Linux培训系列

————第七讲

将介绍 TCP/IP 和以太网 Linux 联网的基本原理，说明如何使用 inetd 和 xinetd 超级服务器，提供保护 Linux 系统的重要技巧，还将说明如何设置和使用 Linux 打印服务器。

内容基础，语言简短简洁

红联Linux论坛是致力于Linux技术讨论的站点，目前网站收录的文章及教程基本能满足不同水平的朋友学习。

红联Linux门户： www.linux110.com
红联Linux论坛： www.linuxdiyf.com/bbs
下载:Linux电子书籍：
http://www.linux286.com/linux/linuxdzsj.htm

目录

打印至远程 LPD 服务器

打印至远程的 MS Windows 或 Samba 服务器

Magicfilter

调整 printcap 以指向 Magicfilter

Linux海量文章

海量Linux技术文章

TCP/IP 联网

TCP/IP 联网

发布时间 :2007-01-28 11:42:17

简介

设置一个由大量 Linux 机器组成的基于以太网的局域网（LAN）是常见且相对简单的任务。通常，需要做的就是确保 Linux 系统都安装了某种以太网卡。然后，使用以太网电缆将机器连接到中央以太网集线器或交换机。若所有系统都把对相应的以太网卡的支持（以及 TCP/IP 支持）编译到内核中，则就技术而言，这些系统已经具备了在新的以太网 LAN 上通信的一切条件。

仅有以太网还不太够

发布时间 :2007-01-28 11:42:49

尽管有了让 LAN 工作所需的所有硬件和内核支持，仍不会有多大用处。绝大多数 Linux 应用程序与服务并不使用原始的以太网信息包或帧交换信息。相反，它们使用称为 TCP/IP 的高级协议。毫无疑问，您一定听说过 TCP/IP — 它是一组大体上形成因特网基础的协议（因此得名：传输控制协议／网际协议）。

解决方案：以太网上的 TCP/IP

发布时间 :2007-01-28 11:43:17

于是，解决方案就是配置新的以太网 LAN 以使它可以交换 TCP/IP 流量。要理解解决方案是如何工作的，首先需要知道一点有关以太网的知识。特别地，以太网 LAN 上每台机器中的以太网卡都有唯一的硬件地址。网卡在生产时就被分配了硬件地址，硬件地址看起来与下面相似：

00:01:02:CB:57:3C

IP 地址简介

发布时间 :2007-01-28 11:43:49

这些硬件地址被用做以太网 LAN 上单个系统的唯一地址。使用硬件地址的话，一台机器可以做一些事情，例如，可以向另一台机器发送以太网帧。这一方法存在的问题是基于 TCP/IP 的通信使用另一寻址方案，即称为 IP 地址的寻址方案。IP 地址看起来如下：

192.168.1.1

将 IP 地址与以太网接口关联

发布时间 :2007-01-28 11:44:20

为了使以太网 LAN 使用 TCP/IP，需要将每台机器的以太网卡（因而也就是它的硬件地址）与一个 IP 地址关联。幸运的是，在 Linux 下有一个将 IP 地址与以太网接口关联的简便方法。事实上，如果当前正在通过 Linux 使用以太网，那么分发版的系统初始化脚本中很可能有类似如下的命令：

ifconfig eth0 192.168.1.1 broadcast 192.168.1.255 netmask 255.255.255.0

以上命令中，ifconfig 命令被用来关联 eth0（也就是 eth0 的硬件地址）和 192.168.1.1 IP 地址。另外，还指定了各种其它与 IP 相关的信息，包括广播地址（192.168.1.255）和网络掩码（255.255.255.0）。当命令完成时，eth0 接口将被启用并具有关联的 IP 地址。

使用 ifconfig -a

发布时间 :2007-01-28 11:44:57

可以通过输入 ifconfig -a 查看当前正在运行的所有网络设备，命令执行结果的输出与下面相似：

```
eth0      Link encap:Ethernet  HWaddr 00:01:02:CB:57:3C
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:5 Base address:0xc400

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1065 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1065 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:262542 (256.3 Kb)  TX bytes:262542 (256.3 Kb)
```

从上面的输出可以看到一个已配置的 eth0 接口和 lo（localhost）接口。lo 接口是特殊的虚拟接口，它经过配置以使您在即便没有网络的情况下也可以在本地运行 TCP/IP 应用程序。

TCP/IP 在运行了

发布时间 :2007-01-28 11:45:25

当所有网络接口都设置好并与相应的 IP 地址关联后，以太网网络也可以用来传送 TCP/IP 流量。LAN 上的系统现在可以用 IP 地址相互寻址，象 ping、telnet 和 ssh 这样的命令将在机器间正常工作。

名称解析限制

发布时间 :2007-01-28 11:45:55

然而，尽管能够输入 ping 192.168.1.1 这样的命令，但还不能通过名称引用机器。例如，不能输入 ping mybox 。要做到这一点，需要在每台 Linux 机器上设置名为 /etc/hosts 的文件。在该文件中，要指定一个 IP 地址，以及与每个 IP 地址关联的名称（或多个名称）。因此，如果我有带三个节点的网络，那么我的 /etc/hosts 文件看起来可能与下面相似：

```
127.0.0.1       localhost
192.168.1.1       mybox mybox.gentoo.org
192.168.1.2       testbox testbox.gentoo.org
192.168.1.3      mailbox mailbox.gentoo.org
```

请注意：/etc/hosts 包含 “ localhost ” 到 127.0.0.1 IP 地址的强制映射。我还指定了 LAN 上所有系统的主机名称，包括短名称（“ mybox ”）和全限定名称（“ mybox.gentoo.org ”）。将这个 /etc/hosts 文件复制到每个系统后，我就能够通过名称而不仅是 IP 地址来引用系统了。ping mybox 现在将可以执行了！

使用 DNS

发布时间 :2007-01-28 11:46:26

虽然这种方法可用于小型 LAN，但不便于在拥有许多系统的大型 LAN 上使用。对于这样的配置，通常最好是将所有的 IP 至主机名映射信息存储在一台机器上，然后在这台机器上设置所谓的 “ DNS 服务器 ”（域名服务服务器）。然后，可以配置每台机器来联系这台特别的机器以接收最新的 IP 至名称映射。要做到这一点，可以在每台机器上创建一个与下面相似的 /etc/resolv.conf 文件：

```
domain gentoo.org
nameserver 192.168.1.1
nameserver 192.168.1.2
```

在上面的 /etc/resolv.conf 中，我告诉系统所有非限定主机名（如与 “ testbox.gentoo.org ” 相对的 “ testbox ”）都应视为本地主机名。我还指定一台运行在 192.168.1.1 上的 DNS 服务器，以及一台运行在 192.168.1.2 上的备份服务器。实际上，几乎所有与网络连接的 Linux PC 都已经在其 resolv.conf 文件中指定了名称服务器，即使它们不在 LAN 上，也是如此。这是因为它们在其因特网服务供应商处被配置为使用 DNS 服务器，以便将主机名映射为 IP 地址（这样，那个系统上的用户就可以浏览 Web 并访问象 ibm.com 这样的著名站点，而无需通过 IP 地址来引用它们！）。

连接至外部世界

发布时间 :2007-01-28 11:46:56

说到与因特网连接，该如何配置我们简单的 3 系统 LAN 以使它与“ 外部世界 ”的系统连接呢？通常，我们会购买某种路由器将我们的以太网网络与 DSL 或有线电视调制解调器、T1 或电话线连接。可以用 IP 地址配置这个路由器以使它能够与我们 LAN 上的系统通信。我们可以依次将 LAN 上每个系统都配置为将这个路由器作为其缺省路由或网关使用。这样做的意义在于：对不在我们 LAN 上的系统寻址的任何网络数据将被路由至我们的路由器，路由器再负责将数据转发至我们 LAN 之外的远程系统。通常，分发版的系统初始化脚本会为您处理缺省路由的设置。这些脚本执行该操作所用的命令看起来可能与下面相似：

route add -net default gw 192.168.1.80 netmask 0.0.0.0 metric 1

在上述 route 命令中，缺省路由设置为 192.168.1.80 — 路由器的 IP 地址。要查看系统上所有配置的路由，可以输入 route -n。目标为“ 0.0.0.0 ”的路由是缺省路由。

因特网服务

因特网服务－ inetd 简介

发布时间 :2007-01-28 11:47:47

单个 Linux 系统可以提供数十、甚至数百个网络服务。例如，使用 telnet 程序，您可以访问远程系统上的 telnet 服务。同样地，使用 ftp 程序，您可与远程系统上的 ftp 服务连接。

为了提供这些服务，远程系统运行每个服务器的实例（例如 /usr/sbin/in.telnetd 和 /usr/sbin/in.ftpd）以接受连接或者运行 inetd。inetd 程序接受每个进入的连接，然后根据其连接的类型启动处理该连接的适当的服务。出于这个原因，inetd 也被称为 “ 因特网超级服务器 ” 。

在典型的安装了 Linux 的系统上，inetd 处理大多数进入的连接。只有少数程序（如 sshd 和 lpd）处理它们自己的网络通信而无需依靠 inetd 接受进入的连接。

配置 inetd：/etc/services

发布时间 :2007-01-28 11:48:25

上页提到 inetd 根据类型对进入的连接进行分类。每个进入的连接都在 TCP/IP 头中包含一些标识字段。我们最感兴趣的字段是源地址、目标地址协议和端口号。进入连接由 inetd 根据端口号和协议（通常是 TCP 或 UDP，请查看 /etc/protocols 以获得完整的 inetd 可以提供的服务列表）进行分类。

每行都有如下格式：

service-name port-number/protocol-name aliases # comment

例如，让我们研究最上面的几项：

```
# grep ^[^#] /etc/services | head -5
tcpmux    1/tcp             # TCP port service multiplexer
echo      7/tcp
echo      7/udp
discard   9/tcp   sink null
discard   9/udp   sink null
```

通常，/etc/services 已经包含所有有用的服务名称和端口。如果您希望添加自己的端口，可以查询已分配端口号列表。

配置 inetd；/etc/inetd.conf

发布时间 :2007-01-28 11:48:59

inetd 的实际配置是在 /etc/inetd.conf 中完成的，配置格式如下：

service-name socket-type protocol wait-flag user server-program

因为服务是在 inetd.conf 中由服务名称而不是端口指定的，所以，为了符合由 inetd 处理的条件，必须将服务列在 /etc/services 中。

让我们看看 /etc/inetd.conf 一些常见的行。例如，telnet 和 ftp 服务：

```
# grep ^telnet /etc/inetd.conf
telnet  stream  tcp     nowait  root    /usr/sbin/in.telnetd
```

```
# grep ^ftp /etc/inetd.conf
ftp     stream  tcp     nowait  root    /usr/sbin/in.ftpd -l -a
```

对这两个服务的配置为：使用 TCP 协议，并以 root 用户的身份运行服务器（in.telnetd 或 in.ftpd）。有关 /etc/inetd.conf 中字段的完整说明，请参阅 inetd(8) 手册页。

禁用服务

发布时间 :2007-01-28 11:49:31

在 inetd 中禁用服务很简单：只要在 /etc/inetd.conf 注释掉该服务所在的行即可。您可能不希望完全除去该行，以便以后需要时可以引用它。例如，有些系统管理员出于安全性的原因宁愿禁用 telnet（因为连接完全是明文）：

```
# vi /etc/inetd.conf
[comment out undesired line]

# grep ^.telnet /etc/inetd.conf
#telnet  stream  tcp     nowait  root    /usr/sbin/in.telnetd
```

使用初始脚本停止／启动 inetd

发布时间 :2007-01-28 11:50:06

我们在上页对 /etc/inetd.conf 所做的更改将在重新启动 inetd 程序后才生效。大多数分发版在 /etc/init.d 或 /etc/rc.d/init.d 中有初始脚本：

```
# /etc/rc.d/init.d/inet stop
Stopping INET services:                 [  OK  ]
```

```
# /etc/rc.d/init.d/inet start
Starting INET services:                 [  OK  ]
```

事实上，通常可以使用“ restart ”作为快捷方式：

```
# /etc/rc.d/init.d/inet restart
Stopping INET services:                 [  OK  ]
Starting INET services:                 [  OK  ]
```

用手工方式停止／启动 inetd

发布时间 :2007-01-28 11:50:44

如果上页中的助手脚本不起作用，老式方法甚至更简单。可以使用 killall 命令停止 inetd：

killall inetd

可以在命令行调用 inetd 来简单地再次启动它。它会自动在后台运行：

/usr/sbin/inetd

有一个快捷方式无需停止 inetd 就可命令它重新读取配置文件：只要向它发送 HUP 信号：

killall -HUP inetd

此刻应该不能 telnet 或 ftp 到这个系统，因为 telnet 和 ftp 被禁用。尝试用 telnet localhost 进行检查。如果需要 telnet 或 ftp 访问，所需做的全部就是重新启用它！

以下是我所遇到的情况：

telnet localhost
telnet: Unable to connect to remote host: Connection refused

TCP 封装器简介

发布时间 :2007-01-28 11:51:14

tcp_wrappers 包提供了一个名为 tcpd 的很小的守护程序，该程序由 inetd 而不是实际的服务守护程序调用。tcpd 程序将每个进入连接的源地址编入日志，并可以过滤它们而只允许来自可信系统的连接。

要使用 tcpd，可以按下列方式将它插入到 inetd 中：

```
ftp    stream  tcp    nowait  root    /usr/sbin/tcpd  in.ftpd -l -a
telnet  stream  tcp    nowait  root    /usr/sbin/tcpd  in.telnetd
```

用 TCP 封装器进行日志记录

发布时间 :2007-01-28 11:51:45

缺省情况下，连接不受限制但会被日志记录下来。例如，我们可以重新启动 inetd 以使在前页做的更改生效。然后一些快速调查应该显示已记录的连接：

```
# telnet localhost
login: (press <ctrl-d> to abort)
```

```
# tail -1 /var/log/secure
Feb 12 23:33:05 firewall in.telnetd[440]: connect from 127.0.0.1
```

tcpd 将记录 telnet 连接尝试，因此看起来有些东西在工作了。因为 tcpd 提供一致的连接日志记录服务，这就免除了单个服务守护程序每次自己将连接编入日志的需要。事实上，在接受连接的工作方面，它与 inetd 相似，因为那使每个守护程序不需要接受自己的连接。Linux（UNIX）的简单程度真是不可思议！

用 TCP 封装器限制对本地用户的访问

发布时间 :2007-01-28 11:52:16

tcpd 程序配置为使用两个文件：/etc/hosts.allow 和 /etc/hosts.deny。这两个文件中行的格式为：

daemon_list : client_list [: shell_command]

按以下顺序授权或拒绝访问。搜索在第一次匹配时停止：

当与 /etc/hosts.allow 中的项匹配时，则授权访问
当与 /etc/hosts.deny 中的项匹配时，则拒绝访问
若没有匹配项，则授权访问
例如，若只允许对内部网络进行 telnet 访问，可通过在 /etc/hosts.deny 中设置策略（拒绝除 localhost 以外的其它来源的所有连接）着手：

in.telnetd: ALL EXCEPT LOCAL

用 TCP 封装器将访问限制为已知主机

发布时间 :2007-01-28 11:52:49

无需重新装入 inetd，因为每当 telnet 端口有连接时，就会调用 tcpd。因此我们可以立即尝试：

```
# telnet box.yourdomain.com
Trying 10.0.0.1...
Connected to box.yourdomain.com.
Escape character is '^]'.
Connection closed by foreign host.
```

哦！被拒绝了！（这是人生中为数不多的几次经历：拒绝表示成功。）要重新启用来自自己网络的访问，可在 /etc/hosts.allow 中插入例外：

```
in.telnetd: .yourdomain.com
```

此刻我们就能够再次成功地用 telnet 访问系统了。而这仅仅触及了 tcp_wrappers 的能力的表面。在 tcpd(8) 和 hosts_access(5) 手册页中还有有关 tcp_wrappers 的更多信息。

xinetd：扩展的 inetd

发布时间 :2007-01-28 11:53:19

尽管 inetd 是经典的因特网超级服务器，但最近对它进行了多次改写以试图添加特性和更多的安全性。xinetd 程序在许多新近的分发版（包括 Red Hat 和 Debian）中取代了 inetd。部分扩展的特性是：

访问控制（内置 TCP 封装器）
详尽的日志记录（连接持续时间和失败的连接等等）
来自另一个主机的服务重定向
IPv6 支持
通过代码片段而不是一个汇总文件进行配置

xinetd 配置

发布时间 :2007-01-28 11:54:01

xinetd 的配置文件是 /etc/xinetd.conf。最常见的情况下，那个文件仅包含为其余服务设置缺省配置参数的几行：

```
# cat /etc/xinetd.conf
defaults
{
   instances      = 60
   log_type       = SYSLOG authpriv
   log_on_success = HOST PID
   log_on_failure = HOST RECORD
}
includedir /etc/xinetd.d
```

文件的最后一行指示 xinetd 从 /etc/xinetd.d 目录的文件代码片段读取额外的配置信息。我们快速地看看 telnet 代码片段：

```
# cat /etc/xinetd.d/telnet
service telnet
{
   flags          = REUSE
   socket_type    = stream
   wait           = no
   user           = root
   server         = /usr/sbin/in.telnetd
   log_on_failure += USERID
}
```

如您所见，配置 xinetd 并不困难，而且它比 inetd 更直观。您可以在 xinetd(8)、xinetd.conf(5) 和 xinetd.log(5) 手册页中获得有关 xinetd 的更多信息。

在 Web 上也有关于 inetd、tcp_wrappers 和 xinetd 的大量信息。

安全性概述

安全性概述

发布时间 :2007-01-28 11:54:38

维护一个完全安全的系统是不可能的。然而，只要勤奋，则有可能使 Linux 机器足够安全，并让大多数偶尔出现的骇客、脚本小子（scr ī pt-kiddies）以及其它的“ 坏家伙 ”止步而去骚扰其他人。请记住：仅仅遵循本教程不会产生一个安全的系统。相反，我们希望您接触到主要主题的多个方面，并向您提供一些有关如何入门的有用示例。

Linux 系统安全性可分为两个部分：内部安全性和外部安全性。内部安全性指预防用户无意或恶意地破坏系统。外部安全性指防止未授权用户获得对系统的访问。

本章将首先介绍内部安全性，然后介绍外部安全性，最后介绍一些常规指导原则和技巧。

日志文件的文件权限

发布时间 :2007-01-28 11:55:09

内部安全性可以是很大的任务，这要看您对用户的信任程度。这里介绍的指导原则是设计用来防止偶然用户访问敏感信息和防止不公平地使用系统资源。

至于文件权限，您可能希望修改以下三种情况的权限：

首先，/var/log 中的日志文件不需要是所有人都可以读取的。没有理由让非 root 用户窥视日志。

root 用户其它文件的文件权限

发布时间 :2007-01-28 11:55:39

其次，root 用户的点文件对于普通用户应是不可读的。检查 root 用户主目录中的文件（ls -la）以确保它们受到适当的保护。甚至可以使整个目录仅对 root 用户可读：

```
# cd
# pwd
/root
# chmod 700 .
```

用户文件的文件权限

发布时间 :2007-01-28 11:56:12

最后，用户文件在缺省情况下通常被创建为所有人可读。那可能不是用户所期望的，而且它当然不是最好的策略。应该使用与下面类似的命令在 /etc/profile 中设置缺省的 umask：

```
if [ "$UID" = 0 ]; then
  # root user; set world-readable by default so that
  # installed files can be read by normal users.
  umask 022
else
  # make user files secure unless they explicitly open them
  # for reading by other users
  umask 077
fi
```

应该查询 umask(2) 和 bash(1) 手册页以获取有关设置 umask 的更多信息。请注意：umask(2) 手册页涉及 C 函数，但它所包含的信息也适用于 bash 命令。

查找 SUID/SGID 程序

发布时间 :2007-01-28 11:56:55

寻求 root 访问权的恶意用户总是会在系统上寻找设置了 SUID 或 SGID 位的程序。

应该仔细考虑每个程序以确定是否需要将其 SUID 或 SGID 位打开。系统上有些 SUID/SGID 程序可能是根本不需要的。

要搜索具有这样性质的程序，可使用 find 命令。例如，可以在 /usr 目录中启动对 SUID/SGID 程序的搜索：

```
# cd /usr
# find . -type f -perm +6000 -xdev -exec ls {} \;
-rwsr-sr-x    1 root     root       593972 11-09 12:47 ./bin/gpg
-r-xr-sr-x    1 root     man         38460 01-27 22:13 ./bin/man
-rwsr-xr-x    1 root     root        15576 09-29 22:51 ./bin/rcp
-rwsr-xr-x    1 root     root         8256 09-29 22:51 ./bin/rsh
-rwsr-xr-x    1 root     root        29520 01-17 19:42 ./bin/chfn
-rwsr-xr-x    1 root     root        27500 01-17 19:42 ./bin/chsh
-rwsr-xr-x    1 lp       root        8812 01-15 23:21 ./bin/lppasswd
-rwsr-x---    1 root     cron        10476 01-15 22:16 ./bin/crontab
```

在这个清单中，我已经发现了需要更仔细检查的侯选对象：lppasswd 是 CUPS 打印软件分发版的一部分。因为没有在系统上提供打印服务，所以我会考虑除去 CUPS，那也会除去 lppasswd 程序。lppasswd 中可能没有危及安全性的错误，但为什么要在不使用的程序上冒险呢？同样地，应该关闭所有不使用的服务。您总是可以在需要时再启用它们。

用 ulimit 设置用户限制

发布时间 :2007-01-28 11:57:29

bash 中的 ulimit 命令提供了限制特定用户的资源使用情况的方法。一旦限制降低，则在进程的生命期内无法提高该限制。此外，该限制会被所有子进程继承。结果是：可以在 /etc/profile 中调用 ulimit，而限制将以不能撤消的方式应用于所有用户（假设用户正在运行 bash 或另一个 shell，该 shell 在登录时运行 /etc/profile）。

要检索当前限制，可使用 ulimit -a：

```
# ulimit -a
core file size        (blocks, -c) 0
data seg size         (kbytes, -d) unlimited
file size             (blocks, -f) unlimited
max locked memory     (kbytes, -l) unlimited
max memory size       (kbytes, -m) unlimited
open files                    (-n) 1024
pipe size          (512 bytes, -p) 8
stack size            (kbytes, -s) unlimited
cpu time             (seconds, -t) unlimited
max user processes            (-u) 3071
virtual memory        (kbytes, -v) unlimited
```

以一种能实际提高系统安全性而不会对合法用户造成麻烦的方式设置这些限制是相当复杂的，所以调整这些设置时要小心。

用 ulimit 设置 CPU 时间限制

发布时间 :2007-01-28 11:58:01

作为 ulimit 的一个示例，我们尝试将一个进程的 CPU 时间设置为 1 秒钟，然后用一个忙循环使它超时。一定要确保启动新的 bash 进程（象我们在下面做的那样），以在其中进行尝试；否则将被注销！

```
# time bash
# ulimit -t 1
# while true; do true; done
Killed

real    0m28.941s
user    0m1.990s
sys     0m0.017s
```

在上面的示例中，“ user ”时间加上“ sys ”时间等于该进程所用的全部 CPU 时间。当 bash 进程到达 2 秒标记时，Linux 断定它超过 1 秒的限制，因此该进程被杀掉。酷吧？

注：一秒钟只是示例而已。不要对您的用户这样做！即使几小时也是不对的，因为 X 真地很消耗时间（我当前的会话已用掉了 69+ 小时的 CPU 时间）。在实际的实现中，您可能要对某些项而不是 CPU 时间执行 ulimit。

关闭未使用的网络服务（超级服务器）

发布时间 :2007-01-28 11:58:55

关闭未使用的网络服务一直是提高入侵预防能力的好方法。例如，如果正在运行因特网超级服务器（如本教程前面描述的 inetd 或 xinetd），那么 in.rshd、in.rlogind 和 in.telnetd 通常都在缺省情况下启用。这些网络服务几乎都已被更安全的替代项（如 ssh）所取代。

要在 inetd 中禁用服务，只需在 /etc/inetd.conf 中在适当的行前面加上 “ # ” 将其注释掉；然后重新启动 inetd 即可。（这在本教程前面已有描述，若需要复习，可返回几页快速浏览。）

要在 xinetd 中禁用服务，可以执行与 /etc/xinetd.d 中适当的代码片段相似的工作。例如，要禁用 telnet，可以将 /etc/xinetd.d/telnet 文件的整个内容注释掉，或简单地删除该文件。重新启动 xinetd 以完成此过程。

如果正在结合 inetd 使用 tcpd，或如果正在使用 xinetd，还可以选择限制与可信的主机进行的进入连接。对于 tcpd，可参阅本教程的前几章。对于 xinetd，可在 xinetd.conf(5) 手册页中搜索 “ only_from ” 。

关闭未使用的网络服务（独立服务器）

发布时间 :2007-01-28 11:59:32

有些服务器并不由 inetd 或 xinetd 启动，但却作为“ 独立 ”服务器始终运行着。这样的服务器通常是 atd、lpd 、 sshd、nfsd 和其它服务器。事实上，inetd 和 xinetd 本身都是独立服务器，如果在它们各自的配置文件中注释掉所有的服务，就选择了将它们完全关闭。

独立服务器通常在系统引导或更改运行级别时由 init 系统启动。

要使 init 系统不再启动服务器，在每个运行级别目录中找到指向该服务器启动脚本的符号链接，然后删除它。运行级别目录的名称通常为 /etc/rc3.d 或 /etc/rc.d/rc3.d（针对运行级别 3）。还需要检查其它运行级别。

除去服务的运行级别符号链接后，仍需要关闭当前运行的服务器。最好用服务的初始化脚本完成这一操作，通常可以在 /etc/init.d 或 /etc/rc.d/init.d 中找到这一脚本。例如，要关闭 sshd：

```
# /etc/init.d/sshd stop
* Stopping sshd... [ ok ]
```

测试更改

发布时间 :2007-01-28 12:00:09

在修改 inetd 或 xinetd 配置以禁用或限制服务，或用服务器初始化脚本关闭该服务器后，应该对所做的更改加以测试。可以使用 telnet 客户机通过指定服务名称或号码来测试 tcp 端口。例如，要验证 rlogin 已被禁用：

```
# grep ^login /etc/services
login           513/tcp
# telnet localhost 513
Trying 127.0.0.1...
telnet: Unable to connect to remote host: Connection refused
```

除了标准 telnet 客户机以外，还应考虑使用实用程序以测试系统“ 开放程度 ”的可能性。我们推荐使用 netcat 和 nmap。

ncat 是“ 网络瑞士军刀 ”：它是使用 TCP 或 UDP 协议、跨越网络连接读写数据的简单 UNIX 实用程序。
nmap 是用于网络探测或安全性审计的实用程序。具体而言，nmap 扫描端口以确定哪个端口打开了。

可以在本教程最后一章（参考资料）中找到指向这些实用程序的链接。

拒绝登录以进行维护

发布时间 :2007-01-28 12:00:44

除了以上方法外，还有通过创建 /etc/nologin 文件来拒绝登录的普通方法。通常这一方法用于短期维护操作。仍然可以允许以 root 用户身份登录，但将拒绝以其他用户身份登录。例如：

```
# cat > /etc/nologin
==============================================

  System is currently undergoing maintenance
  until 2:00.  Please come back later.

==============================================
# telnet localhost
login: agriffis
Password:
==============================================

  System is currently undergoing maintenance
  until 2:00.  Please come back later.

==============================================

Login incorrect
```

完成维护后，一定要删除这个文件，否则在您想起以前，没人能登录！我可没这么做过，对，我没有。

iptables（ipchains）简介

发布时间 :2007-01-28 12:01:16

iptables 和 ipchains 命令用于在运行的 Linux 内核中调整和检查网络信息包过滤器规则。ipchains 命令用于 2.2.x 版本内核，尽管它仍可用于 2.4.x 内核，但已被 iptables 取代。

可设置信息包过滤器规则进行防火墙和路由器的活动。可以对 iptables 命令加上 -L 选项来检查当前的规则：

```
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

这是一个非常开放的系统示例，没有启用路由或防火墙。

iptables 和 Linux 信息包过滤器

发布时间 :2007-01-28 12:01:50

有效地使用 Linux 信息包过滤器需要对 TCP/IP 联网及其如何在 Linux 内核中实现有扎实的理解。netfilter 主页（请参阅本教程最后一章的参考资料，以获得链接）是学习更多知识的好去处。

在能自如地构建自己的规则集以前，有许多脚本可以让您入门，只要您信任它们的作者即可。最完整的脚本之一是 gShield（请参阅参考资料）。您可以调整其注释良好且相当简单的配置文件以设置信息包过滤器规则最常规的格式。

入侵检测 — 系统日志（syslog）

发布时间 :2007-01-28 12:02:23

入侵检测通常被那些相信自己安置的入侵预防设备的系统管理员所忽略。不幸的是，这意味着一旦黑客找到可以入侵的细微漏洞，在注意到他们的存在以前，系统可能很长一段时间都处于他们的控制之下。

入侵检测最基本的形式是注意系统日志。这些文件通常出现在 /var/log 目录中，不过实际的文件名会因分发版和配置而有所不同。

```
# less /var/log/messages
Feb 17 21:21:38 [kernel]   Vendor: SONY      Model: CD-RW  CRX140E    Rev: 1.0n
Feb 17 21:21:39 [kernel] eth0: generic NE2100 found at 0xe800, Version 0x031243,
DMA 3 (autodetected), IRQ 11 (autodetected).
Feb 17 21:21:39 [kernel] ne.c:v1.10 9/23/94 Donald Becker ( becker@scyld.com)
Feb 17 21:22:11 [kernel] NVRM: AGPGART: VIA MVP3 chipset
Feb 17 21:22:11 [kernel] NVRM: AGPGART: allocated 16 pages
Feb 17 22:20:05 [PAM_pwdb] authentication failure; (uid=1000)
  -> root for su service
Feb 17 22:20:06 [su] pam_authenticate: Authentication failure
Feb 17 22:20:06 [su] - pts/3 chouser-root
```

要理解所有这些消息可能需要进行一些实践，但大多数重要消息都相当清楚。例如，在日志的末尾，我们可以看到用户“ chouser ”试图使用 su 成为 root 用户，但失败了。

入侵检测 — tripwire

发布时间 :2007-01-28 12:02:52

有许多可用的包可以对整个文件系统进行“ 快照 ”，然后将它与较早的快照比较以了解什么发生了更改。若能明确地定义哪些文件作为系统正常操作的一部分应该发生更改，则这些包能很快提醒黑客的存在及其活动。

Tripwire 是最流行的入侵检测包之一（请参阅本教程末尾的参考资料以获取链接）。安装 tripwire 后，必须定制它的配置文件以使它知道哪些文件应该更改而哪些不应更改。还需要告诉它如何向您发送有关发生什么更改的报告，以及它应隔多久运行一次（通常每天一次）。

入侵检测 — portsentry

发布时间 :2007-01-28 12:03:25

PortSentry 包来自 Psionic Technologies，它实际上有点介于入侵预防与检测之间。该包监控网络连接，并且如果它看到任何它认为“ 可疑 ”的与系统连接的尝试，它会把这一事件编入日志然后阻止它再次发生。该包也可以在本教程末尾的参考资料中找到。

当安装了该包并运行它时，将能够在 syslog 中看到所有尝试的连接，并看到 PortSentry 如何对它们做出反应：

```
# tail /var/log/messages
Oct 15 00:21:24 mycroft portsentry[603]: attackalert:
  SYN/Normal scan from host: 302.174.40.34/302.174.40.34 to TCP port: 111
Oct 15 00:21:24 mycroft portsentry[603]: attackalert:
  Host 302.174.40.34  has  been  blocked  via wrappers with string:
  "ALL: 302.174.40.34"
Oct 15 00:21:24 mycroft portsentry[603]: attackalert:
  Host 302.174.40.34 has been blocked via dropped route using command:
  "/sbin/route add -host 302.174.40.34 reject"
Oct 15 00:21:24 mycroft portsentry[603]: attackalert:
  SYN/Normal scan from host: 302.174.40.34/302.174.40.34 to TCP port: 111
Oct 15 00:21:24 mycroft portsentry[603]: attackalert:
  Host: 302.174.40.34/302.174.40.34 is already blocked Ignoring
Oct 15 00:33:59 mycroft portsentry[603]: attackalert:
  SYN/Normal scan from host: 302.106.103.19/302.106.103.19 to TCP port: 111
Oct 15 00:33:59 mycroft portsentry[603]: attackalert:
  Host 302.106.103.19 has been blocked via wrappers with string:
  "ALL: 302.106.103.19"
Oct 15 00:33:59 mycroft portsentry[603]: attackalert:
  Host 302.106.103.19 has been blocked via dropped route using command:
  "/sbin/route add -host 302.106.103.19 reject"
```

常规指南：保持软件为最新

发布时间 :2007-01-28 12:03:54

因为所有软件都可能有安全性漏洞，所以重要的是：只要获得包的安全性修正包就立刻安装。这是安全专家最常提出的一条建议，也是管理员新手们最常忽略的一条建议。不要吃过苦头才吸取教训 — 机器因为您忽视了使补丁程序保持最新而被人通过存在数年之久的后门侵入。

对于开放源码和封闭源码哪个更安全的争论非常激烈。迄今最好的结论是：管理正确时，两者都足够安全，这里的管理包括保持安全性补丁程序为最新！

有几个网站可以帮助保持软件为最新，并有助于提防已知的威胁。包括特别注意安全性的 CERT 和 SecurityFocus 的 BugTraq 列表，以及通常的软件更新站点（象freshmeat.net）和分发版的主页。我们还将在参考资料中重复这些 URL，不过安全性真的非常重要 — 如果还不熟悉这些站点的话，建议您现在就花几分钟访问头两个站点。

打印

打印

发布时间 :2007-01-28 12:04:37

这一章将介绍 Linux 上的经典 UNIX 打印系统（有时被称为 Berkeley LPD）的设置和使用。其它可用于 Linux 的系统则不在这里介绍；请参阅本教程末尾的参考资料一章以获取有关这些系统的信息。

物理上安装打印机超出了本教程的范围。当打印机正确连接后，则要安装打印假脱机程序守护程序，以使网络上的机器（包括运行假脱机程序的机器）能够将打印作业发送给打印机。

安装打印假脱机程序守护程序（lpd）

发布时间 :2007-01-28 12:05:11

最好的 LPD 打印假脱机程序之一是 LPRng。其安装方法取决于分发版；请参阅 LPI 102 系列第 1 部分以获取有关在 Red Hat 或 Debian 中安装软件包的详细信息。

安装打印假脱机程序守护程序（正式名称为行式打印机守护程序）以后，就可以从命令行运行。以普通用户身份登录，然后试着运行以下命令：

```
$ /usr/sbin/lpd --help
--X option form illegal
usage: lpd [-FV] [-D dbg] [-L log]
 Options
 -D dbg      - set debug level and flags
               Example: -D10,remote=5
               set debug level to 10, remote flag = 5
 -F          - run in foreground, log to STDERR
              Example: -D10,remote=5
 -L logfile  - append log information to logfile
 -V          - show version info
```

既然已安装了守护程序，则应确保将它设置为自动运行。

基本打印机设置（/etc/printcap）

发布时间 :2007-01-28 12:05:48

打印假脱机程序守护程序起着一种管道的作用。它接受来自各个打印客户机的打印作业，然后将这些作业传递到适当的打印机。当打印机忙时，这些作业就“ 假脱机 ”，等待打印机会。

当在本地打印机上打印时，该“ 管道 ”的两端都由配置文件 /etc/printcap（有时位于 /etc/lprng/printcap）描述。printcap（printer capabilities 的缩写）中的每一项描述一个打印假脱机文件：

```
$ more /etc/printcap
lp|Generic dot-matrix printer entry:\
        :lp=/dev/lp0:\
        :sd=/var/spool/lpd/lp:\
        :pl#66:\
        :pw#80:\
        :pc#150:\
        :mx#0:\
        :sh:
```

请注意：项的最后一行没有尾随的反斜杠（\）。

您的分发版可能有其它项，并且可能更复杂，但它们都大致有这样的形式。首先是项的名称 lp，随后是对这个假脱机文件较长的描述。关键字／值对 lp=/dev/lp0 指定将要打印假脱机文件中打印作业的 Linux 设备，而 sd 关键字则给出打印作业前存放它们的目录。

余下的关键字／值对则提供有关连接到 /dev/lp0 的打印机类型的详细信息。printcap 手册页对它们做了描述，稍后我们将介绍其中的一部分。

创建假脱机文件目录

发布时间 :2007-01-28 12:06:25

如果创建一个打印假脱机项，则需要确保假脱机文件的目录存在并且具有正确的权限。如果希望打印机守护程序（通常以用户 lp 的身份运行）能访问假脱机文件目录，则必须以 root 用户的身份运行以下命令：

```
# mkdir -p /var/spool/lpd/lp
# chown lp /var/spool/lpd/lp
# chmod 700 /var/spool/lpd/lp
# checkpc -f
# /etc/init.d/lprng restart
```

LPRng 包含一个用于检查 printcap 的有用工具。它甚至会为您设置假脱机文件目录（如果您忘了以手工方式这么做的话）：

```
# checkpc -f
```

最后，重新启动 lpd。为了使更改生效，每次更改 printcap 时都需要这么做。您可能需要使用 lpd 而不是 lprng：

```
# /etc/init.d/lprng restart
```

较老的 Berkeley 打印系统不包含 checkpc 工具，所以您必须亲自在各台打印机上打印测试页，以确保 printcap 和打印假脱机文件目录是正确的。

使用打印假脱机程序客户机

发布时间 :2007-01-28 12:07:05

打印假脱机程序本身带有几个客户机以便与服务器守护程序通信。使用最多的可能是 lpr，它仅仅将文件发送至服务器以在假脱机文件中排队然后打印。要尝试该程序，首先找到或制作一个小的样本文本文件。然后输入命令：

```
$ lpr sample.txt
```

若该命令起作用，则在屏幕上应该看不到响应，但打印机应该开始运行，而且应很快就能打印出该样本文本的硬拷贝。如果该命令执行的输出看起来不太正确，不必担心；稍后我们将设置过滤器，它应能确保所有种类的文件格式都能正确地打印。

可以用 lpq 命令检查打印假脱机文件队列中的打印作业列表。选项 -P 指定要显示的队列名称；如果不使用该选项，则 lpq 将使用缺省打印假脱机文件（就象 lpr 在前面所做的那样）：

```
$ lpq -Plp
Printer: lp@localhost 'Generic dot-matrix printer entry'
 Queue: 1 printable job
 Server: pid 1671 active
 Unspooler: pid 1672 active
 Rank   Owner/ID               Class Job Files                 Size Time
active  chouser@localhost+670       A   670 sample.txt             8 21:57:30
```

如果要停止打印作业，可以使用 lprm 命令。若一个作业花的时间过长，或者用户不小心发送了多份相同文件，则可能要执行该命令。只要从上面列出的 lpq 命令复制作业标识即可：

```
$ lprm  chouser@localhost+670
Printer  lp@localhost
  checking perms  'chouser@localhost+670'
  dequeued  'chouser@localhost+670'
```

可以使用交互式工具 lpc 对打印假脱机文件进行许多其它操作。请参阅 lpc 手册页以获取详细信息。

打印至远程 LPD 服务器

发布时间 :2007-01-28 12:07:46

即使本地机器上没有打印机，仍可以使用 lpd 跨越网络将打印作业发送至与别的机器相连的打印机。在客户机机器上，可以向 /etc/printcap 添加一条看似本地打印机而实际上将打印作业路由至服务器机器的打印假脱机文件项。该项看起来应与下面相似：

```
farawaylp|Remote printer entry:\
        :rm=faraway:\
        :rp=lp:\
        :sd=/var/spool/lpd/farawaylp:\
        :mx#0:\
        :sh:
```

这里我们希望执行打印作业的机器名称是 faraway，而那台机器上打印机的名称是 lp。假脱机文件目录 /var/spool/lpd/farawaylp 是打印作业在能够被发送至远程打印假脱机程序以前在本地保存的位置，而且在打印作业能发送到打印机以前，可能还要在远程打印假脱机程序处再次对它们进行假脱机处理。同样地，将需要创建这个假脱机文件目录并设置其权限：

```
# mkdir -p /var/spool/lpd/farawaylp
# chown lp /var/spool/lpd/farawaylp
# chmod 700 /var/spool/lpd/farawaylp
# checkpc -f
# /etc/init.d/lprng restart
```

在本地，我们将这个远程打印机命名为 farawaylp，因此我们可以将打印作业发送至 farawaylp：

```
$ lpr -Pfarawaylp sample.txt
```

打印至远程的 MS Windows 或 Samba 服务器

发布时间 :2007-01-28 12:08:33

感谢 Samba，打印至远程 Microsoft Windows 打印服务器只稍稍复杂一点。首先，添加本地 printcap 项：

```
smb|Remote windows printer:\
        :if=/usr/bin/smbprint:\
        :lp=/dev/null:\
        :sd=/var/spool/lpd/smb:\
        :mx#0:
```

这里新的关键字是 if，即输入过滤器。将它指向 smbprint 脚本将使打印作业被发送至 Windows 服务器而不是 lp 设备。我们仍必须列出打印守护程序用于锁定而使用的设备（此例中是 /dev/null）。但实际上将没有打印作业被发送到那里。

不要忘记创建假脱机文件目录

```
# mkdir -p /var/spool/lpd/smb
# chown lp /var/spool/lpd/smb
# chmod 700 /var/spool/lpd/smb
# checkpc -f
# /etc/init.d/lprng restart
```

在您喜爱的编辑器中，在上面命名的假脱机文件目录中创建一个 .config 文件。在本例中，该文件为 /var/spool/lpd/smb/.config：

```
server="WindowsServerName"
service="PrinterName"
password=""
user=""
```

调整这些值以指向希望使用的 Windows 机器和打印机名称，然后就可以使用以下命令：

```
$ lpr -Psmb sample.txt
```

smbprint 脚本应该与 Samba 一起提供，但该脚本并不包含在所有分发版中。如果在系统上找不到这个脚本，可以从 Samba HOWTO 获得。

Magicfilter

发布时间 :2007-01-28 12:09:03

迄今我们只尝试了打印文本文件，这还不是特别令人兴奋。通常，任何一台打印机只能打印一种格式的图形文件 — 然而我们希望打印的格式有几十种：Postscr ī pt、gif、jpeg 等等。名为 Magicfilter 的程序起着输入过滤器的作用，很象 smbprint 的所为。Magicfilter 并不转换文件格式，而是提供标识您正在尝试打印的文档类型的框架，然后通过适当的转换工具运行该文档：转换工具必须单独安装。到目前为止，最重要的转换工具是 Ghostscr ī pt，它可以将文件从 Postscr ī pt 格式转换成许多打印机的本机格式。

调整 printcap 以指向 Magicfilter

发布时间 :2007-01-28 12:09:45

安装这些工具后，只需再调整 printtcap 一次即可。添加 if 关键字以指向与打印机配合的 Magicfilter：

```
lp|The EPSON Stylus Color 777 sitting under my desk:\
        :if=/usr/share/magicfilter/StylusColor-777@720dpi-filter:\
        :gqfilter:\
        :lp=/dev/usb/lp0:\
        :sd=/var/spool/lpd/lp:\
        :pl#66:\
        :pw#80:\
        :pc#150:\
        :mx#0:\
        :sh:
```

在 /usr/share/magicfilter 中有用于许多不同打印机和打印机设置的过滤器，因此要确保使用的是适合您打印机的过滤器。每个过滤器都是一个文本文件，并且打印机的全称常常位于顶部。当您不清楚过滤器的文件名是什么时，这会对您有所帮助。

我还向这个 printcap 项添加了 gqfilter 标志，这样，即使打印作业来自远程打印机，也可使用输入过滤器。这种方法只适用于 LPRng。

因为较早的时候就设置了 /var/spool/lpd/lp 打印假脱机文件目录，所以我只需要检查 printcap 语法，然后重新启动服务器：

```
# checkpc -f
# /etc/init.d/lprng restart
```

现在您能够打印各种文档，包括 Postscr ī pt 文件。换句话说，现在可以从您喜爱的 Web 浏览器的菜单中选择“ Print ”来进行工作了。

Linux海量文章

海量Linux技术文章

发布时间 :2006-11-24 16:50:29

下面是linux技术文章快速入口。需要联网：
Linux 技术交流

http://www.linuxdiyf.com/bbs/forum-3-1.html

Linux 应用

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=1

Linux 安装及学习指导

http://www.linuxdiyf.com/bbs/forum-45-1.html

Linux 系统安装

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=45&filter=type&typeid=11

Linux 学习指导

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=45&filter=type&typeid=12

Linux 软件安装

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=45&filter=type&typeid=13

shell

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=3

Linux 壁纸

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=4

红旗

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=5

Redhat

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=6

SuSE

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=7

Linux 认证

http://www.linuxdiyf.com/bbs/forumdisplay.php?fid=3&filter=type&typeid=9

Linux下载分享{酷件、书籍、视频分享 }

http://www.linuxdiyf.com/bbs/forum-6-1.html

服务器应用

http://www.linuxdiyf.com/bbs/forum-7-1.html

数据库应用

http://www.linuxdiyf.com/bbs/forum-8-1.html

Linux 编程与内核

http://www.linuxdiyf.com/bbs/forum-9-1.html

UniX 技术文章

http://www.linuxdiyf.com/bbs/forum-32-1.html

Linux 业界声音、新闻

http://www.linuxdiyf.com/bbs/forum-11-1.html

Linux 人才招聘信息

http://www.linuxdiyf.com/bbs/forum-46-1.html

网络转载，感谢原创作者！
制作：红联Linux论坛
祝您阅读愉快！