Linux mit Sicherheit

Linux Professionell 100 Tipps Titelstory

Verfügbarkeit, Vertraulichkeit und Integrität der Daten sind die drei Forderungen der IT-Sicherheit. Diese zu erfüllen, ist kein Problem mit Linux und den folgenden 20 Tipps.

ACHIM WAGENKNECHT

Für die meisten Sicherheitsaufgaben bringt Linux von Haus aus mächtige Werkzeuge mit. Gleich, ob Daten gesichert, Verbindungen verschlüsselt oder Regeln durchgesetzt werden sollen: Mit den folgenden 20 Tipps können Sie Hacker nicht nur abwehren, sondern auch gezielt in die Falle locken. Sie können Ihre Daten nicht nur verschlüsseln, sondern dabei auch noch alle Spuren verwischen. Zudem basiert die sicherste Firewall der Welt auf Linux.

Tipp 01: Käfighaltung

Sicherheitskritische Dienste wie Datei- oder Mail-Server sollten immer in *chroot*-Umgebungen laufen. chroot setzt dem Dienst ein eigenes Startverzeichnis als *root* vor, so dass der Dienst auf das eigentliche Dateisystem keinen Zugriff hat. Wird der Dienst gehackt, kann der Angreifer nur innerhalb des chroot-Käfigs Schaden anrichten. Der Mail-Server Postfix treibt das auf die Spitze, indem er seinen Dienst in mehrere Teilschritte aufteilt, die jeweils von eigenen Dämonen in eigenen chroot-Käfigen ausgeführt werden. Um einen Dienst in einem chroot-Käfig zu starten, stellen Sie dem Startbefehl die chroot-Funktion voraus, etwa mit dem Kommando:

```
chroot /nimm/dies/als/root
    ftpd -o option1
```

Tipp 02: Schnelltest auf Root-Kits

Hacker versuchen in der Regel, ihre Prozesse vor den Blicken eines aufmerksamen Administrators zu verstecken. Mit dem Befehl *ps* sind Prozesse nicht zu sehen, im Systempfad */proc* tauchen diese aber meistens trotzdem auf. Die folgende Befehlszeile zählt die Prozesse, die *ps* anzeigt, und die, die in */proc* gelistet sind:

```
ls -d /proc/* | grep
    [0-9]|wc -l; ps ax | wc -l
```

Weichen beide Zahlen erheblich voneinander ab, ist das ein Indiz für ein Root-Kit: Achtung: In dem Ordner, von dem aus diese Zeile aufgerufen wird, dürfen keine Dateien sein, die Ziffern in ihren Namen enthalten. Sonst wird das Ergebnis verfälscht.

Tipp 03: Passwörter erzwingen

Eine oft vernachlässigte Sicherheitslücke sind Benutzerpasswörter. Viele sind zu kurz, zu leicht zu erraten und werden zu selten gewechselt. Dabei ist es mit aktuellen Distributionen kein Problem, Passwort-Regeln verbindlich durchzusetzen. Unter Suse Linux findet sich die Einstellung im Yast-Kontrollzentrum unter *Sicherheit und Benutzer/Einstellungen zur Sicherheit*. Kreuzen Sie hier an, dass der Rechner *Neue Passwörter überprüfen* und dabei einen *Test auf komplizierte Passwörter* durchführen soll. Die Mindestlänge für ein Passwort sollten Sie auf acht Zeichen setzen, das Höchstalter auf 90 Tage.

Tipp 04: Passwort vom Token

Je strenger Sie Ihre Passwort-Regeln nach Tipp 3 definieren, desto eher werden die Benutzer dazu neigen, ihre Passwörter an unsicheren Stellen zu notieren. Den Ausweg aus dem Dilemma bieten Security-Tokens. Das sind Hardware-Schlüssel, die zeitabhängige Einmalpasswörter erzeugen. Für Linux gut geeignet sind die eTokens von Aladdin. Rat und Infos zu deren Einsatz finden Sie unter *www.etokenonlinux.org*.

Tipp 05: SSL für alle

Ihr bevorzugter News-Server wurde auf SSL-Verschlüsselung umgestellt und KNode kann nicht mehr zugreifen? Oder Sie möchten Ihre eigenen Server mit SSL ausrüsten, aber die eingesetzten Server-Dämonen beherrschen das nicht? Kein Problem mit Stunnel. Der universelle SSL-Wrapper verpackt beliebige TCP-Verbindungen ins Secure Socket Layer. Das Tool kann unter *www.stunnel.org* geladen werden.

Tipp 06: Hochsicherheitstrakt

Die sicherste Firewall der Welt erweitert das Sicherheitskonzept der isolierten Internet-Workstation um Fernzugriff per VNC. Der Aufbau nennt sich *Grafische Firewall* und besteht aus drei Servern: einer äußeren Firewall, einer inneren Firewall und dazwischen ein Internet-Client, der Webseiten per VNC an die Rechner im Hochsicherheitstrakt weiterleitet. Alle drei Server können auf normaler PC-Hardware installiert werden. Auf dem mittleren System wird für jeden Benutzer eine eigene Arbeitsumgebung installiert, auf die er mit einer eigenen Port-Nummer per VNC zugreift.

Tipp 07: VPN schnell und einfach

Virtuelle private Netze auf der Basis von IPsec sind oft knifflig zu installieren, nicht alle Clients vertragen sich mit allen Servern. OpenVPN *(www.openvpn.net)* schafft Abhilfe. Es wird an beiden Endpunkten der Verbindung installiert. Ein einzelnes Semikolon in der Konfigurationsdatei ernennt den einen Rechner zum Server und den an-

deren zum Client. OpenVPN steht auch für Windows zur Verfügung. Es benutzt standardmäßig Port *5000* mit dem Protokoll UDP, der muss also in allen beteiligten Firewalls freigeschaltet werden.

Tipp 08:
Hackquarium

Wer alle Standard-Sicherheitsmaßnahmen umgesetzt hat und noch mehr tun will, kann einen Honeypot aufstellen. Der sieht von außen aus wie ein lohnendes Angriffsziel. Von innen lassen sich aber die Aktivitäten des Hackers im Honeypot isolieren und beobachten wie in einem Aquarium. Schnell und einfach ist so eine Hackerfalle mit Honeywall aufgesetzt. Honeywall ist eine unsichtbare Bridging-Firewall, die zwei Netzwerk-Segmente verbindet. Der Traffic wird analysiert und über eine dritte Netzwerk-Schnittstelle zur Beobachtung weitergeleitet. Zwei alte PCs reichen, um die Falle aufzustellen. Die bootfähige Honeywall-CD kann unter *www.honeynet.org/tools/cdrom/* geladen werden.

Tipp 09:
Würmer angeln

Linux-Admins müssen meist auch Windows-Rechner vor Würmern schützen. Da ist es hilfreich, so schnell wie möglich über neue Wurm-Epidemien informiert zu sein. Nepenthes *(nepenthes.mwcollect.org)* ist ein Daemon, der wurmanfällige Windows-Dienste unter Linux emuliert. Spricht ein Schädling eine dieser Schnittstellen an, wird er heruntergeladen und zwecks Analyse gespeichert. Mit den gefangenen Würmern können Sie eigene Analysen anstellen. Oder testen, wie schnell Ihr Antivirus-Hersteller auf neue Epidemien reagiert.

Tipp 10:
Überwachungs-Pingpong

Zwei Server können sich gegenseitig überwachen, ob sie verfügbar sind. Dazu wird auf beiden Servern ein PHP-Script installiert. Wird ein Script aufgerufen, sendet es zunächst eine Antwort und wartet dann mit der *sleep()*-Funktion das Prüfintervall ab. Dann ruft es wieder sein Partner-Script auf, wartet die Antwort ab und beendet sich. Bleibt die Antwort aus, schlägt das Script Alarm. Achtung: Diese Lösung ist nur für dedizierte Server geeignet. Auf Shared-Hosting-Accounts verbraucht sie zu viel Rechenzeit. Und die Pingpong-Methode hat zunächst einen Haken: Weil immer nur ein Server aktiv ist, bleibt der Alarm aus, wenn ausgerechnet dieser Server ausfällt. Um das zu vermeiden, werden zwei zusätzliche Scripts installiert, die jeweils vor der Sleep-Pause aktiviert werden. Diese Scripts führen einen Prüflauf durch und beenden sich dann, ohne ihrerseits ein Script aufzurufen. Die kompletten Scripts können Sie im Web unter *www.linux-professionell.net* laden.

Tipp 11:
Einzelhaft mit UML

Wem die Prozess-Isolierung mit chroot noch zu durchlässig ist, der kann seine Dämonen stattdessen mit einem eigenen Kernel starten. Zu diesem Zweck gibt es den Kernel User Mode Linux, kurz UML. So bieten auch eventuelle Kernel-Exploits kein Einfallstor mehr, um das Hauptsystem zu kapern. Wenn UML installiert ist, können Sie es aus einem beliebigen Verzeichnis heraus mit dem Befehl *linux* starten. Der Kernel erwartet sein Dateisystem in diesem Verzeichnis in einem Ordner namens *root_fs*. Ein eigenes chroot-Jail als virtuelle Umgebung ist nicht nötig. Nach dem Start erscheint ein Log-in-Prompt und Sie können die Dienste, die Sie mit UML isolieren wollen, einrichten.

Tipp 12:
Unveränderliche Dateien

Die Dateisysteme EXT2 und EXT3 bieten zusätzliche Dateiattribute, mit deren Hilfe Dateien noch besser geschützt werden können. Die Attribute lassen sich mit dem Kommando *lsattr* anzeigen und mit dem Befehl *chattr* ändern. Dateien, die Sie besonders gut schützen wollen, können Sie mit dem Attribut *-i* für immutable versehen. Diese Dateien bleiben dann selbst für *root* unveränderlich, bis das Attribut wieder entfernt wird. Mit dem Attribut *-a* können neue Daten an eine Datei nur angehängt werden. Das versehentliche Überschreiben wird so verhindert. Weitere Attribute enthüllt der Admin mit dem Befehl *chattr*.

Tipp 13:
USB-Stick verschlüsseln

Als Backup-Medien für wichtige Dateien sind USB-Sticks optimal – solange sie nicht gestohlen werden oder verloren gehen. USB-Sticks sollten daher immer verschlüsselt werden. Das lässt sich am besten mit LUKS erledigen, dem Linux Unified Key Setup. LUKS ist in vielen neuen Distributionen enthalten und lässt sich ansonsten unter *luks.endorphin.org* laden. Auf dem USB-Stick wird mit *cryptsetup* ein verschlüsselter Container erzeugt, etwa mit:

```
cryptsetup -c
    aes-cbc-essiv:sha256
    -s 256 luksFormat
    /dev/sda1
```

Im Container kann dann ein Dateisystem angelegt werden. Eine Anleitung finden Sie unter *home.ircnet.de/cru/luks/*.

Tipp 14:
Auto-Backup

Sicherungskopien sollten auf zentralen Servern angelegt werden. Wer die Backups den Benutzern an den Clients überlässt, wird im Ernstfall Schiffbruch erleiden. Besser ist es, einmal wöchentlich ein Komplett-Backup mit *tar* auf den zentralen Server zu spielen und dieses täglich mit *rsync* zu aktualisieren. Es bietet sich an, das rsync-Kommando auf den Clients als Shutdown-Script anzulegen. Die Kommandozeile könnte dann zum Beispiel so aussehen:

```
rsync -az /home/edgar
    backupzentrale:/home/edgar
```

Dabei schaltet die Option *-a* die rekursive Kopie zu Archiv-Zwecken ein und *z* sorgt für Komprimierung.

Tipp 15:
USB abschalten

USB-Schnittstellen sind ein ernsthaftes Sicherheitsrisiko. Wer die USB-Ports nicht gleich ausbauen oder im BIOS deaktivieren

will, kann einen Kernel ohne USB-Module kompilieren. Das könnte sich aber ebenfalls als unpraktisch erweisen. Seit Suse Linux 9.3 ist der Hardware Abstraction Layer HAL für das Einhängen von Wechselmedien zuständig. Wie die Medien zu handhaben sind, steht in Policy-Dateien, die Suse Linux 10.1 im XML-Format unter */usr/share/hal/fdi/policy/* aufbewahrt. Hier können Sie einen Ordner mit dem Namen *95userpolicy* anlegen und dort eigene Policy-Dateien hinterlegen. Hier können Sie USB-Speichermedien mit folgenden Zeilen deaktivieren:

```
<match key="storage.bus"
    string="usb">
  <merge key="storage.policy.
    should_mount" type="bool">
    false
  </merge>
</match>
```

Tipp 16: Rechteverwaltung

Wenn Sie als *root* das Kommando */usr/bin/visudo* aufrufen, bekommen Sie eine Datei zu sehen, die Ausführungsrechte für *sudo* enthält. Mit Hilfe dieser Datei können Sie Admin-Aufgaben komfortabel aufteilen. Die Voreinstellung lautet:

```
root     ALL=(ALL) ALL
```

Das bedeutet, dass der Benutzer *root* auf allen Rechnern als jeder Benutzer jedes Kommando ausführen darf. Das wussten wir schon. Interessant wird es aber, wenn Sie andere Zeilen hinzufügen. So gibt die folgende Zeile etwa dem Benutzer *peter* das Zugriffsrecht auf den Rechner *test.domain.de*:

```
peter test.domain.de=(ALL) ALL
```

Und diese Zeile ermöglicht es der Benutzerin *helga*, auf dem Server *mail.domain.de* als *root* die Befehle */usr/sbin/befehl1* und */usr/sbin/befehl2* zu benutzen:

```
helga mail.domain.de=(root)
    /usr/sbin/befehl1,
    /usr/sbin/befehl2g
```

Tipp 17: Firewall-Regel ändern

Um die Firewall grundlegend zu konfigurieren, ist ein grafisches Werkzeug wie *fwbuilder* das Mittel der Wahl. Aber um auf die Schnelle eine Regel zu ändern, ist es oft praktischer, IPtables von der Kommandozeile zu starten. Die folgende Zeile etwa erlaubt den Zugriff von einer bestimmten IP-Adresse:

```
iptables -A INPUT -t filter -s
    62.140.213.91 -j ACCEPT
```

Soll ein bestimmter Port erlaubt werden, zum Beispiel Port *80* für HTTP, dann sieht die Zeile so aus:

```
iptables -A INPUT -t filter -p
    tcp --dport 80 -j ACCEPT
```

Tipp 18: Dienste abschalten

Um mögliche Angriffsziele zu minimieren, sollte der Admin grundsätzlich alle nicht benötigten Dienste abschalten, etwa *echo*, *charges*, *discard*, *daytime*, *time*, *talk*, *ntalk* sowie die als extrem unsicher geltenden Kommandos *rsh*, *rlogin* und *rcp*. Für Letztere gibt es die sicheren Alternativen *ssh* und *scp*. Nachdem die nicht benötigten Dienste deaktiviert sind, sollte der Systemverwalter prüfen, ob der Dienst *inetd* überhaupt noch benötigt wird – Dienste können alternativ auch als Daemon gestartet werden. Wenn ja, gibt es Alternativen zu inetd, die vielfältiger konfiguriert werden können, beispielsweise *xinetd* oder *rlinetd*.

Tipp 19: Dateien überwachen

Eine wichtige Aufgabe des Systemadministrators ist die regelmäßige Überwachung aller Dateien im System. Angreifer können etwa durch das Verändern von Systemdateien oder von Programmdateien die Kontrolle über einen Rechner erhalten. Das Auffinden dieser Veränderungen ist nur möglich, wenn der Stand vor der Veränderung bekannt ist. Debian-basierte Linux-Distributionen überprüfen mithilfe des ausgeklügelten Paketsystems die installierten Dateien. Das Tool *cruft* untersucht das komplette Dateisystem nach Dateien, die im Grunde nicht vorhanden sein sollten, beziehungsweise nach Dateien, die sich nicht mehr im Dateisystem finden lassen. Hierzu wird im Wesentlichen auf die Informationen aus den Dateien im Verzeichnis */var/lib/dpkg/info/* zugegriffen. Cruft überwacht zusätzlich die *alternatives*-Informationen, die *lost+found*-Verzeichnisse in einem EXT2-Dateisystem sowie die Home-Verzeichnisse.

Tipp 20: Admin per SMS alarmieren

Wenn ein Überwachungs-Programm abends oder am Wochenende Alarm schlägt, sollte es das am besten per SMS (Short Message Service) tun und eine Nachricht auf das Mobiltelefon des Admins schicken. Damit das funktioniert, wird ein Account bei einem SMS-Provider benötigt. Nachfolgend finden Sie einige kommerzielle Anbieter für den SMS-Versand:

- *www.absolutsms.de*
- *www.comunified.com*
- *www.grouptime.de*
- *www.ina-germany.de*
- *www.mobilant.com*
- *www.topconcepts.de*

Je nach Größe des eingekauften SMS-Kontingents bezahlen Unternehmen zwischen 5 und 10 Cent pro verschickter Nachricht. Alle genannten Anbieter unterhalten ein Gateway, das aus Scripts heraus per HTTP-Request angesprochen werden kann. Soll ein Shell-Script die Alarmierungs-SMS verschicken, kann der Kommandozeilen-Befehl *smssend* verwendet werden. Alternativ können Sie den Versand auch in ein PHP-Script einbinden und wie folgt steuern:

```
$url = "http://www.comunified.
   com/gateway/sendsms.php";
$user = "xxxxxxx";
$pass = "xxxxxxx";
$ergebnisseite = fopen($url.
'?user= '.$user.
'&password='.$pass.
'&message='.$message.
'&recipient='.$recipient, 'r')
or die($php_errormsg);
```