 IP 地址和目标 IP 地址完全相同，如图 CS-CNS-10003.5 中所示。这是 IP 欺骗的一个示例。网络基础设施应该为

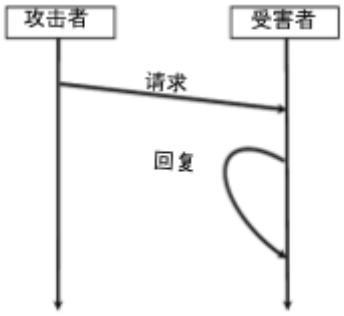


图 **CS-CNS-10003.5**

Land 洪水攻击。

设置为拒绝伪造的数据包。使用最佳安全实践，你应该永远不会让任何具有你的 IP 空间源地址的流量进入你的环境。若要模拟 Land 攻击，假设你想要测试一个 IP 地址为 192.168.0.3 且端口 80 已打开的系统。若要执行此操作，你将发出以下命令：

```
$ sudo hping3 -S 192.168.0.3 -a 192.168.0.3 -k -s 80 -p 80 --flood
```

请注意，最新的 Linux 操作系统通常附带地址欺骗攻击保护。如果此消息通过网关节点（例如 Linux），它可能会停止转发伪造的数据包。若要在网关上禁用地址欺骗保护，你可以执行以下操作：

1. 在 */etc/sysctl.conf* 中设置 *net.ipv4.conf.al l.rp_filter = 0*。
2. 重新启动网关。

然后，网关将能够转发伪造的数据包。

4.7 基于 TCP 的洪水攻击

4.7.1 TCP 洪水

如图 CS-CNS-10003.6 中所示，攻击者使用多个具有伪造 IP 的 SYN 请求淹没受害者。接收的 TCP 请求可以快速填满受害者计算机的最大 SYN 请求限制。下面的示例描述了针对 demo-and-test.com 的 SYN 攻击的一些变体：

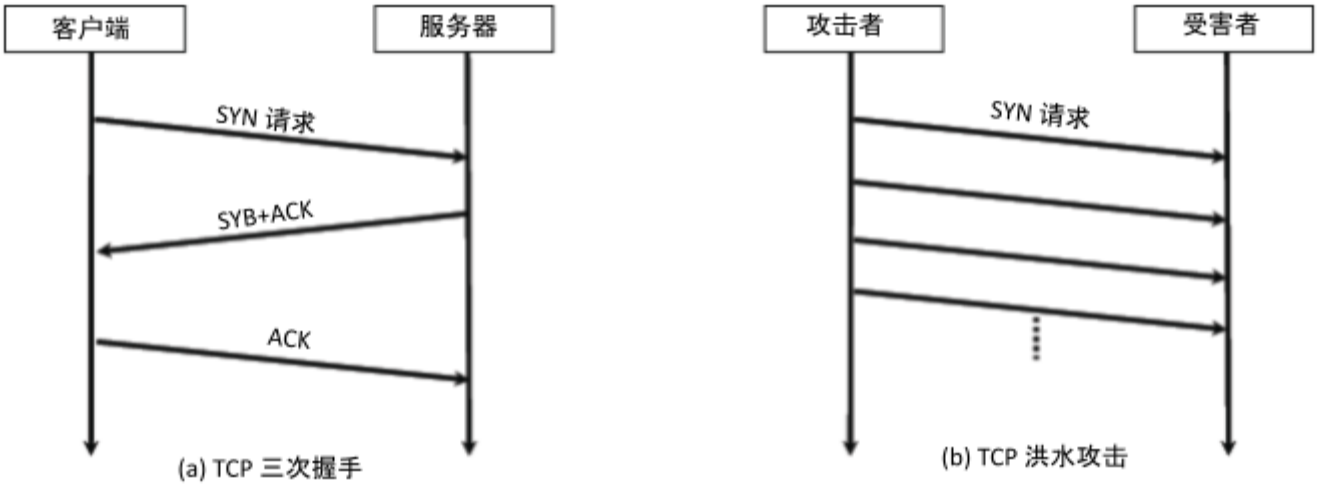


图 CS-CNS-10003.6
TCP 洪水攻击。

```
$ sudo hping3 demo-and-test.com -q -n -d 120 -S -p 80 —flood —rand-source
```

在此命令中，“-q”设置简短输出，“-n”显示目标 IP 而不是主机，“-d 120”设置数据包大小，最后，“--rand-source”通过随机选择源 IP 地址隐藏 IP 地址。

以下命令伪造给定的 IP 地址，而不是使用随机 IP 地址：192.168.0.100：

```
$ sudo hping3 demo-and-test.com -q -n -d 120 -S -p 80 —flood -a 192.168.0.100
```

以下示例显示如何针对端口 80 部署 SYN 洪水：

```
$ sudo hping3 —rand-source demo-and-test.com -S -q -p 80 —flood
```

通过 hping3，你还可以使用一个虚假的 IP（即 IP 洪水攻击）来攻击目标，为了绕过防火墙，你甚至可以克隆目标 IP 本身或你可能知道的任何允许的地址。语法将为：

```
$ sudo hping3 -a <FAKE IP> <target> -S -q -p 80 —faster -c2
```

在此实际示例中，攻击似乎是：

```
$ sudo hping3 -a 10.0.0.4 demo-and-test.com -S -q -p 80 —faster -c2
```

5. 相关信息和资源

hping 教程：https://www.radarhack.com/tutorial/hping2.pdf
hping 文档：http://www.hping.org/documentation.php