Aus dem Nähkästchen geplaudert: Netzwerkdienste finden und abschalten

Quasselpause

Unix-Systeme warten mit unzähligen Funktionen und Diensten auf. Doch nicht jeder Administrator weiß, was sich im Einzelnen auf seiner Maschine herumtreibt. Dieser Workshop leistet Hilfestellung beim gezielten Abschalten unnötiger Kommunikationsdienste. Marc André Selig

Früher war der Hersteller eines verbreiteten Fenstersystems berühmt-berüchtigt für seine freizügigen Voreinstellungen. Jeder Anwender sollte mit seinem Computer sofort losarbeiten können, egal wie. Weil er wie die meisten Hersteller nicht so genau weiß, was der Benutzer will, aktivierte er vorsichtshalber alle möglichen Netzwerkdienste. So traten in der Hinsicht kaum Probleme auf: Was der Nutzer versuchte, funktionierte auf Anhieb – abgesehen von fehlerhaften Diensten und Funktionen, doch das ist eine andere Geschichte.

Das Problem an dieser Haltung ist freilich, dass sie böswilligen Zeitgenossen relativ freien Zugriff auf den Computer gibt. Es ist zwar bequem, wenn jeder Dienst per Default über das Netzwerk zugänglich ist, allerdings profitieren davon auch die bösen Geister. Finden diese einen ausnutzbaren Fehler, dann hat der reguläre Nutzer ein Problem.

Viele Programme, die Netzwerkdienste bereitstellen, sind jedoch auf den meisten vernetzten Computern verzichtbar. Sicherheitshalber sollten sie also ausgeschaltet bleiben. Beim Fensterln erledigen das – hoffentlich – die Sicherheits-Updates. Unter Unix sind Sie als Administrator gefragt, denn leider schalten nicht alle Distributionen alle Dienste per Default aus.

Prozesse anzeigen

Zunächst sollten Sie ermitteln, welche Dienste auf Ihrem Computer laufen, am einfachsten per »ps« (Process Status). Die ausführliche Version »ps ax« (BSD-Stil) oder »ps -ef« (Posix-Stil) erzeugt eine handliche Liste aller auf dem Unix-Computer laufenden Prozesse (Abbildung 1).

Im Idealfall finden Sie hier keinen Prozess, den Sie nicht kennen und von dem Sie nicht sicher wissen, dass er auf diesem System wirklich erforderlich ist. Allerdings beherbergt selbst ein fast arbeitsloser Rechner mit nur einem Benutzer bereits knapp 100 Prozesse. Hier kann nur ein sehr erfahrener Admin wirklich alle Aktivitäten einordnen. Es gilt also zu differenzieren.

Offene Türen finden

Das größte Gefahrenpotenzial geht zweifellos von Diensten aus, die Netzwerkverbindungen anbieten oder aufbauen. Selbstverständlich bietet Unix einen Befehl, um diese anzuzeigen: »netstat -a« meldet alle aktiven oder wartenden Sockets, also alle Verbindungen, über die laufende Programme mit der Außenwelt kommunizieren.

Unter Linux grenzen Sie die gewünschten Sockets mit »netstat -tuan« ein (Abbildung 2). Die zusätzlichen Optionen reduzieren die Ausgabe auf TCP- und UDP-Sockets, unterbinden also die Unix-Domain-Sockets, die nur für den lokalen Datenaustausch zwischen Programmen auf demselben Computer dienen. Die Option »-n« verhindert die Namensauflösung und spart damit kostbare Zeit. Allerdings sehen Sie nur noch die IP-Adresse der Kommunikationspartner, nicht deren DNS-Namen.

Die Ausgabe enthält für jeden Socket das Protokoll (TCP oder UDP) sowie die beiden Endpunkte und den Zustand der

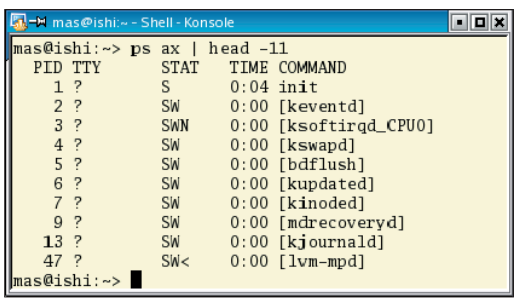

Abbildung 1: »ps« zeigt alle laufenden Prozesse an. Das angehängte »head -11« beschränkt die Ausgabe auf die ersten 11 Zeilen.

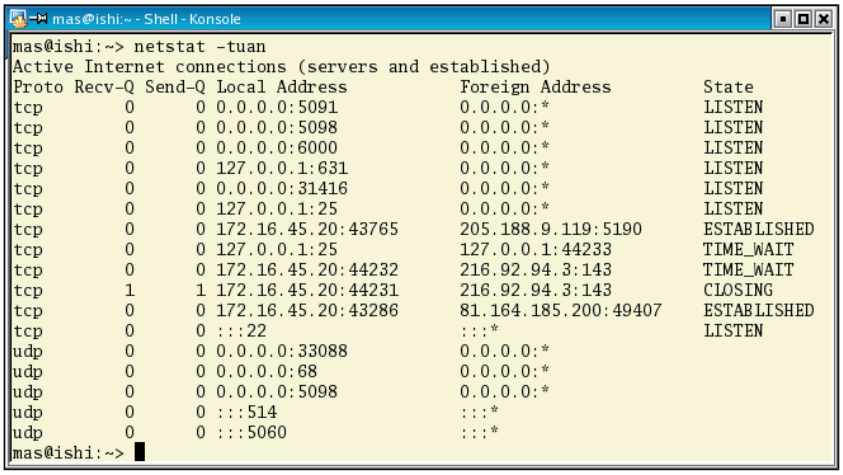

Abbildung 2: Der Aufruf »netstat -tuan« zeigt alle TCP- und UDP-Sockets, die die Prozesse der lokalen Maschine zum Datenaustausch mit der Außenwelt verwenden.

Verbindung. Der Zustand fehlt nur bei den verbindungslosen UDP-Sockets. Ein aktiver TCP-Socket (Zustand »ESTABLISHED«) trägt eine Verbindung zwischen zwei definierten Ports zweier IP-Adressen. Zustände wie »TIME_WAIT« oder »CLOSING« kennzeichnen Sockets, die gerade abgebaut werden oder schon geschlossen sind. Ein UDP-Socket kennt keine Verbindungen und damit auch keinen Zustand. Er nimmt einfach Pakete entgegen, sobald welche ankommen.

Vorsicht Server

Ein Alarmsignal für Sie sind alle Sockets im Zustand »LISTEN« sowie die UDP-Sockets. Hier handelt es sich um Server, die Daten von fremden Systemen annehmen. Bei diesen Sockets lohnt die genaue Betrachtung der lokalen Adresse mit IP- und Portnummer.

Die Portnummer eines Servers gibt häufig ersten Aufschluss über das Programm und die Funktion, die sich hinter dem Socket verbirgt. So ist ein TCP-Socket mit der Portnummer 25 meist einem Mailserver zugeordnet, ein UDP-Socket auf Port 68 deutet auf einen DHCP-Client hin. Nachschlagen können Sie die verbreiteten Zuordnungen in der Datei »/etc/services«. Wenn Sie wissen wollen, welcher TCP-Service auf Port 6000 lauscht, dann fördert »grep 6000/tcp /etc/services« die Information ans Licht:

```
x11     6000/tcp    # X Window System
quake   26000/tcp   # quake
```

Die IP-Adresse eines »LISTEN«-Sockets verrät, ob er Daten tatsächlich aus dem ganzen Internet entgegennimmt, wie in Abbildung 2 beispielsweise der ICQ-Server auf 0.0.0.0:5091 (IP-Adresse 0.0.0.0, Portnummer 5091). Hingegen lauscht der Socket des Mailservers auf 127.0.0.1:25. Die IP-Adresse 127.0.0.1 ist dem Loopback-Interface zugeordnet. Hier besteht kein Grund zur Besorgnis: Dieser SMTP-Socket (Simple Mail Transfer Protocol) nimmt nur Verbindungen von Loopback entgegen und dieses Interface ist nur mit dem lokalen Rechner verbunden.

Sehr aufschlussreich ist die zusätzliche Option »-p«, die Sie vor allem als Root sinnvoll nutzen können (Abbildung 3). Mit dieser Option bemüht sich Netstat darum, den Prozess herauszufinden, der einen Socket belegt hält. So können Sie Dienste identifizieren, die Ihnen auf Anhieb spanisch vorkommen.

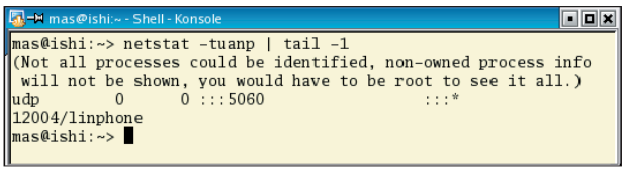

Abbildung 3: »netstat -tuanp« zeigt zusätzlich zur Darstellung in Abbildung 2 den Prozess, der einem Socket zugeordnet ist. Hier lauscht Linphone auf UDP-Port 5060, seine Prozess-ID lautet 12004.

Selbst wenn ein Socket sperrangelweit offen steht, ist er nicht unbedingt auch aus dem Internet erreichbar. Dazwischengeschaltete Paketfilter, besonders solche mit NAT-Funktionalität, können Verbindungsversuche zuverlässig verhindern, egal ob sie gut oder böse gemeint sind.

Jedes Linux-System enthält einen eingebauten Paketfilter. Seit den späten Kernelversionen 2.3 heißt er Netfilter [1] und wird per »iptables«-Kommando gesteuert. Viele Distributionen aktivieren eine Basiskonfiguration von Netfilter. Ob diese sinnvolle Regeln enthält, steht jedoch auf einem anderen Blatt. Die aktuell gültigen Regeln für den Paketfilter zeigt Ihnen der Befehl »iptables -L -n -v«. Die Ausgabe von »iptables« ist ein wenig kryptisch; ein künftiger Artikel wird beim Entschlüsseln helfen.

Türen schließen

Wenn »netstat« und »ps« Dienste identifizieren, die auf diesem Computer nichts verloren haben, sollten Sie als gewissenhafter Admin die zugehörigen Programme deaktivieren. Verfügt ein Dienst über einen eigenen Daemon, müssen Sie dazu meist ein Init-Skript in »/etc/rc*.d/« deaktivieren [2]. Manche Dienste werden aber erst bei Bedarf über einen zentralen Steuer-Daemon gestartet. In diesem Fall modifizieren Sie die Konfiguration in »/etc/inetd.conf« oder im Verzeichnis »/etc/xinetd.d« [3].

Wenn Sie Ihr System für abgesichert halten, sollten Sie einen rudimentären Test dieser Annahme durchführen. Ideal eignet sich dafür ein Port-Scanner wie Nmap [4], der das System von außen auf offene Ports abklopft. Eine Warnung zum Schluss: Nicht alle Dienste sollten Sie ungefragt ausschalten. Einige grundlegende Daemons sind für das einwandfreie Funktionieren eines Unix-Systems unabdingbar. Ziehen Sie im Zweifelsfall ein frisch aufgesetztes System zum Vergleich heran. *(fjl)*

Infos

[1] Netfilter: [http://www.netfilter.org]
[2] Marc André Selig, „Startaufstellung – Der Sys-V-Init-Prozess“: Linux-Magazin 06/04, S. 78
[3] Marc André Selig, „Finger-Server einrichten und nutzen“: Linux-Magazin 11/03, S. 58
[4] Nmap: [http://www.insecure.org/nmap/]