

# Linux-Sicherheit

Daniel Rohde | Computer- und Medienservice, Systemsoftware und Kommunikation | d.rohde@cms.hu-berlin.de

Dieser Artikel wendet sich hauptsächlich an Benutzer, die Linux auf dem Desktop einsetzen (möchten) und weniger an Linux-Administratoren. Letztere – die gerade mit Linux auf Servern anfangen – bekommen hier aber auch einen Vorgeschmack auf die möglichen Sicherheitsprobleme von Linux. Denn wer glaubt, dass man mit Linux auf der „sicheren Seite“ ist, der irrt. Einem aktuellen Blog-Eintrag des Antivirensoftware-Herstellers Sophos [1] zufolge, sind gerade Linux-Systeme besonders beliebt bei Kriminellen als Ausgangsbasis für den Aufbau von Bot-Netzen, da die meisten Linux-Systeme als Server rund um die Uhr in Betrieb sind. Das wiederum heißt, dass sich kriminelle Hacker oder wohl eher Cracker immer öfter die Mühe machen, Sicherheitsschwächen von Linux ausfindig zu machen und letztlich auszunutzen.

Neben den Problemen mit Kriminellen, Viren und Bot-Netzen muss man sich auch um die Datensicherheit Gedanken machen, vor allem dann, wenn man Linux auf dem Notebook einsetzen möchte. Wie schnell passiert es, dass es verloren geht oder gestohlen wird. Wenn dann persönliche oder sehr vertrauliche Daten wie Bankverbindungsdaten oder Zugangskennungen unverschlüsselt auf der Festplatte herumliegen, kann ein Missbrauch nicht ausgeschlossen werden – wie sagt man so schön: „Gelegenheit macht Diebe“. Außerdem ist man mit einem Notebook dank Wireless-LAN auch schnell mal online. Probleme mit unsicheren WLAN-Protokollen hat man ja nicht nur unter Windows oder MacOS X.

Haupteinfallstore auf Linux-Systemen sind Sicherheitslücken (Programmierfehler) oder schlechte Konfigurationen von Server-Anwendungen, die übers Netz erreichbar sind. Webserver, Datenbank-Management-Systeme oder einfach nur SSH-Zugänge werden gern von Kriminellen oder Würmern genutzt. Ich will an dieser Stelle nicht weiter auf Angriffsarten wie SQL-Injection oder Cross-Site Scripting (XSS) eingehen. Dabei spielen aber oft nicht nur Sicherheitslücken eine große Rolle, vor allem unsichere Passwörter sind ein echtes Übel. Selbst wenn das root-Passwort besonders gut ist, kann ein unsicheres Benutzer-Passwort schon reichen, um ein Linux-System bloßzustellen.

Daneben kann man natürlich auch durch das eigene Verhalten Einfluss auf die Linux-Sicherheit nehmen. Wer über unverschlüsselte Netzwerkverbindungen Passwörter versendet (telnet, rsh, E-Mail: pop, imap, smtp ohne SSL/TLS ...), Software aus nicht vertrauenswürdigen Quellen installiert, nur selten Updates einspielt, ständig als root auf dem System arbeitet und/oder keine Firewall verwendet, darf sich nicht wundern, wenn der eigene Linux-Rechner zur Wurm-Schleuder mutiert oder für kriminelle Machenschaften missbraucht wird.

Unter Linux gibt es eine ganze Reihe von Möglichkeiten, Probleme zu erkennen und zu vermeiden. Ein System richtig zu „härten“, ist schon in wenigen Schritten möglich:

- regelmäßig (Sicherheits-)Updates einspielen
- sichere Passwörter für root und Benutzern setzen und bei Servern mit Benutzerzugang entsprechende Passwort-Policies aktivieren

*Kein Betriebssystem ist wirklich „sicher“. Dieser Beitrag soll weniger Linux-Administratoren als Linux-Benutzern und solchen, die es werden wollen, einen kleinen Überblick darüber geben, welche Sicherheitsprobleme Linux mit sich bringt und wie es sich ein klein wenig sicherer machen lässt.*

- alle Dienste, die Ports öffnen und nicht benötigt werden, beenden und am besten deinstallieren, alle benötigten Dienste absichern und regelmäßig prüfen
- Firewall, falls nicht schon durch die Linux-Distribution aktiviert, installieren
- Vor allem Laptop-Benutzer sollten ihre vertraulichen Daten in Crypto-Filesystemen oder in sicheren Datenspeicher-Programmen ablegen.
- Regelmäßige Viren-Scans sind mittlerweile ebenfalls nötig.

Die wichtigsten Linux-Distributionen (ich meine damit Ubuntu/Debian, Novells SuSe und Redhat) bringen alle einfache Werkzeuge mit, um Updates einzuspielen. Wer es geschafft hat, Linux zu installieren, kennt auch die Werkzeuge, wer sie nicht kennt, schaut auf die Support-Webseiten seines Distributors.

Wer ein Problem damit hat, sich ein sicheres Passwort auszudenken, kann auch auf einen der zahllos vorhandenen Passwort-Generatoren zurückgreifen, beispielsweise liefert „pwgen -sy 16“ recht brauchbare root-Passwörter. Für Benutzer reicht aber auch schon „pwgen -y 8“, um sich ein „merkbares“ Passwort zu generieren. Das pwgen-Programm steht zumindest bei Debian-basierten Distributionen (Ubuntu, Knoppix, ...) als installierbares Paket bereit. Wie man Passwort-Policies ändert, ist aber eher für Server-Administratoren interessant (via PAM realisierbar, siehe auch <http://tldp.org/HOWTO/Security-HOWTO/>).

Um offene Ports und die entsprechenden Dienste auffindig zu machen, gibt es gleich mehrere Möglichkeiten. Die Kommandos „netstat -anp | grep -iw listen“ und „lsof -i | grep -iw listen“ - beide als root-User ausgeführt - liefern eine Liste von offenen Ports samt der dazugehörigen Programme („listen“ muss bei einigen Linux-Distributionen mit deutscher Lokalisation aber durch „abhören“ ersetzt werden). Natürlich kann man ein Linux-System auch mit Hilfe von Port-Scannern analysieren. Wer sich also mit „nmap“ oder „nessus“ herumschlagen möchte, muss aber wissen, dass die neuere deutsche Rechtsprechung ein Problem mit solchen „Hacker“-Werkzeugen hat – aber „wo kein Richter, da kein Henker“. Um offene Ports „dicht“ zu bekommen, gibt

es wiederum mehrere Möglichkeiten:

- Dienste deinstallieren oder deaktivieren
- Dienste nur auf das localhost-Interface (127.0.0.1) binden lassen
- Einige Dienste wurden evtl. mit dem TCP-Wrapper übersetzt und lassen sich dann via hosts.allow/hosts.deny absichern (z.B. sshd).
- Firewall einschalten

An dieser Stelle zu erklären, wie man das alles machen kann, würde den Rahmen dieses Artikels sprengen. Für vieles davon gibt es unter Linux Dokumentationen (man, info bzw. HOWTOs: <http://www.linuxhaven.de/dlhp/>) oder Foren im Web.[2]

Einige Linux-Distributionen installieren schon bei der Grund-Konfiguration eine Firewall (z. B. (Open)SuSe: via yast zu konfigurieren). Andere Distributionen liefern zum Teil eine ganze Liste von Hilfs-Programmen und -Skripten aus, um eine Firewall einzurichten. Für Debian- und Ubuntu-Benutzer, die sich nicht mit den Details von Linux-IP-Filter-Regeln herumschlagen wollen, empfiehlt sich „firestarter“, das eine einfach zu bedienende GTK-Oberfläche mitbringt. Wer nicht so genau weiß, ob schon eine Firewall läuft, kann in einer root-Shell einfach mal „iptables -L“ eintippen. Wenn dabei eine lange Liste herauskommt, deren Zeilen beispielsweise mit „ACCEPT“, „DROP“, „REJECT“ oder „LOG“ anfangen, der hat eine aktive Firewall.

Ja, auch unter Linux gibt es Viren, Würmer und Backdoors. Allesamt lassen sie sich auch nur mit entsprechenden Scannern erkennen und manchmal sogar entfernen. Hier mal eine Liste von frei verfügbaren Virensclannern für Linux:

- Avira AntiVir PersonalEdition Classic
- AVG Anti-Virus Free Edition for Linux
- avast! Linux Home Edition
- Clam AntiVirus
- F-PROT Antivirus for Linux Workstations

Für den nichtkommerziellen Clam Anti-Virus liefern die meisten Linux-Distributionen fertige Pakete einschließlich komfortabler Bedienoberflächen (z. B. ClamTK, AVScan, KlamAV). Für Humboldtianer, und das gilt sowohl für Mitarbeiter als auch für Studierende, steht die kommerzielle Kommandozeilenversion

von McAfee auf den „Viren“-Webseiten des CMS zum Download bereit. Man darf sie sogar, dank der Campus-Lizenz, auch auf dem heimischen PC oder Laptop installieren. Dem Installationspaket liegt eine readme.txt für die Installation und eine komplette Doku in PDF (e520oupg.pdf) bei, die man aber auch benötigt, wenn man wissen will, wie man damit richtig scannt und vor allem, wie man die Viren-Signaturen aktualisiert.

Unter Linux lassen sich natürlich auch sämtliche Dateien verschlüsselt ablegen. Selbst das Wurzel-Dateisystem lässt sich verschlüsseln [3].

Dank der Loopback-Geräte kann man auch Dateien, die ein komplettes verschlüsseltes Filesystem enthalten, anlegen und mounten[4]. Einigen Distributionen liegt auch „encfs“, ein verschlüsseltes Dateisystem im Userspace, als Paket bei: <http://www.argo.net/encfs.5>

Wer sich mit der Ablage von Passwörtern oder Ähnlichem begnügt, kann Programme wie kwalletmanager, mypasswordsafe oder gringotts nutzen.

Natürlich kann man noch viel mehr tun. Einige HOWTOs findet man zum Beispiel auf tldb.org, einschließlich einer (etwas veralteten) Linux Security Quick Reference Card (<http://tldp.org/docs.html>), die auch beschreibt, wie man Veränderungen im Dateisystem erkennen kann (Tripwire), wie und wo man das System-Logging einstellt oder wie man „sudo“ konfiguriert, um nur das Nötigste mit root-Rechten ausführen zu können.

## Literatur und Links

- [1] BACHFELD, D.: *Sophos: Linux spielt wichtige Rolle in Botnetzen*. <http://www.heise.de/newsticker/meldung/103563/>, 2008
- [2] *Deutsches Linux HOWTO Projekt*. <http://www.linuxhaven.de/dlhp/>, 2008
- [3] PETULLO, M.: *Encrypt Your Root Filesystem*. <http://www.linuxjournal.com/article/7743>, 2004
- [4] MUTZ, M.: *Linux Encryption Howto*. <http://encryptionhowto.sourceforge.net/Encryption-HOWTO-4.html>, 2000
- [5] *EncFS Encrypted Filesystem*. <http://www.argo.net/encfs>, 2008