第21章　安全性问题

如果你需要下列问题的一个快速解决方案	请查阅节号
关闭简单的薄弱环节	21.2.1
列出成功登录的记录清单	21.2.2
列出不成功登录的记录清单	21.2.3
查找安全漏洞	21.2.4
密切注意系统安全性方面的进展	21.2.5
确定是否需要一堵防火墙	21.2.6
强化对网络驱动器访问的控制	21.2.7
过滤向外发送的数据包	21.2.8
安装ipchains	21.2.9
在重启动/关机之前保存数据包过滤规则	21.2.10
开机引导后恢复数据包过滤规则	21.2.11
安装SOCKS	21.2.12
安装代理服务器程序	21.2.13
配置SOCKS	21.2.14
设置Linux机器通过代理防火墙	21.2.15

21.1　概述

任何连接到因特网上的计算机或者 LAN都必须采用一些安全性措施来防备侵入者。其中某些措施是很容易添加到系统中的。这样做需要了解正在使用的系统和它的基本薄弱环节，一旦发现了漏洞，就应该把它们堵上。另外一些措施比较复杂一点，需要额外的软件或硬件才能实现。重要的是要知道它们是什么，评价系统安全性要求，然后正确地实现它们。

防火墙选项

在关于网络安全性的讨论中，防火墙是经常被强调的重点，但它们并不是一个万能的工具。它们的基本功能是过滤并阻挡网络或者网络的某个部分与外部世界 (也就是因特网)之间的数据传送。在其工作过程中，它们将接管在此之前从网络内部存取因特网和从因特网存取该内部网络的路由设置。请参考 21.2.6节中的项目清单，它会帮助你决定增加一个防火墙是否就是系统安全性所要求的最佳解决方案。

1. IP过滤防火墙

要想实现IP过滤防火墙需要使用 ipchains软件包 —— 2.1及以上的内核版本都能够支持这个软件包。这个软件包管理着对 Linux操作系统内核本身极为重要的那些 IP帐户和防火墙功能。用户使用的内核必须已经在编译时激活 ipchains功能；如果打算使用这个功能，却又没有内核级的ipchains支持的话，请阅读第10章，那里介绍了怎样编译内核。

如果想掌握ipchains运行在哪一个层次，就必须在数据包级别上精通 TCP/IP网络的数据传输情况。数据包是一段段的数据，其中同时包括了用来把它们发送到各自的目的地所必须的信息。你可以把数据包想象成一个邮包：数据包本身就是邮包中的数据，而信封上则是所有用来把这些信息发送到正确的机器和正确的程序中去的书信抬头，它同时还包含着回信地址

等方面的信息。

ipchains是一个数据包过滤器。数据包过滤器是一个用来检查数据包的信封内容的程序；同时，根据用户设定的一系列规则（请参阅 21.2.8节中关于如何设定这类规则的内容），还可以决定是否允许某些数据包通过和把它们发送到什么地方。

2. 代理防火墙

要想实现代理防火墙功能，需要下载并安装这样的软件 —— 这类软件能够控制用户从某个网络的外部可以使用和不可以使用什么样的网络服务功能。代理防火墙不允许有任何连接接入到它后面的机器上，不存在任何数据包过滤的效果，从接入连接的角度考虑，它就是一堵密不透风的砖墙。

最常见的用来设置代理防火墙的软件包被称为 SOCKS（请参阅21.2.12节中的内容来获得并安装这个程序，它没有被包括在本书讨论的两种发行版本中）。

21.2 快速解决方案

21.2.1 关闭简单的薄弱环节

不论你安装的是 Red Hat还是Caldera版本的Linux操作系统，其中都有一些能够禁用的薄弱环节，这样做会帮助你提高系统的安全性：

1) 在/etc/inetd.conf文件中禁用任何你不打算使用的服务功能，方法是把它们改为注释语句（在那些语句开头加上一个 #符号）。任何一种服务都会为试图进入系统的那些人多打开一条通路，因此应该只使用你确实需要的服务。

2) 编辑/etc/issue.net文件，删除其中关于在你的机器上运行的特定硬件和 Linux发行版本的介绍信息。这些信息会在诸如Telnet之类的远程登录任务操作过程中显示在登录端的屏幕上。如果他们对屏幕上显示的安装方式熟悉的话，任何暴露了你机器上这些特殊信息的东西都会使那些试图闯入的人们了解应该去攻击哪些薄弱环节。

3) 把某些特殊的用户们分配到 console用户组中，这样当这些人实际坐在服务器计算机前面时确实可以执行命令，但同时要禁止任何其他人调用这些命令。举例来说，如果需要强调安全性，那就应该只允许 console用户组的成员可以挂装磁盘。

4) 查看/etc/securetty文件，确定其中列出的设备都是真实存在的物理 ttys（比如tty1到tty8）端口。这个文件限制了人们能够以根用户身份登录进入系统的位置。允许任何远端用户以根用户身份登录进入系统是极其危险的，因为这样做就为潜在的侵入者缩短了侵入超级用户帐户的过程。

5) 一定要保证在FTP服务器上唯一允许匿名用户进行写入操作的部分是 /incoming目录。

21.2.2 列出成功登录的记录清单

如果想查看都有哪些人最近成功地登录进入了系统，可以使用 last命令。如果想列出比缺省数目更多的记录，可以使用格式“ last -n number ”告诉last命令需要显示多少登录记录。

窍门 如果这个命令执行失败，说明登录操作还没有被记录下来。为了确保它们能够被记录下来，请以根用户身份执行“ touch /sar/log/wtmp ”命令来建立日志记录文件。

21.2.3 列出不成功登录的记录清单

如果想查看都有哪些人最近没有成功地登录进入系统，可以使用 lastb命令。和last命令相类似，如果你想列出比缺省数目更多的不成功尝试记录的话，可以使用“ lastb -n number ”格式。

窍门 如果这个命令执行失败，说明不成功登录操作还没有被记录下来。为了确保它们能够被记录下来，请以根用户身份执行“touch /var/log/btmp ”命令来建立日志记录文件。

21.2.4 查找安全漏洞

如果安全出现了漏洞，或者你认为已经出现了漏洞的话，就可以抓住这个机会从系统登录记录或者其他地方查找蛛丝马迹。比较好的办法是：

- 系统会记录下每一次成功的登录操作和不成功的登录尝试操作。请阅读 21.2.2节和21.2.3节中介绍的如何获得这些登录记录清单的内容。特别是那个“ 列出不成功的尝试登录记录清单 ”，它能够让你很快地查找出是否有人在千方百计地获取根用户权限或者猜试某个用户的口令字。
- 如果你使用的是shadow隐藏口令字软件，请检查 /etc/passwd文件，看看其中是否禁用了shadow隐藏口令字功能 —— 如果是这样，口令字将会被保存在 /etc/passwd文件里。修改这个设置必须拥有根用户的权限，所以如果情况真的如上面所说的这样，你就应该知道在这台机器上黑客已经获得了根用户的操作权限。
- 检查是否有不熟悉的用户帐户拥有特殊的优先权。
- 一个拥有高级工具和知识的黑客能够替换某些重要的系统功能。用一台独立的计算机或者一台你相信没有遭到攻击的机器把诸如 ls之类的命令备份到一张软盘上去 —— 最好的办法是直接从发行版本的 CD-ROM光盘上制作备份，然后检查程序 ls、find、ps和所有网络守护进程的系统版本中的日期与它们生成时的日期数据是否一致。
- 如果你的网络服务日志或者任何系统日志中有时间缺口，而你知道你并没有在时间缺口期间关闭过系统日志记录功能的话，就应该怀疑安全性受到破坏。

窍门 如果安全性对你来说极为重要，请考虑编写一个程序来监视日志记录文件并随时报告可疑的活动。

21.2.5 密切注意系统安全性方面的进展

你应该经常到下列的一些地方去查看是否有提高安全性方面的建议和是否有那些修补最新发现的系统漏洞的链接：

- Red Hat和Caldera两种发行版本都有通告邮件清单，安全性方面的补丁程序上载到他们站点时会立刻通知大家。如果想加入 Red Hat公司的清单，请向地址 redhat-announce-list-request@redhat.com发一封电子邮件，在标题行写上单词 subscribe（订阅），邮件内容部分空着就可以了。如果想加入 Caldera公司的清单，请向地址 majordomo@lists.calderasystems.com 发一封电子邮件，标题栏空着，在邮件正文部分添上 Subscribe Announce（订阅通告）字样就可以了。

- 如果安装的是 Red Hat的发行版本，请定期（每周一次甚至更频繁）浏览他们的站点www.redhat.com/errata，看看有没有安全性方面的升级和补丁程序。
- 如果安装的是Caldera发行版本,请到地址www.calderasystem.com/support/resources 上查找升级信息。
- 因特网上一个比较安全地提供操作系统安全性方面信息的中枢是 Computer Emergency Response Team（计算机紧急情况快速反应组，简称 CERT）站点。它们的Web主页是：www.cert.com。

21.2.6 确定是否需要一堵防火墙

防火墙对某些站点来说是非常优秀的防护措施，但是对另外一些来说可能就是一个长期性的问题。用户在安装一堵防火墙之前，请先考虑防火墙是否适用于用户的 LAN，是不是还有其他更好的措施来提高系统的安全性。

1. 考虑安装一堵防火墙

如果处于下列几种情况，用户至少应该考虑安装一道防火墙：

- 用户有一些非常重要的商业数据，但是又不能把它们保存在独立于用户的网络之外的某台计算机上。如果这些数据被偷会发生什么状况呢？如果被侵入者删除或者修改了又该怎么办呢？虽然好的备份技巧可以保存某一时期的数据不至于受到损失，但有的时候那怕很短时间的数据丢失都是灾难性的。
- 用户的Web站点频频出现在大众传播媒介中，也许是因为其拥有者的行为，也许是因为某些用户。如果某个用户不合时宜并且触怒了大众，或者这个站点本身的基调就是不合时宜的，就会有人试图闯入它来“惩罚冒犯者”。如果这个站点仅仅是因为有一个很知名的名称而频频曝光的话，可能就会有人试图闯入它来显示自己的能力。
- 用户需要分析出到底有多少台机器是真正连在自己的网络之中，并且要考虑为哪一台机器分配哪一个地址可以让这台机器看上去对潜在的侵入者没有什么吸引力。

相关解决方案	请查阅节号
选择一种备份策略	20.2.3

2. 提高安全性，但是不需要使用防火墙

解决用户安全性需求的答案可能根本就不是一堵防火墙：

- 用户的整个网络是不是非得连接到因特网上不可呢？也许用户的办公室或者 LAN只需要有一两台上网的机器就足够了，而这几台机器也不是非得连到 LAN的其他部分中去。
- 那个保存着最关键数据的机器能不能移到 LAN以外呢？也许这么做就足够了。
- 因特网连接是永久性的还是临时的？一个在必要时才建立的连接通常并不需要一堵防火墙；除非它上网的时间持续到接近永久性连接的地步，这样来自外界的攻击就有可能了。如果因特网连接每次使用的都是不相同的 IP地址（动态IP地址分配）的话，上面说的情况就极可能会发生。

3. 防火墙防不胜防

对某些类型的网络安装和使用需求来说，有时候维护防火墙太麻烦，以至于没有什么使用价值。这些类型的情况有下列几个例子：

- 用户站点上的用户需要能够使用因特网上许多不同类型的服务。如果是这样的话，那么防火墙最终会被“ 凿 ”得千疮百孔，因为用户必须允许数据有进有出，这样就使防火墙实际上起不到什么真正的作用，而用户还将花费大量的时间去调试防火墙的规则集合。
- 有一些软件不能透过防火墙运行，但是可以通过特殊的编程方法实现这一点。如果用户的LAN中需要使用的某个软件不能透过防火墙运行的话，就只好选择另外一种安全性解决方案了。

21.2.7 强化对网络驱动器访问的控制

在安装诸如FTP、Samba或者NFS等服务的时候，用户就会增加潜在的危险，因为这些服务都允许人们远程存取那些网络中的驱动器。在安全性方面有一条原则：如果你提供了一种网络服务，就肯定会有人想找一条进来的通路。请按照下面的步骤强化对网络驱动器存取的控制。

1) 打开想强化的某项服务的存取控制文件。

2) 检查打算通过这项服务提供些什么东西？是只想接纳很少的一部分人呢，还是某个项目小组，亦或是整个办公室呢？

3) 准确地界定确实想把某个特殊的服务提供给某些特定范围的人们，逐步细化到某个 IP 地址范围或者某个域名上的特定主机。

4) 为这些能够使用这项服务的用户确定精细的使用规则，同时对所有其他人屏蔽这个服务。

21.2.8 过滤向外发送的数据包

使用ipchains安装防火墙就需要配置 ipchains的数据包过滤功能。请按照下面的方法配置这个功能：

1) 以root用户身份登录进入系统。

2) 输入“ rpm -q ipchains ”命令，检查系统上是否已经安装了 ipchains。如果这个软件包没有被安装的话，请阅读21.2.9节，然后再回到下面的第3个步骤。

3) 输入“ ipchains -L ”命令，查看当前已经存在的链。如果你还没有对这些选项进行过配置的话，应该会看到下面这样的几行：

```
> chain input ( policy ACCEPT ) ;
> chain forward ( policy ACCEPT ) ;
> chain output ( policy ACCEPT ) ;
>
```

4) 选择打算配置的链。一个链就是一系列加在一组数据包类型上的规则。

- input chain（输入链）指的是进入到防火墙机器中的数据包。
- forward chain（转发链）指的是进入到防火墙机器中而现在又需要发送到网络中的另外一台机器上的数据包。
- output chain（输出链）指的是要向外发送的数据包。

5) 输入“ ipchains -L chain ”命令，列出打算编辑的链中当前已经存在的规则。在缺省的情况下，所有的链中的规则都是空白的。

6) 输入“ ipchain -A chain ”，告诉ipchains程序需要针对哪个链建立一个新的数据包过滤

规则。举例来说，用户可能想对输入链建立些新规则，那么就输入“ ipchains -A input ”。现在还不能按回车键，还有事情要做。

7) 现在需要键入规则本身的内容。我们假定用户想做的设置是：除了从某一台特定的远程工作站点之外，人们没有办法远程登录连接到防火墙后面的任何机器。在你正在建立的那个规则的格式中使用-s(source，源) 标志设置地址（或地址范围）：

- 单个完整的IP地址，比如：192.168.152.24。
- 整个一类的IP地址范围，比如：192.168.152.0/255.255.255.0 —— 它表示从192.168.152.0到192.168.152.255范围内的地址。
- 主机名，比如：blue。
- 完整的域名，比如：blue.cdors.org。
- 某个IP地址范围，要使用IP地址和网络屏蔽码（netmask）认真地构造之。

我们这个示例的目标是只允许从三台特定的机器上接入远程登录服务，它们的 IP地址是192.156.12.1到192.156.12.3。表示这个地址范围的 IP地址和网络屏蔽码组合是192.156.12.1/255.255.255.252。现在用户的规则语句看起来应该是这个样子的：

```
ipchains -A input -s 192.156.12.1/255.255.255.252
```

还是不能按回车键，你还有事情要做。

8) 如果除了来自这三台机器以外，用户不打算让其他任何东西通过防火墙进入到它后面的机器中去的话，就不用再往下看了。但是这条规则还可以继续细化，告诉 ipchains程序允许通过防火墙进入的接收程序必须要使用哪一种协议。用户可以使用 -p（protocol，协议）标志。远程登录进程使用的是TCP协议。现在这条规则看起来像是：

```
ipchains -A input -s 192.156.12.1/255.255.255.252 -p TCP
```

还是不要按回车键。

9) 围绕某个特定的协议建立一条规则的选择太多了，但是用户可以为你选定的进程指定其端口（port）来细化那条规则。用户可以把端口的名称加到规则的末尾，也可以用一个冒号引导该端口的端口号。从 /etc/services文件中可以查到任何网络服务所使用的端口号。现在这条规则看起来是下列这两行文字中的某个样子：

```
ipchains -A input -s 192.156.12.1/255.255.255.252 -p TCP :23 ALLOW
ipchains -A input -s 192.156.12.1/255.255.255.252 -p TCP Telnet ALLOW
```

现在可以按下回车键了。

10) 用户在确实屏蔽了其他接入类型之前，这条规则实际上还是没有什么用处的。最好的解决方法(因为我们的目的是为用户所有希望允许的东西建立规则)是使用-P（policy，策略）标志（注意是大写字母）屏蔽所有的输入，然后 ipchains就会去检查特定的规则看看到底什么才能被允许进入。这样的一个策略语句如下所示：

```
ipchains -P input DENY
```

21.2.9 安装ipchains

ipchains软件包以一个RPM包的形式保存在发行版本的 CD-ROM光盘上。请按照下面的方法安装它：

1) 以根用户身份登录进入系统。

2) 发行版本的CD-ROM光盘放入CD-ROM驱动器中。

3) 使用“ mount /mnt/cdrom ”命令挂装上这个CD-ROM光盘。

4) 切换到目录：

• Red Hat发行版本的CD-ROM光盘，到/mnt/cdrom/Red Hat/RPMS

• Caldera发行版本的CD-ROM光盘，到/mnt/cdrom/Packages/RPMS

5) 输入“ rpm -ivh ipchains* ”命令安装ipchains软件包。

相关解决方案	请查阅节号
挂装到文件系统上	9.2.2
从文件系统上卸载	9.2.3
安装一个RPM包	15.2.1

21.2.10 在重启动/关机之前保存数据包过滤规则

没有哪个配置文件是用来自动保存数据包过滤规则供你下次启动机器的时候使用的。因此，选择某种方法自己来完成这项工作就十分重要，否则下一次你就还得从头一点一滴地重新写出所有的规则。下面介绍一个保存数据包过滤规则的方法：

1) 以根用户身份登录进入系统。

2) 使用ipchains-save脚本程序把用户已经编写好的规则设置保存到一个文件中去，比如/root/ipchains-settings文件。如下所示，输入“ ipchains-save > /root/ipchains-settings ”。

21.2.11 开机引导后恢复数据包过滤规则

用户在计算机重启动之后，必须恢复数据包过滤规则。请按照下面的方法完成这项任务：

1) 以根用户身份登录进入系统。

2) 使用ipchains-restore脚本程序在某个文件中检索用户已经编写好的规则，比如从文件/root/ipchains-settings中恢复数据包过滤规则。如下所示，输入“ ipchains-restore < /root/ipchains-settings ”。

21.2.12 安装SOCKS

请按照下面的方法安装SOCKS代理防火墙软件包：

1) 从站点www.socks.nec.com处把SOCKS的源代码下载到一个临时目录中。

2) 使用gunzip命令解压缩源代码。

3) 使用tar命令解包源代码。

4) 使用cd命令进入到临时目录中新的SOCKS目录去。

5)使用more命令阅读INSTALL文件，请比较在当前版本中的安装指导说明和我们在这里列出来的内容有没有差别。

6) 请确定C语言编译器已经安装了。可以使用“ rpm -q egcs ”命令检查是否如此。如果这个软件包还没有安装，那么在Red Hat和Caldera两种发行版本的CD-ROM光盘上都能找到。

7) 请确定C语言函数库已经安装了。可以使用“ rpm -q glibc-devel ”命令检查是否如此。如果这个软件包还没有安装，那么在Red Hat和Caldera两种发行版本的CD-ROM光盘上都能找到。

8) 在命令提示符处输入“ ./configure ”。当这个自动配置程序运行的时候，会在屏幕上显示出它正在检查的计算机中的每一个部件。

注意 这一个步骤和下一个步骤可能会需要一段时间才能完成。

9) 如果是要实际编译SOCKS服务器程序，请在命令提示符处输入“ make ”命令。根据使用的计算机的速度，编译时的消息会慢慢地从屏幕上滚过。

10) 输入“ make install ”命令把编译好的服务器程序及其组件安装到它们该去的位置。

11) 输入“ make clean ”命令清除在编译过程中生成的临时文件。

窍门 输入“ man socks5 ”命令就可以找到SOCKS的使用手册页。

相关解决方案	请查阅节号
对文件进行解tar归档操作	15.2.2
查看文本文件，不使用文本编辑器程序	5.2.19
挂装到文件系统上	9.2.2
从文件系统上卸载	9.2.3
安装一个RPM软件包	15.2.1

21.2.13 安装代理服务器程序

在Linux操作系统经常使用的代理服务器程序叫做 Squid。如果你正在运行一个代理防火墙就必须使用这个程序。请按照下列步骤安装这个软件。

1) 在安装着代理防火墙的机器上以根用户身份登录进入系统。

2) 使用“ mount ”命令挂装上系统发行版本的CD-ROM光盘。

3) 把路径切换到系统发行版本的RPM软件包目录中。

4) 使用rpm命令安装squid软件包。

相关解决方案	请查阅节号
挂装到文件系统上	9.2.2
从文件系统上卸载	9.2.3
安装一个RPM软件包	15.2.1

21.2.14 配置SOCKS

请按照下面的方法配置SOCKS代理防火墙：

1) 以根用户身份登录进入系统。

2) 使用vi编辑器打开/etc/inetd.conf文件

3) 在文件中加上一条语句，确保超级守护进程（ inetd）可以在需要的时候打开代理服务器。这一行的内容是：

```
socks stream tcp nowait nobody /usr/local/etc/sockd sockd
```

4) 保存文件并退出编辑器。

窍门 在包含着SOCKS源代码的子目录中输入“ cd examples ”命令就可以进入这个子目录，其中有针对各种类型的网络状况编写的示范性配置文件。用户究竟会怎样配置其初始化过程可能与我们下面给出的例子有很大差异。我们的例子假定用户允许局域

网络中的人们访问因特网上他们想要的任何东西。防火墙最常见设置的唯一目的通常就是希望把不相干的人排除在某个局域以外。

5) 输入“ vi /etc/socks5.conf ”命令编辑这个服务器程序的配置文件。

6) 首先必须要告诉服务器在验证用户是否就是他们自己声称的那个人时接受哪一种验证（认证）方法。这个语句的格式是：“ auth host port method ”，其中：

- host —— 说明请求是从哪里来的
- port —— 说明从哪个端口对用户要求的服务进行响应
- method —— 说明可以接受的验证方法

加上下面的语句就可以允许任何形式的验证检查，只要还使用着验证功能 —— 连字符（-）是代表“ 任意 ”含义的通配符。

```
auth - - -
```

窍门 不使用任何auth语句等于告诉SOCKS任何类型的验证方法都可以接受。但是加上上面这一行至少可以强迫你考虑一下自己到底想干什么，而且在系统开机引导时也不至于太磨蹭。

7) 要想明确地设定什么人可以使用什么样的服务，请按照“ permit authmethod cmdallowed hostfrom hostto portfrom portto user ”的格式再加上一条permit语句，其中：

- authmethod —— 是接受的验证方法。
- cmdallowed —— 说明这条permit规则都分配指定到哪些命令。
- hostfrom —— 说明这条permit规则允许哪些主机使用这些命令。
- hostto —— 说明这条permit规则允许在哪些主机上执行这些命令。
- portfrom —— 说明这条permit规则允许从哪些端口上接受这些命令。
- portto —— 说明这条permit规则允许向哪些端口传递这些命令。
- user —— 明确地说明这条permit规则适用于哪些用户，这是一个可选项。

在语句中加入一些和下面的例子相似的内容，这样就可以允许任何人从 LAN内部的任何端口上执行任何命令；我们下面的例子假定其 LAN覆盖192.168.165.*的地址范围，如下所示：

```
permit - - 192.168.15. - - -
```

8) 保存文件并退出编辑器。

9) 重新启动超级守护进程。

21.2.15 设置Linux机器通过代理防火墙

要想配置SOCKS代理防火墙后面的Linux机器使用这个代理防火墙，请按照下面的方法对每一台客户机器进行下面的设置：

1) 以根用户身份登录进入系统。

2) 编辑/etc/libsocks5.conf文件。

3) 在文件中添上一条语句，允许在这个受保护的 LAN中各个机器之间都可以不通过防火墙彼此直接连接。这需要按照“ noproxy command hostto portto user server ”的格式在文件中加上一条noproxy语句，其中：

- command是不要求通过代理服务器的那些命令。

• hostto是允许客户程序不通过代理服务器就可以联系的计算机。

• portto是允许客户程序不通过代理服务器就可以联系的计算机上的端口。

• user是一个可选项，用来定义哪些用户可以绕过这一行定义的代理。

• server是一个可选项，用来定义这条规则适用于哪一个代理服务器。

在语句中加入一些和下面的例子相似的内容，这样就可以允许 LAN内部的任何人不通过代理服务器就可以在同一个 LAN内的计算机的任何端口上执行任何命令；我们下面的例子假定其LAN覆盖192.168.165.*的地址范围，如下所示：

```
permit - 192.168.15. - -
```

4) 加上一条语句告诉客户程序要想访问 LAN以外的任何东西，就必须经过代理服务器。幸运的是，这个文件中的所有语句的格式都是一致的。唯一的差别是这一行的开头是 sock5，表示它指向SOCKS版本5的服务器。假定代理服务器是安装在 192.168.15.3处的计算机上，那么这一条语句和下面的内容就差不多：

```
socks5 - - - - 192.168.15.3
```

5) 保存文件并退出编辑器。