

Ralf Spenneberg

# Linux-Firewalls mit iptables & Co.

Sicherheit mit Kernel 2.4 und 2.6  
für Linux-Server und -Netzwerke



 ADDISON-WESLEY

---

An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England  
Don Mills, Ontario • Sydney • Mexico City  
Madrid • Amsterdam



# 6

## Härtung eines Linux-Systems

Wenn Sie eine Linux-Distribution als Firewall-System nutzen wollen, sollte diese gehärtet sein. Die üblichen Distributionen (SUSE Linux, SUSE Linux OSS, Fedora Core und Debian) sind noch nicht an die besonderen Anforderungen einer Firewall angepasst. Wenn Sie eine für diesen speziellen Zweck vorbereitete Distribution nutzen, haben die Hersteller meist bereits eine gute Vorarbeit geleistet. Dennoch kann es nicht schaden, wenn Sie trotzdem noch einmal einen Blick auf die Härtung werfen.

Was ist Härtung? Eine gute Frage. Es gibt keine allgemeine Antwort. Kein RFC oder Standard definiert die Härtung eines Betriebssystems. Im Folgenden gebe ich Ihnen meine Vorstellung von der Härtung eines Betriebssystems wieder.

Da die Härtung eines Betriebssystems keine einfache Aufgabe ist, schnell wichtige Punkte vergessen werden und häufig ähnliche Systeme anschließend unterschiedlich konfiguriert wurden, stelle ich Ihnen anschließend im Kapitel Bastille-Linux vor. Bastille-Linux ist ein Werkzeug für die wichtigsten Linux-Distributionen (und auch Mac OS X und HP-UX), das Sie bei dieser Arbeit menügeführt unterstützt. Selbst wenn Sie es nicht für diesen Zweck einsetzen möchten, können Sie es sehr gut als Assessment-Werkzeug verwenden, um Ihre manuelle Konfiguration zu prüfen.

### 6.1 Warum sollte eine Firewall gehärtet werden?

Um zu klären, warum eine Firewall gehärtet werden sollte, sollten wir uns zunächst ansehen, was Härtung bedeutet. Wenn Sie eine aktuelle Linux-Distribution durch bloßes Anwählen der Default-Einstellungen installieren, so installieren Sie zahllose, für eine Firewall überflüssige Softwarepakete. Diese Softwarepakete sind häufig nicht nur überflüssig, sondern können auch Sicherheitslücken enthalten. Wenn die Distribution nach einem Neustart auch einige dieser überflüssigen Netzwerkdienste aktiviert, so dass sie von außen erreichbar sind, könnten diese Dienste auch über das Netz angreifbar sein.

Natürlich installieren Sie Ihre Firewall-Regeln so, dass ein Zugriff auf die Dienste gar nicht erst möglich ist. Aber vielleicht unterläuft Ihnen bei der Konfiguration ein Fehler? Um derartige Probleme von vornherein zu umgehen, sollten Sie diese Dienste deaktivieren und besser gar nicht installieren. Auch sollten alle überflüs-

sigen Benutzerkonten von dem System entfernt werden. Der Bootvorgang sollte auf Möglichkeiten zur Kompromittierung des Systems überprüft werden, und die Rechte der Verzeichnisse und der ausführbaren Befehle (besonders die SetUID- und SetGID-Rechte) sollten überprüft und angepasst werden.

## 6.2 Installation des Linux-Systems

Ich möchte Ihnen hier nicht eine Empfehlung für eine Linux-Distribution aussprechen, denn ich denke, dass es nicht *die beste* Distribution gibt. Sicherlich gibt es Unterschiede im Design und in der Anwendung, und es mag subjektive Gründe geben, warum Sie und ich die eine oder andere Distribution vorziehen. Allerdings können Sie jede Distribution für eine Firewall einsetzen. Es ist nicht erforderlich, eine spezielle Distribution wie Fli4L<sup>1</sup> oder IPCop<sup>2</sup> zu nutzen. Wenn Sie diese Distributionen nicht kennen, sind sie sicherlich einen Blick wert. Ob es allerdings Sinn macht, beim Aufbau einer Firewall eine neue Distribution zu lernen, möchte ich bezweifeln. Verwenden Sie die Distribution, die Sie am besten kennen und konfigurieren können. Lediglich wenn Sie vor unüberwindbare Probleme stoßen, die der Distribution zuzuschreiben sind, würde ich Ihnen raten, eine andere Distribution zu testen.

Dennoch gibt es einige allgemeine Hinweise für die Installation des Systems.

Zunächst sollten Sie sich Gedanken über die Ausfallsicherheit des Systems machen. Die Firewall stellt Ihre Internetverbindung dar. Wenn diese ausfällt, kann das verheerende Konsequenzen für Ihr Geschäftsmodell haben, wenn Ihre gesamte Kommunikation (E-Mail, VoIP etc.) auf dem Internet basiert. Falls es sich lediglich um eine Firewall für private Zwecke handelt, ist das sicherlich nicht so tragisch, aber auch hier kann ein Ausfall des Internets unerwünscht sein.

Wählen Sie also eine möglichst robuste Hardware. Wenn das System ununterbrochen laufen soll, achten Sie besonders auf die Luftzufuhr und Kühlung. Wählen Sie lieber einen stromsparenden Prozessor, der auch weniger Abwärme erzeugt als die neueste Heizplatte. Ein reiner Paketfilter benötigt nicht einen Pentium 4 mit 3 GHz. Wenn Sie auch noch andere Funktionen (z.B. VPN oder Proxy) auf demselben System betreiben möchten, sieht das natürlich anders aus. Aber meist reicht auch hier noch ein einfacher stromsparender Prozessor.

Investieren Sie Ihr Geld lieber in zwei Festplatten und wenn möglich in ein Mainboard mit redundanter Stromversorgung über zwei Netzteile. Falls es Ihr Budget hergibt, ist auch ein Hardware-Raid-Controller sinnvoll. Achten Sie aber darauf, dass der Controller auch von Linux unterstützt wird. Ein Festplattenspiegel mit Raid 1 reicht vollkommen aus. Ein Raid 5 ist nicht erforderlich.

Selbst wenn Sie nicht das Geld für einen Raid-Controller ausgeben möchten, empfehle ich den Einsatz von zwei Festplatten mit einem Software-Raid 1. In einem

---

<sup>1</sup> <http://www.fli4l.de>

<sup>2</sup> <http://www.ipcop.org>

Fehlerfall ist die Wiederherstellung zwar etwas aufwendiger, jedoch verlieren Sie nicht Ihre Daten. Die meisten großen Distributionen können bereits bei der Installation ein Software-Raid einrichten und installieren. Ansonsten finden Sie in dem Linux Software-Raid-Howto (<http://www.tldp.org/HOWTO/Software-RAID-HOWTO.html>) entsprechende Hinweise für die Einrichtung.

Der wesentliche Vorteil bei einem Raid-System ist die Tatsache, dass das System auch bei Ausfall einer Festplatte weiterarbeitet. Aus demselben Grund empfehle ich auch die Verwendung redundanter Netzteile. Die Netzteile und die Festplatten sind am häufigsten für den Ausfall eines Systems verantwortlich. Sind beide redundant vorhanden, arbeitet das System trotz Ausfall einer Komponente weiter.

Wenn Sie das ganze Firewall-System redundant implementieren möchten, sollten Sie das Kapitel über den HA-Firewall-Cluster (Kapitel 26) lesen. Dort wird die Konfiguration eines Firewall-Clusters auf der Basis von Linux beschrieben.

Sobald Sie die Hardware ausgewählt und angeschafft haben, müssen Sie sich mit der Installation beschäftigen. Hier sind zunächst die Partitionierung und die Paketauswahl wichtig. Bei der Partitionierung sollten Sie bedenken, dass die spätere Firewall nur einen geringen Platz für die binären Pakete benötigt. Jedoch werden die Systeme Protokolldateien erzeugen, die schnell sehr groß werden können. Dies sollten Sie in der Partionierung berücksichtigen. Folgendes Schema hat sich bewährt:

- `/boot` 128M. So haben Sie immer ausreichend Platz für Kernel-Updates.
- `/usr` 1G. Sie installieren nur wenige Pakete.
- `/` 512M. Die Konfigurationsdateien (`/etc/`) und Geräte (`/dev`) benötigen nicht viel Speicher.
- `/tmp` 256M. Der temporäre Speicher sollte auf einer eigenen Partition liegen, so dass er nicht die Root-Partition füllen kann.
- `/var` Rest. Hier liegen die Protokolldateien. Dieses Verzeichnis wird bei der Verwendung der Firewall wachsen. Eventuell ist es sinnvoll, das Verzeichnis `/var/log` auch noch auf eine eigene Partition auszulagern.

Bei der Paketauswahl sollten Sie dann eine Minimalinstallation wählen. Sollte der oben angegebene Platz nicht ausreichen, müssen Sie die eine oder andere Partitionsgröße anpassen. Allerdings kenne ich keine Distribution, bei der das aktuell der Fall wäre.

Nach der Installation sollten Sie sich für den unwahrscheinlichen Fall, dass beide Festplatten ausfallen, mit einer Disaster-Recovery-Lösung wappnen. Dies ist umso wichtiger, wenn Sie nur eine Festplatte verwenden. Zwei Disaster-Recovery-Lösungen sind mir bekannt und werden von mir verwendet:

- Mondo-Rescue: <http://www.mondorescue.org/>
- Mkdrec: <http://mkdrec.ota.be/>

Beide Programme analysieren das System und erzeugen ein oder mehrere bootfähige CD-Images, die für die Wiederherstellung des Systems genutzt werden kön-

nen. Dabei wird das System nicht blockweise kopiert, sondern die Programme analysieren die Partitionierung und Formatierung, sichern alle Dateien und können diese dann auf einer neuen, mindestens gleich großen Festplatte wiederherstellen. Beide Systeme können mit Software-Raid- und Logical-Volume-Manager-Partitionen (LVM) umgehen.

Es empfiehlt sich, eine der beiden Lösungen zu installieren und zu testen. Später sollten Sie dann nach jeder wesentlichen Änderung des Systems oder vielleicht monatlich ein neues Disaster-Recovery-Image erzeugen. Achten Sie darauf, dass die Protokolle nicht gesichert werden. Dies bläht die Images nur unnötig auf.

## 6.3 Updates

Wenn Sie Ihr Linux-System mit Hilfe von CDs installieren, dann sind die darauf enthaltenen Dateien sicherlich bei der Installation zumindest teilweise veraltet. Bei vielen Paketen wurden Fehler gefunden, und bei einigen Paketen sind die Fehler möglicherweise sogar sicherheitsrelevant. Sie sollten daher, direkt anschließend an die Installation, die neuesten Updates installieren. Sehr einfach ist das, wenn Ihre Distribution dafür einen Mechanismus anbietet. Dies ist unter anderem bei Debian, SUSE, RedHat und Fedora Core der Fall. Die Anwendung dieser Mechanismen wird weiter unten erläutert.

Sie sollten sich auch überlegen, ob Sie diesen Update-Mechanismus automatisch einbinden wollen. Für mich persönlich habe ich auf vielen Systemen diese Entscheidung getroffen. So werden auf meinen Systemen die Updates stündlich geprüft und eingespielt. Manch einer mag nun die Nase rümpfen. Was passiert, wenn das Update fehlschlägt? Nun, ich habe für mich eine Risikoabwägung durchgeführt. Was passiert, wenn eine Sicherheitslücke bekannt wird und ich nicht rechtzeitig das Update durchführe? Der Schaden ist ungemein größer, als wenn automatisch ein Update durchgeführt wird, dieses fehlschlägt und der Dienst anschließend für einige Zeit nicht zur Verfügung steht. Er kann zumindest nicht mehr angegriffen werden.

Außerdem sollten Sie sich überlegen, wie Sie vorgehen, wenn Sie selbst manuell das Paket aktualisieren. Wahrscheinlich werden Sie das Update von Ihrer Distribution installieren. Im Grunde führen Sie dieselben Schritte durch, nur manuell und später. Das Update kann genauso fehlschlagen. Sie müssen das System genauso reparieren.

Ich persönlich habe erst ein einziges Mal eine schlechte Erfahrung mit der automatischen Aktualisierung der Pakete gemacht. Dabei handelte es sich um das `bind`-Paket der RHEL-3 Distribution. Nachdem das Update eingespielt wurde, wurde der Nameserver gestoppt, aber nicht wieder gestartet. Da es sich um einen primären Nameserver handelte, für den es noch weitere sekundäre Nameserver gab, fiel dies nicht sofort auf. Erst nach zwei Wochen, als die sekundären Nameserver aufgrund des fehlenden primären Nameservers ihre Arbeit einstellten, funktionierte die Namensauflösung nicht mehr.

Wenn Sie wie ich nachts schlafen, ab und zu in den Urlaub fahren und nicht 24 Stunden am Tag ein Support-Team beschäftigen möchten, kann ich Ihnen nur empfehlen, eine automatische Aktualisierung der kritischen Systeme einzurichten.

### 6.3.1 Debian-Updates

Auf einem Debian-System ist ein Update einfach durchzuführen. Hierzu rufen Sie nacheinander die folgenden Befehle auf:

```
/usr/bin/apt-get update -q -y  
/usr/bin/apt-get upgrade -q -y
```

Der erste Befehl aktualisiert die lokal vorgehaltenen Paketlisten der verfügbaren Programmpakete. Der zweite Befehl aktualisiert die Pakete anhand dieser Liste. Die Option `-q` unterdrückt die Ausgabe von unnötigen Informationen (quiet), und die Option `-y` beantwortet automatisch alle möglichen Fragen mit Ja (yes).

### 6.3.2 SUSE-Updates

Bei SUSE kann das YaST-Online-Update (YOU, Abbildung 6.1) über das Werkzeug YaST konfiguriert werden. Hierzu müssen Sie aber auch sicherstellen, dass das Online-Update-Paket installiert wurde. Bei dem Einsatz eines SUSE-Linux-Enterprise-Servers (SLES) benötigen Sie für das Online-Update auch noch ein Kennwort.

### 6.3.3 Red Hat-Updates

Bei den kommerziellen Red Hat Enterprise Linux-(RHEL-)Systemen erfolgt das Update der Pakete mit dem Kommando `up2date`. Dieses Kommando besitzt eine grafische Oberfläche. Jedoch können Sie diese auch mit der Option `-nox` deaktivieren. Um dieses Werkzeug nutzen zu können, benötigen Sie ähnlich wie beim SLES ein Red Hat Network-(RHN-)Login und einen Wartungsvertrag.

### 6.3.4 Fedora Core-Updates

Bei der freien Fedora Core-Distribution ist ein Update der Pakete sehr einfach mit dem Befehl `yum` möglich. Ein `yum update` aktualisiert das System mit den aktuellen Paketen. Die Option `-y` beantwortet auch hier alle Fragen mit Ja.

## 6.4 Deaktivieren überflüssiger Dienste

Sobald Sie die Installation des Systems abgeschlossen und eventuelle Updates eingespielt haben, sollten Sie überflüssige Dienste deaktivieren.

Wie finden Sie nun diese Dienste? Im Grunde gibt es drei verschiedene Varianten, wie auf einem typischen Linux-System ein Dienst automatisch gestartet wird.

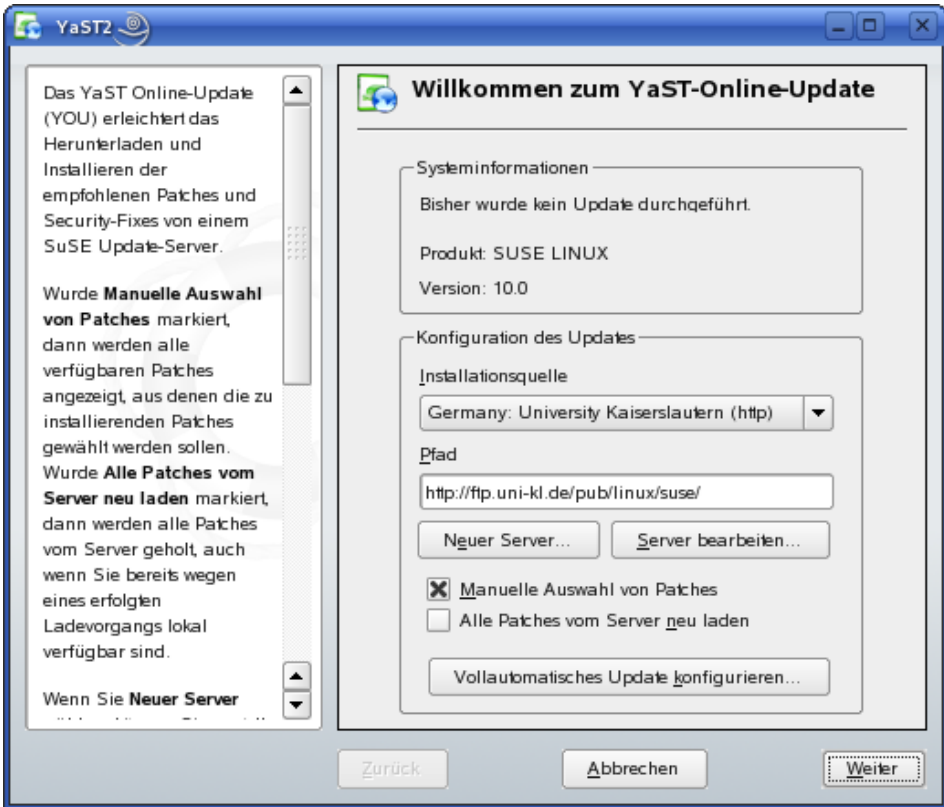


Abbildung 6.1: YaST ermöglicht das Online-Update.

- Startskripten. Viele Linux-Distributionen nutzen SysVinit-Skripten, um die Dienste zu starten. Einige, vor allem kleinere Distributionen nutzen ein einziges Startskript.
- Internet-Super-Server. Der Internet-Super-Server `inetd` oder `xinetd` startet Netzwerkdienste bei Bedarf.
- Cron-Daemon. Der Cron-Daemon startet automatisch Programme zu bestimmten Zeiten.

Die Konfiguration dieser Varianten wird weiter unten erläutert.

Woran erkennen Sie nun einen überflüssigen Dienst? Ich hoffe, dass Sie das bei vielen Diensten selbst erkennen können, da Sie wissen, was der Dienst treibt. Falls das nicht der Fall ist, sollten Sie versuchen, es herauszufinden. Hilfreich sind hier die Manpage, die Dokumentation des Pakets<sup>3</sup> und natürlich Google. Falls Ihnen diese Informationen nicht weiterhelfen, können Sie versuchen, zum Test den Dienst zu

<sup>3</sup>Mit dem Befehl `rpm -qif /etc/init.d/<dienst>` zeigen Sie die Informationen über das RPM-Paket des Dienstes an.

deaktivieren. Falls bei einem Reboot keine Fehlermeldung auftritt und alle wichtigen Funktionen zur Verfügung stehen, war der Dienst nicht erforderlich.

### 6.4.1 Startskripten

Die meisten Distributionen nutzen SysVInit-Startskripten. Diese Skripten befinden sich in dem Verzeichnis `/etc/init.d`. Ob ein Dienst gestartet wird oder nicht, wird über Verknüpfungen in den Verzeichnissen `/etc/rc.d/rc[0-6].d` konfiguriert. Anstatt diese Verknüpfungen jedoch von Hand zu modifizieren, sollten Sie den Befehl `chkconfig` verwenden, der auf den meisten Distributionen vorhanden ist. Falls dieser Befehl bei Ihnen nicht existiert, lesen Sie bitte in der Dokumentation der Distribution nach.

Mit dem Befehl `chkconfig --list postfix` können Sie den aktuellen Zustand der Startskripten anzeigen.

```
# chkconfig --list postfix
postfix      0:off 1:off 2:on  3:on  4:on  5:on  6:off
```

`chkconfig <dienst> off` schaltet den entsprechenden Dienst ab.

Alternativ zu den SysVInit-Skripten gibt es vor allem auf kleinen Distributionen ein einzelnes Startskript, das die Dienste startet. Hier müssen Sie in die Datei schauen und die entsprechenden Zeilen durch Kommentieren deaktivieren. Teilweise (z.B. OpenWRT) existiert auch ein Verzeichnis `/etc/init.d`, und jedes Skript in diesem Verzeichnis wird automatisch aufgerufen. Um einen Dienst zu deaktivieren, genügt es dann meist, die Ausführrechte von der Datei zu entfernen.

#### Achtung



Änderungen in den Startskripten machen sich erst bei einem Neustart oder dem Wechsel des Runlevels bemerkbar.

### 6.4.2 Internet-Super-Server

Fast jede Linux-Distribution verfügt über den Internet-Super-Server `inetd` oder `xinetd`. Wird dieser Dienst über ein Startskript gestartet, so startet er bei Bedarf, entsprechend seiner Konfiguration, weitere Dienste. Das bedeutet, dass Sie den Telnet-Server in der Ausgabe von `ps -ef` nicht sehen, obwohl er verfügbar ist. Er wird erst bei Bedarf von dem `(x)inetd` gestartet.

Der Inetd verfügt über eine Konfigurationsdatei `/etc/inetd.conf`, in der pro Zeile ein Dienst konfiguriert wird. Sie schalten die entsprechenden Dienste durch Auskommentieren der Zeile ab.

Bei dem Xinetd werden die Dienste meist über einzelne Dateien im Verzeichnis `/etc/xinetd.d` gesteuert. Diese Dateien enthalten eine Zeile `enable=` oder `disable=`. Durch Setzen des entsprechenden Wertes (`enable=no` bzw. `disable=yes`) deaktivieren Sie den Dienst.



#### Achtung

Natürlich müssen Sie nach einer Modifikation der Konfiguration den Internet-Super-Server neu starten.

### 6.4.3 Cron-Daemon

Schließlich können Programme auch automatisch von dem Cron-Daemon gestartet werden. Der Cron-Daemon wird über Cron-Tabellen konfiguriert. Auf den meisten Linux-Systemen existieren zwei Arten von Cron-Tabellen.

Jeder Benutzer kann eine persönliche Cron-Tabelle anlegen, wenn ihm dies über die Dateien `/etc/cron.allow` oder `/etc/cron.deny` erlaubt wird. Existiert die erste Datei, so dürfen nur die dort aufgeführten Benutzer eine Cron-Tabelle anlegen. Existiert die zweite Datei, so dürfen alle Benutzer, außer den dort aufgeführten, eine Cron-Tabelle anlegen. Diese Cron-Tabelle wird mit dem Befehl `crontab -e` angelegt und editiert. Mit dem Befehl `crontab -l` zeigen Sie Ihre eigene Cron-Tabelle an, und mit `crontab -r` löschen Sie diese.

Die Syntax dieser Datei ist recht einfach. Es handelt sich um eine Tabelle mit sechs Spalten. Die ersten fünf Spalten enthalten die Definition eines Zeitpunkts: Minuten, Stunden, Tag, Monat und Wochentag. Dabei wird der Wochentag ebenfalls mit einer Ziffer angegeben (0,7 = Sonntag). Der Cron-Daemon vergleicht jede Minute die aktuelle Systemzeit mit der eingetragenen Zeit und führt bei Übereinstimmung den Befehl in der sechsten Spalte aus. Betrachten Sie folgendes Beispiel:

```
0 5 * * * /bin/test.sh
*/30 7-18 * * 1-5 /bin/getmail.sh
```

Das Skript `/bin/test.sh` wird um 5:00 Uhr an jedem Tag (\*), in jedem Monat (\*) und unabhängig vom Wochentag (\*) ausgeführt. Das Skript `/bin/getmail.sh` wird von 7:00 Uhr bis 18:30 alle 30 Minuten (wenn die Division ohne Rest aufgeht) von Montag (1) bis Freitag (5) unabhängig vom Datum ausgeführt.

Zusätzlich zu den Cron-Tabellen der Benutzer existiert auf einem modernen Linux-System auch eine System-Cron-Tabelle `/etc/crontab`. Diese Cron-Tabelle gleicht stark den Tabellen der Benutzer, weist jedoch eine zusätzliche Spalte zwischen der fünften und sechsten Spalte auf. Hier kann der Benutzer angegeben werden, in dessen Kontext der Cron-Daemon den Befehl in der nun siebten Spalte aufrufen soll.

```
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

Die obigen Zeilen stammen von einer Fedora Core Linux-Distribution. Der hier von dem Cron-Daemon aufgerufene Befehl ist `run-parts`. Diesem Befehl werden in Abhängigkeit von der Zeit unterschiedliche Verzeichnisse als Argument übergeben. Der Befehl durchsucht das angegebene Verzeichnis und ruft alle Befehle in diesem Verzeichnis auf. So wird immer eine Minute nach der vollen Stunde jede ausführbare Datei in dem Verzeichnis `/etc/cron.hourly` aufgerufen. Ähnlich wird immer Sonntag (0) um 4:22 das Verzeichnis `/etc/cron.weekly` abgearbeitet.

Um zu erkennen, welche Programme aufgerufen werden, sollten Sie daher die Verzeichnisse analysieren und überflüssige Programme entfernen oder ihnen das Ausführrecht entziehen. Dieses Verhalten ist eigentlich bei allen Linux-Distributionen ähnlich.



### Achtung

Änderungen an den Cron-Tabellen der Benutzer sind sofort aktiv. Ein Neustart des Cron-Daemons ist nur bei einer Modifikation der Datei `/etc/crontab` erforderlich. Auch Änderungen in den Verzeichnissen `/etc/cron.*` erfordern keinen Neustart.

## 6.5 Entfernen überflüssiger Software

Nachdem Sie überflüssige Dienste deaktiviert haben, können Sie nun die entsprechenden Pakete und vielleicht auch noch weitere Pakete entfernen. Das Entfernen von überflüssigen Paketen ist unter Linux seit der Einführung der Paketverwaltungssysteme recht einfach geworden, da diese die Abhängigkeiten überwachen und im Zweifelsfall die Deinstallation verhindern. Beginnen Sie am einfachsten damit, dass Sie alle Pakete auflisten. Dies geht auf einer RPM-basierten Distribution recht einfach mit `rpm -qa`. Auf einer Debian-Distribution zeigt der Befehl `dpkg -l` die installierten Pakete an. Untersuchen Sie die Liste, und versuchen Sie, Pakete zu deinstallieren, die Sie nicht benötigen. Sobald das Paket noch von weiteren Paketen benötigt wird, wird die Paketverwaltung Ihnen einen Abhängigkeitskonflikt anzeigen. Entscheiden Sie dann, ob Sie diese Pakete auch deinstallieren wollen.



### Tipp

Wenn Sie diese Entscheidungen mangels Erfahrung noch nicht treffen können oder wollen, stellen Sie zumindest sicher, dass Sie möglicherweise vorhandene C-Compiler von dem System entfernen. In der Vergangenheit sind bereits mehrfach Würmer aufgetreten, die nach einem Einbruch auf einem System zunächst sich selbst übersetzt haben. Fehlt der C-Compiler, ist die Verbreitung eines solchen Wurms gestoppt.

## 6.6 Sicherheit auf Dateisebene

Die meisten Linux-Distributionen sind nicht speziell auf Firewall-Zwecke ausgerichtet. Ihre Zielsetzung ist eher ein Multifunktionssystem. Es soll möglichst einfach sein, sowohl Netzwerkdienste anzubieten als auch als normaler Benutzer über die grafische Oberfläche eine DVD abzuspielen und zu brennen. Hierfür müssen diese Systeme viele Tricks anwenden, damit das auch so unproblematisch funktioniert, wie es der Anwender möchte. So hat normalerweise ein einfacher Benutzer nicht das Recht, auf das CD- oder DVD-Laufwerk zuzugreifen, und er besitzt dort schon gar keine Schreibrechte. Auch ein einfaches Ping darf der normale Benutzer nicht starten. Diese und viele weitere Aktionen dürfen nur von dem privilegierten Benutzer *root* durchgeführt werden. Damit dennoch ein Benutzer den Befehl benutzen darf, wird entweder der Benutzer für die Ausführung des Befehls mit *root*-Privilegien ausgestattet oder die Rechte der Ressource, auf die der Benutzer zugreifen möchte, werden entsprechend angepasst.

Die grundsätzliche Problematik ist schon sehr alt. Wie soll zum Beispiel ein Benutzer sein Kennwort ändern, wenn er keine Schreibrechte an der Datei */etc/passwd* oder */etc/shadow* hat? Daher wurde sehr früh in der Geschichte von Unix (1971) bereits ein Mechanismus geschaffen, der es erlaubt, für die Ausführung eines Befehls dem Prozess erweiterte Privilegien zu übertragen. Das SetUID- und das SetGID-Recht waren geboren. Verfügt ein Befehl über diese Rechte, so werden für die Ausführung des Befehls die Rechte des Eigentümers (SetUID) oder der Gruppe (SetGID) auf den Prozess übertragen. Sie erkennen diese Rechte an einem kleinen *s*.

```
-r-s--x--x 1 root root 18840 7. Mär 2005 /usr/bin/passwd
-r-xr-sr-x 1 root tty 9752 27. Apr 17:42 /usr/bin/wall
```

Wenn ein Benutzer den Befehl */usr/bin/passwd* aufruft, erbt er für die Ausführung die Rechte des Eigentümers *root*. Bei Aufruf des Befehls */usr/bin/wall* erbt er die Rechte der Gruppe *tty*. Es ist dann Aufgabe der Befehle zu prüfen, dass der Anwender auch tatsächlich nur den vorgesehenen Vorgang durchführt und die Rechte nicht missbraucht. Leider wiesen die Befehle in der Vergangenheit immer wieder Fehler auf, so dass für die Zukunft weitere Fehler nicht ausgeschlossen werden können.

Daher sollten Sie sich fragen, ob auf Ihrer Firewall überhaupt normale unprivilegierte Benutzer arbeiten sollen. Wenn Sie beim Aufruf der Befehle bereits *root* sind

oder die Rechte immer über `sudo` (siehe unten) erhalten, ist es nicht nötig, dass diese Rechte gesetzt sind.

Diese Rechte haben den zusätzlichen Nachteil, dass Sie jedem Benutzer, auch einem Systembenutzer wie *nobody*, für die Ausführung die Privilegien übertragen. Sie sollten daher diese Rechte entfernen. Um das System nach derartigen Befehlen abzusuchen, können Sie die folgenden Befehle verwenden:

```
# find / -perm +4000 -ls
# find / -perm +2000 -ls
```

Sie können die Rechte mit `chmod u-s <datei>` und `chmod g-s <datei>` entfernen. Prüfen Sie auch hier anschließend mit einem Neustart, ob das System noch so arbeitet, wie Sie es sich vorstellen.

### Tipp: Sudo



Sinnvollerweise legen Sie auf der Firewall lediglich für jeden Firewall-Administrator ein Benutzerkonto an, das Sie mit individuellen Kennwörtern versehen. Die Administration des Systems kann dann mit dem Befehl `sudo` erfolgen. Mit diesem Befehl können Sie einem bestimmten Benutzer bei der Ausführung eines bestimmten Befehls erweiterte Privilegien zuweisen. Bei der ersten Ausführung verlangt der Befehl zusätzlich noch die Authentifizierung des Benutzers, die Sie aber auch abschalten können.

Sie konfigurieren den Befehl `sudo` in der Datei `/etc/sudoers`. Diese Datei sollten Sie jedoch nicht direkt, sondern mit dem Befehl `visudo` editieren. Dieser Befehl prüft die Syntax und warnt Sie vor dem Abspeichern bei Fehlern in der Datei.

Um nun eine Gruppe von Administratoren mit *root*-Privilegien auszustatten, können Sie die folgenden Zeilen verwenden:

```
User_Alias ADMIN = spenneb, oliver
ADMIN ALL=(root) ALL
```

Die erste Zeile definiert einen Alias `ADMIN` für die Benutzer *spenneb* und *oliver*. Die zweite Zeile erlaubt diesen Benutzern, auf allen Rechnern als *root* jeden Befehl auszuführen. Die Einschränkung der Rechner ermöglicht es Ihnen, eine zentrale Datei zu erzeugen und für mehrere Systeme zu verwenden und zu verteilen. Dann können Sie hier den Rechnernamen angeben, für den diese Zeile gelten soll.

Die Benutzer *spenneb* und *oliver* benötigen nun nicht mehr das *root*-Kennwort. Sie können selbst jeden Befehl ausführen. Bei der ersten Verwendung und nach Ablauf von 5 Minuten müssen Sie sich mit Ihrem eigenen Kennwort authentifizieren.

Jeder Zugriff wird außerdem protokolliert:

```
Sep 11 15:16:59 bibo sudo: spenneb : TTY=pts/2 ; PWD=/buch/fw_buch/buch ;  
USER=root ; COMMAND=/usr/bin/tail /var/log/messages
```

## 6.7 Sicherheit beim Bootvorgang

Damit Ihre Bemühungen, das System zu sichern, nicht vergebens sind, sollten Sie auch den Bootvorgang des Systems in Ihre Betrachtungen mit einbeziehen.

Zunächst sollten Sie sorgfältig den Aufstellungsort für das System wählen. Achten Sie darauf, dass kein Unbefugter physikalischen Zugang zu dem System erhält. Wählen Sie also nicht die Besenkammer am Ende des Flurs, sondern einen Raum, den Sie abschließen können.

Zusätzlich sollten Sie darauf achten, dass die Bootreihenfolge im BIOS es nicht ermöglicht, das System von Diskette, CD oder USB-Stick zu booten. Sichern Sie das BIOS gegen unbefugte Änderungen mit einem Kennwort.

Achten Sie auch darauf, dass der Boot-Manager mit einem Kennwort gegen unbefugte Änderungen geschützt ist. Erfreulicherweise bieten moderne Distributionen Ihnen dies direkt bei der Installation an. Falls dies nicht der Fall ist, fügen Sie ein Kennwort zur Konfiguration hinzu.

Bei dem Boot-Manager Lilo können Sie einfach zwei Zeilen an den Anfang der Datei `/etc/lilo.conf` anfügen:

```
password=geheimes_Kennwort  
restricted
```

Leider müssen Sie das Kennwort in Klartext angeben. Sie sollten daher darauf achten, dass keiner außer `root` diese Datei lesen darf. Der Parameter `restricted` sorgt dafür, dass lediglich Modifikationen des Bootvorgangs ein Kennwort verlangen. Nach der Modifikation der Datei müssen Sie Lilo neu installieren. Rufen Sie hierzu `lilo -v` auf.

Bei dem Boot-Manager Grub können Sie mit dem Befehl `grub-md5-crypt` ein Kennwort verschlüsseln und in der Grub-Konfigurationsdatei mit dem Parameter `password --md5 $1$. . .` angeben. Anschließend ist für jede Modifikation des Bootverhaltens das Kennwort anzugeben.

Auf Red Hat- und Fedora Core Linux-Distributionen existiert zusätzlich noch die Möglichkeit, während des Startens der Dienste diese interaktiv zu bestätigen. Dazu müssen Sie, während der Init-Prozess startet, `I` eingeben. Sie sollten auch diese Möglichkeit abschalten, damit niemand beim Boot einen Dienst überspringen kann. Setzen Sie hierzu in der Datei `/etc/sysconfig/init` die Variable `Prompt=no`.

Nun sollte keine unbefugte Modifikation des Bootvorgangs mehr möglich sein.

## 6.8 Bastille-Linux

Die von mir bisher beschriebenen Schritte zur Härtung können nur einen allgemeinen Weg aufzeigen. Je nachdem, welche Distribution Sie verwenden, sind vielleicht noch weitergehende Schritte erforderlich, oder Sie können den einen oder anderen Schritt überspringen. Es ist in jedem Fall ein großer Aufwand, alle Schritte richtig und in der richtigen Reihenfolge reproduzierbar und dokumentiert durchzuführen. Bastille-Linux hilft Ihnen dabei.

Bastille-Linux ist ein Härtungswerkzeug für Linux- und Unix-Systeme. Es unterstützt RedHat Linux, RedHat Enterprise Linux, Fedora Core, SUSE, Mandrake, Gentoo, Debian, HP-UX und MacOS X. Sie können mit diesem Werkzeug einfach, reproduzier- und automatisierbar Dienste und Systemeinstellungen konfigurieren. Es schaltet unnötige Dienste ab und kann sogar Dienste in einem Chroot laufen lassen.

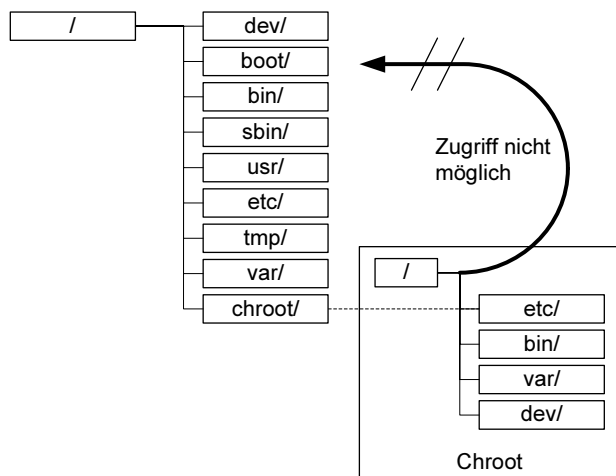


Abbildung 6.2: Nach einem Chroot kann ein Prozess auf Dateien außerhalb des Chroot nicht mehr zugreifen.

### Tipp: Chroot



Ein Chroot stellt eine zusätzliche Sicherheitsfunktion dar. Dabei wechselt ein Prozess beim Start sein Root-Verzeichnis (siehe Abbildung 6.2). Anschließend kann der Prozess auf Dateien außerhalb des eigenen Root-Verzeichnisses nicht mehr zugreifen. Ein Angreifer hat so nach einem Einbruch über diesen Prozess keinen Zugriff auf Dateien des restlichen Systems.

Leider ist die Konfiguration eines derartigen Chroot sehr aufwendig, da alle Dateien, die der Prozess während seiner Ausführung

benötigt, in diesem Chroot-Verzeichnis physikalisch vorhanden sein müssen. Eine symbolische Verknüpfung reicht hier nicht aus. So benötigen viele Dienste Zugriff auf weitere Bibliotheken und Dateien wie `/etc/passwd`, `/etc/resolv.conf`, `/etc/hosts` etc. Die Administration eines Chroot ist daher recht umständlich, aber bei einigen Diensten den Aufwand wert. Dienste, die von Haus aus ein Chroot unterstützen, sind zum Beispiel der Bind-Nameserver und Snort. Weitere Dienste können mit dem Kommandozeilenbefehl `chroot` in einem Chroot-Verzeichnis gestartet werden.

Leider ist ein Chroot auf einem Linux-System nicht so effektiv wie auf einigen anderen Unix-Plattformen. Sobald ein Prozess in dem Chroot-Verzeichnis Privilegien des Benutzers `root` benötigt, besteht die Gefahr, dass ein Angreifer aus dem Chroot-Verzeichnis ausbrechen kann. Ein Chroot schützt nicht vor `root`! Stellen Sie daher sicher, dass alle Prozesse nach dem Wechsel in das Chroot auch ihre Benutzeridentität wechseln.

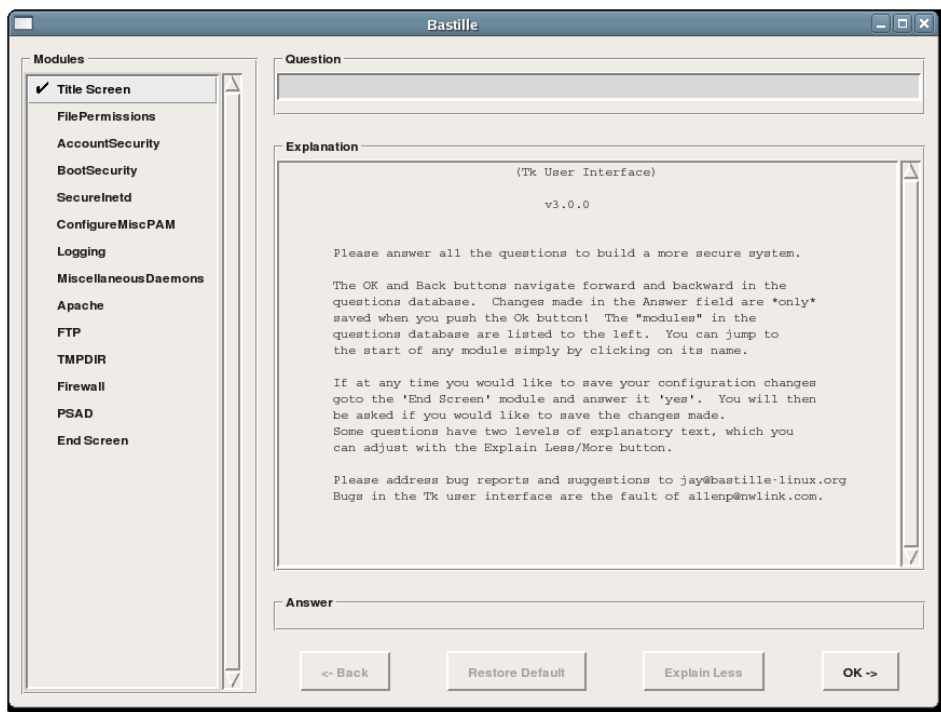


Abbildung 6.3: Die grafische Oberfläche kann mit der Maus bedient werden (`bastille -x`).

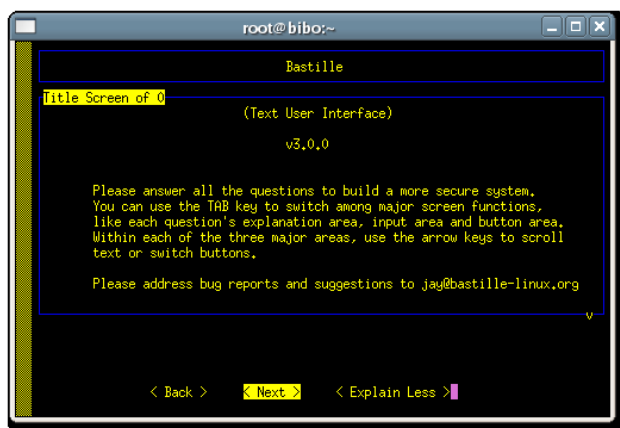


Abbildung 6.4: Firewall-Systeme verfügen meist nicht über eine grafische Oberfläche. Hier können Sie Bastille im Text-Modus einsetzen (`bastille -c`).

Sie können Bastille-Linux in zwei verschiedenen Varianten verwenden: interaktiv und nicht-interaktiv. Im interaktiven Modus können Sie zwischen einer grafischen Perl/Tk- (Abbildung 6.3) und einer textbasierten Perl/Curses-Oberfläche (Abbildung 6.4) wählen. Wenn Sie es interaktiv einsetzen, wird Ihnen Bastille-Linux die verschiedenen Optionen erklären, und Sie entscheiden, ob und wie die Probleme abgestellt werden. Der nicht-interaktive Modus bietet sich für die automatische Vervielfältigung einer Sicherheitsrichtlinie auf vielen Systemen an. Sie stellen so die Nachvollziehbarkeit der Systemhärtung sicher.

Ich werde Ihnen nun kurz eine Führung durch die verschiedenen Funktionen von Bastille-Linux 3.0.X geben.

Bastille-Linux beginnt mit der Härtung der Dateirechte. Zunächst werden die Rechte wichtiger Befehle der Systemadministration so modifiziert, dass ein normaler Benutzer keinen Zugriff erhält (Abbildung 6.5).

Anschließend können Sie entscheiden, ob die Befehle `at`, `ping`, `traceroute`, `mount` etc. über die SetUID-Rechte verfügen sollen oder nicht. Damit Bastille-Linux die Sicherheit erhöht, müssen Sie auf diesen Bildschirmen jeweils `Yes` anwählen.

Im Bereich *AccountSecurity* können Sie wählen, ob die R-Werkzeuge, die eine Klartextanmeldung ohne Kennwort, basierend auf der IP-Adresse, erlauben, deaktiviert werden sollen. Zusätzlich bietet Bastille-Linux die Implementierung von Richtlinien, die das Alter von Kennwörtern überprüfen, das Setzen einer Umask und das Ablehnen des `root`-Logins. Achten Sie darauf, dass Bastille-Linux einige Fragen bereits per Default mit `Yes` beantwortet!

Bei der Betrachtung der *Boot-Security* kann Bastille-Linux den Reboot per `(Ctrl)-[Alt]-[Delete]` abstellen und den Single-User-Modus mit einem Kennwort schützen, falls dies noch nicht der Fall sein sollte.

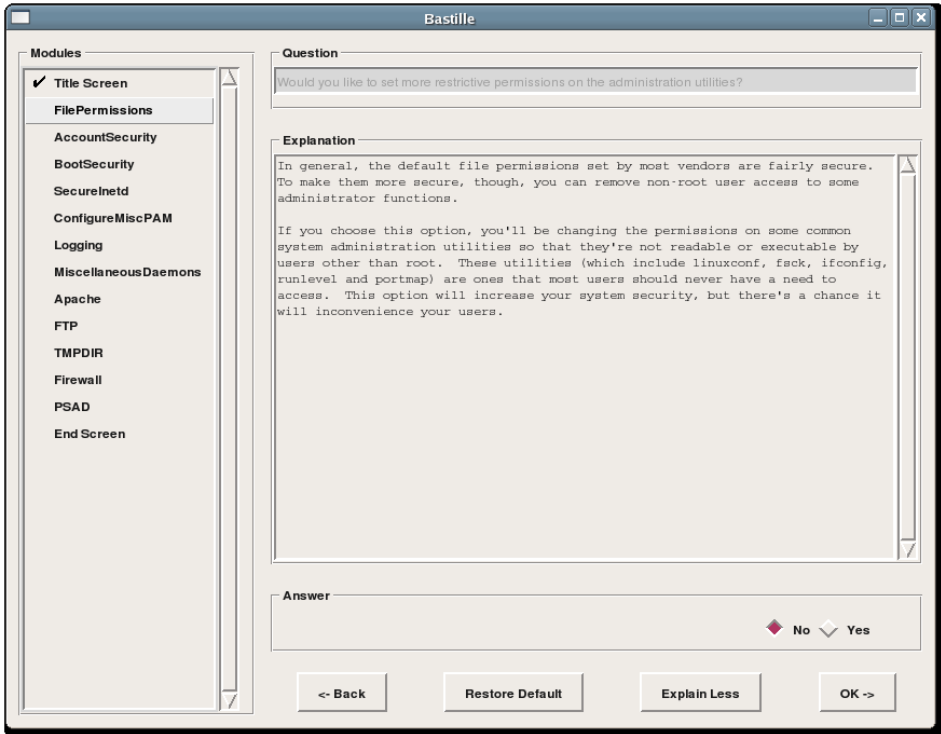


Abbildung 6.5: Bastille entfernt bei Sysadmin-Befehlen die Lese- und Ausführrechte für unprivilegierte Benutzer.

### Tipp



Falls Sie beim Aufruf von Bastille-Linux weniger oder darüber hinausgehende Fragen gestellt bekommen sollten, hängt dies wahrscheinlich mit Ihrer Linux-Distribution zusammen. Bastille-Linux analysiert das System und stellt die Fragen entsprechend.

Bei der Konfiguration des `inetd` bietet Bastille-Linux zunächst die Sicherung der Dienste mit TCP-Wrappers über einen Default-Eintrag in der Datei `/etc/hosts.deny` (siehe Abbildung 6.6) und die Konfiguration von Standard-Bannern für die typischen Dienste Telnet, FTP etc.

Der nächste Bildschirm bietet die Konfiguration der Pluggable Authentication Modules (PAM) an. Hier können Sie zum Beispiel den Konsolenzugriff auf eine bestimmte Liste von Konten beschränken. Alle weiteren Konten dürfen sich nicht auf der Konsole anmelden.

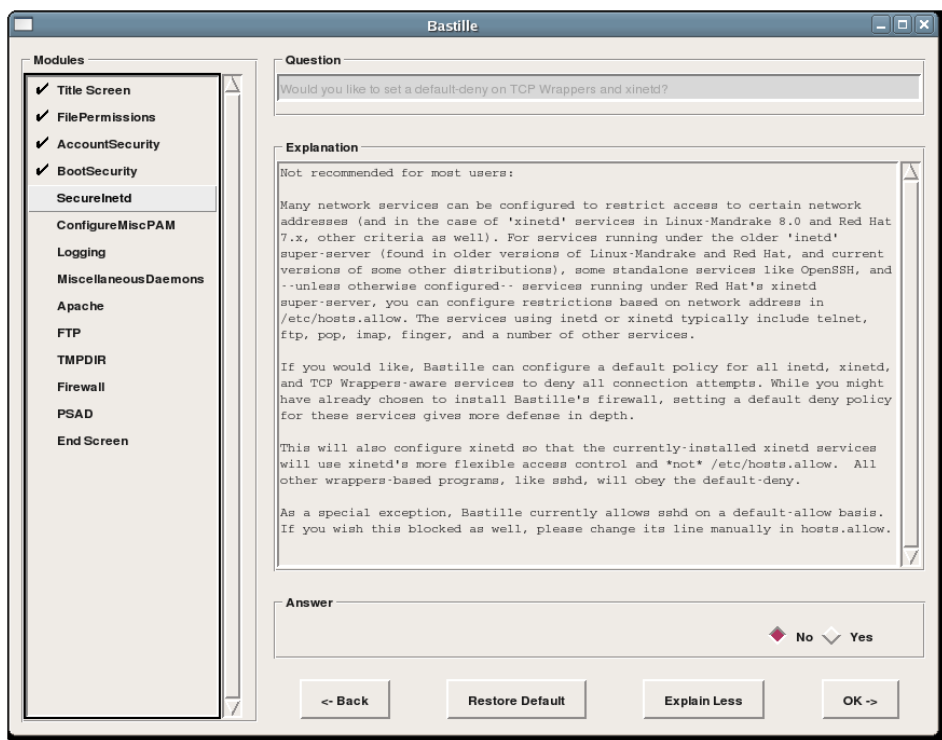


Abbildung 6.6: Viele Netzwerkdienste nutzen TCP-Wrappers als Zugangskontrolle. Bastille-Linux kann den Zugriff auf diese Dienste mit einem Default-Eintrag ähnlich einer Firewall beschränken.

Der nächste Punkt, *Logging*, erlaubt die Aktivierung des Process-Accounting. Hiermit überwacht und protokolliert der Linux-Kernel jeden aufgerufenen Befehl. Achtung, die Protokolle wachsen sehr schnell auf einem normalen System. Für eine Firewall kann dies aber durchaus sinnvoll sein. Denken Sie jedoch daran, dass auch die von Cron aufgerufenen Befehle protokolliert werden.

Im Bereich *Miscellaneous Daemons* können Sie Dienste wie NFS und Samba deaktivieren, die Unterstützung für den HP Office Jet (HPOJ) abschalten und ISDN deaktivieren. Dienste wie Apache und FTP werden gesondert behandelt.

Bei der Konfiguration des Apache können Sie diesen zunächst nur auf dem lokalen Rechner (localhost) anbieten, auf eine spezielle IP-Adresse binden und zum Beispiel die Verwendung von symbolischen Verknüpfungen abschalten.

Die FTP-Server-Konfiguration durch Bastille-Linux kann sowohl den WU-ftpds als auch den heute üblicheren Vsftpd konfigurieren und gegen Angriffe sichern.

Unter dem Punkt *TMPDIR* bietet Bastille-Linux eine sehr interessante Möglichkeit zur Erzeugung temporärer sicherer `/tmp`-Verzeichnisse. Diese werden für jeden Be-

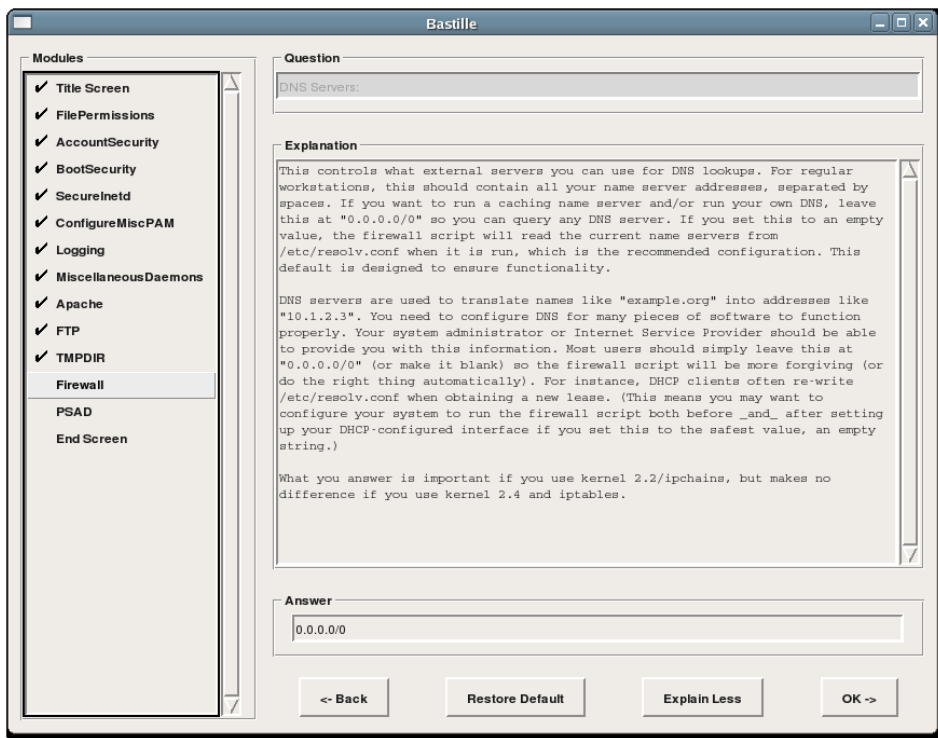


Abbildung 6.7: Bastille-Linux kann sogar nach Beantwortung einiger Fragen ein Firewall-Skript erzeugen.

nutzer automatisch bei seiner Anmeldung erzeugt. So verfügt jeder Benutzer über ein eigenes temporäres Verzeichnis. Viele Angriffe, die dieses Verzeichnis und seine allgemeine Verfügbarkeit ausnutzen, sind so nicht möglich.

Im Grunde versucht Bastille-Linux sogar, dieses Buch zu ersetzen, in dem es die Möglichkeit einer skriptgesteuerten Erzeugung einer Firewall-Konfiguration ermöglicht. Hierzu müssen Sie in mehreren Bildschirmen Fragen zu Ihren genutzten Diensten (z.B. DNS, siehe Abbildung 6.7) beantworten, und Bastille-Linux generiert am Ende ein fertiges Firewall-Skript.

Natürlich ermöglicht es Ihnen Bastille-Linux nicht, sämtliche Feinheiten des Iptables-Befehls auszuschöpfen, jedoch stellt das Skript auch einen guten Start für die Eigenentwicklung dar. Vielleicht finden Sie in diesem Skript einige gute Ideen, die Sie übernehmen wollen.

Als letzten Punkt bietet Bastille-Linux noch die Konfiguration des Port Scan Attack Detector (psad, <http://www.cipherdyne.org/psad/>) an. Dieses Werkzeug analysiert Netfilter-Protokolle und ermittelt mögliche Portscans und Angriffe. Dieses Werkzeug wird auch im Kapitel 12 besprochen. Daher möchte ich für weitere Informationen auf dieses Kapitel verweisen.

Eine besonders nette Funktion von Bastille-Linux ist die Tatsache, dass Sie alle Änderungen wieder rückgängig machen oder auf einer anderen Maschine klonen können. Dies ist möglich, da die interaktive Oberfläche lediglich eine Konfigurationsdatei erzeugt, aber diese noch nicht aktiviert. Diese Aktivierung erfolgt anschließend und kann auch auf einem anderen System durchgeführt werden. Diese Konfigurationsdatei befindet sich anschließend in dem Verzeichnis `/etc/Bastille`. In dieser Datei befinden sich die Fragen und die von Ihnen gegebenen Antworten.

```
# Q: Are you finished making changes to your Bastille configuration?
.End_Screen="Y"
# Q: Would you like to enforce password aging? [Y]
AccountSecurity.passwdage="Y"
# Q: Should Bastille disable clear-text r-protocols that use IP-based authentication? [Y]
AccountSecurity.protectrhost="Y"
# Q: Should we disallow root login on tty's 1-6? [N]
AccountSecurity.rootttylogins="N"
# Q: What umask would you like to set for users on the system? [077]
AccountSecurity.umask="077"
# Q: Do you want to set the default umask? [Y]
AccountSecurity.umaskyn="Y"

... gekürzt ...
```

Bastille-Linux kann Sie so sehr mächtig und leicht reproduzier- und dokumentierbar bei der Härtung Ihrer Linux-Systeme unterstützen. Sicherlich ist die Härtung eines Systems eine Aufgabe, die Sie nicht auf jedem System durchführen werden. Es empfiehlt sich jedoch für alle exponierten Systeme in der DMZ und Firewalls.

## 6.9 Mandatory-Access-Control-Systeme (MAC)

Leider unterstützt ein klassisches Linux-System nur ein sehr einfaches Rechte-Modell. So können der Benutzer `root` und der Eigentümer einer Datei die Rechte dieser Datei modifizieren. Dabei können die Rechte für den Eigentümer, eine Gruppe und alle weiteren Benutzer definiert werden. Mit modernen Dateisystemen können auch die POSIX-ACLs verwendet werden (siehe Tipp). Dann ist es möglich, auch mehreren Benutzern unterschiedliche Rechte zuzuweisen.

Auch beim Einsatz der POSIX-ACLs haben Sie lediglich die Rechte `r`, `w` und `x`. Sie können also lediglich die Rechte für das Lesen, Schreiben und Ausführen verwalten. Weitergehende Rechte existieren nicht.

Wünschenswert wäre die Möglichkeit, die Privilegien des Benutzers `root` einzelnen normalen Benutzern für bestimmte Aufgaben zukommen zu lassen. Obwohl dies seit Jahren mit dem Capability-System möglich ist, existierte bis zur Einführung von SELinux keine sinnvolle Administrationsoberfläche, die zum festen Bestandteil von Linux geworden ist. Alle anderen verschiedenen Ansätze (LIDS, grsecurity, RSBAC etc.) sind nur als Patch verfügbar. Diese Systeme werden häufig auch als

Mandatory-Access-Control-Systeme (MAC) bezeichnet. Dies setzt sie von dem klassischen Unix als Discretionary-Access-Control-System (DAC) ab. Unix ist ein DAC-System, da hier der Eigentümer einer Datei ihre Rechte definiert (die Rechte werden entsprechend der Diskretion des Eigentümers definiert). Der Eigentümer kann hierbei auch leicht Fehler machen. Bei einem MAC-System werden die Rechte zentral für alle Dateien verwaltet. Selbst wenn der Eigentümer *root* fälschlicherweise einem anderen Benutzer Leserecht an der Datei */etc/shadow* zuweisen würde, könnte das MAC-System, das von einem über *root* stehenden Superuser verwaltet wird, diesen Zugriff noch verhindern.

Alle MAC-Systeme verfügen über einen derartigen von *root* unterschiedlichen Superuser. Häufig wird hierfür ein weiteres Kennwort verlangt. Teilweise ist zur Laufzeit keine Modifikation des MAC-Systems möglich.

Alle MAC-Systeme für Linux stellen nur zusätzliche Systeme dar. Das bedeutet, dass beim Zugriff auf eine Datei sowohl das DAC- als auch das MAC-System den Zugriff erlauben müssen.

Derartige Systeme können stark die Sicherheit des Betriebssystems erhöhen. Leider ist eine komplette Besprechung dieser Systeme im Rahmen dieses Buches nicht möglich und sinnvoll. Das Linux Intrusion Detection System (LIDS) wurde aber von mir bereits in dem Buch »Intrusion Detection und Prevention mit Snort 2 & Co.« ausführlich behandelt.

#### Tipp



Verwenden Sie einfach eine Distribution, die bereits ein derartiges MAC-System vorkonfiguriert mitbringt. Aktuell sind das zum Beispiel Fedora Core 3 und 4, RHEL 4 und Gentoo.

#### Tipp: POSIX-ACLs



Die klassischen Unix- und Linux-Dateisysteme erlauben es nur, die Rechte für den Eigentümer, eine Gruppe und den Rest (others) zu verwalten. Eine weitere Abstufung ist nicht möglich. Es kann nicht eine Gruppe mit Leserechten und eine weitere Gruppe mit Schreiberechten ausgestattet werden. Viele andere Betriebssysteme bieten diese Möglichkeit. Für Unix wurden die POSIX-ACLs geschaffen, um dieses Problem zu beheben. Alle aktuellen Linux-Dateisysteme unterstützen inzwischen auch diese erweiterten ACLs.

Um die ACLs zu nutzen, müssen Sie das Dateisystem mit der Option *acl* mounten. Hierzu tragen Sie die Option in der Datei */etc/fstab* in der entsprechenden Spalte ein oder geben sie beim

Mount-Vorgang auf der Kommandozeile ein. Um das Dateisystem `/home` mit ACL-Optionen neu zu mounten, verwenden Sie den folgenden Befehl:

```
mount -o remount,acl /home
```

Nun können Sie für Dateien erweiterte ACLs definieren. Hierfür gibt es die Befehle `setfacl` und `getfacl`. Leider unterstützen die Befehle `ls` und `chmod` die ACLs nicht.

```
# ls -l datei
-rw-r--r-- 1 root root 0 19. Sep 18:17 datei
# setfacl -m u:test:rw datei
# ls -l datei
-rw-rw-r--+ 1 root root 0 19. Sep 18:17 datei
# getfacl datei
# file: datei
# owner: root
# group: root
user::rw-
user:test:rw-
group::r--
mask::rw-
other::r--

# setfacl -m u:test:rwx datei
# setfacl -m m::rx datei
# ls -l datei
-rw-r-xr--+ 1 root root 0 19. Sep 18:17 datei
# getfacl datei
# file: datei
# owner: root
# group: root
user::rw-
user:test:rwx
group::r--
mask::r-x
other::r--
#effective:r-x
```

Sobald Sie eine ACL einer Datei hinzugefügt haben, zeigt der Befehl `ls` bei den Rechten ein `+` an. Dieses zeigt Ihnen, dass weitere ACLs verborgen sind. Außerdem wird an der Stelle, an der üblicherweise die Gruppenrechte angezeigt werden, nun die Maske angezeigt. Diese Maske definiert die maximalen Rechte, die mit den ACLs

vergeben werden können. Alle über die Maske per ACL hinausgehenden Rechte werden automatisch nicht aktiv.

Wenn Sie ACLs einsetzen, sollten Sie darauf achten, dass das Dateisystem immer mit der Option `acl` gemountet wird. Ansonsten sind die ACLs nicht aktiv. Außerdem sollten Sie Ihre Backup-Programme überprüfen. Viele Backup-Programme können nicht mit ACLs umgehen. So sichert `tar` die ACLs nicht! Anstelle von `tar` können Sie aber `star` verwenden.