Der eigene Server mit FreeBSD 9

Konfiguration, Sicherheit und Pflege

von
Benedikt Nießen

1. Auflage

dpunkt.verlag 2012

Verlag C.H. Beck im Internet:
www.beck.de
ISBN 978 3 89864 814 1

1 Eine Einführung

1.1 Was bedeutet es, Administrator zu sein?

Als »Administrator« (auch »Superuser«) wird im Bereich der Systemverwaltung derjenige Benutzer bezeichnet, der für die Wartung und Pflege des Netzwerks und seiner Komponenten verantwortlich ist.

Aufgrund seiner Funktion und der damit verbundenen Aufgaben hat er unbeschränkten Zugriff auf sämtliche Systeme und trägt daher eine besondere Verantwortung, die leider zu oft unterschätzt wird.

In diesem Buch befassen wir uns mit der Betreuung lediglich eines Systems, Du wirst aber schnell feststellen, dass mit der Anzahl der darauf laufenden Dienste (der sog. Daemons) der Pflegeaufwand drastisch ansteigt.

In vielen Internetforen, die sich mit der Betreuung von Servern befassen, existieren zahlreiche Themen, in denen Anfänger banale Fragen stellen, die deutlich machen, dass sie sich mit der Materie nicht oder nicht ausreichend auseinandergesetzt haben, aber bereits »am offenen Herzen operieren«.

Diese Leichtsinnigkeit ist nicht nur ein Risiko für sämtliche Systeme, die sich im gleichen Netzwerk (zum Beispiel dem Internet) befinden, sondern kann sich auch schnell als finanzielles Fiasko für den Eigentümer entpuppen, was viele allerdings nicht wahrhaben wollen.

Wurde ein Server erst einmal kompromittiert (und im günstigsten Fall rechtzeitig vom Provider gesperrt), ist das Gejammer meist groß. Die Kosten, um das System wieder in einen funktionsfähigen Zustand zu versetzen, ohne dabei einen Datenverlust zu erleiden, sind dabei nicht zu unterschätzen, selbst dann, wenn eine Datensicherung existiert.

Bevor wir uns nun näher mit FreeBSD und dem spannenden Thema Systemwartung beschäftigen, möchte ich noch eine Ermahnung an Dich richten:

Ratschlag: Falls Du nicht bereit bist, einen (Groß-)Teil Deiner Freizeit mit Lernen und regelmäßiger Systempflege zu verbringen, solltest Du Dich von dem Gedanken verabschieden einen eigenen Server zu betreiben.

Stattdessen empfiehlt es sich in diesem Fall, einen Dienstleister zu finden, der diese Arbeit für Dich übernimmt, oder Deine Anforderungen zu überdenken und gegebenenfalls zurückzuschrauben.

Admins haften: http://serverzeit.de/FreeBSD/admins-haften/

Root und kein Plan: http://www.root-und-kein-plan.ath.cx/

1.2 Warum FreeBSD?

FreeBSD gehört zu einem der besten und zuverlässigsten Serverbetriebssysteme, was durch Statistiken zu Webseiten mit den niedrigsten Ausfallzeiten verdeutlicht wird. Hier liegt FreeBSD regelmäßig auf Platz 1, und dennoch – betrachten wir einmal den deutschen Servermarkt – fällt auf, dass FreeBSD von nur wenigen Anbietern für ihre Server angeboten oder gar beworben wird.

Dies hat mehrere Ursachen, unter anderem die, dass Linux als Sinnbild für quelloffene Betriebssysteme verwendet wird und somit einen höheren Bekanntheitsgrad genießt. Hinzu kommt, dass aufgrund seiner Verbreitung im Desktop-Bereich eine gewisse Vertrautheit bei Einsteigern besteht.

Die geringere Verbreitung von FreeBSD auf Servern, die für private Zwecke genutzt werden, hat allerdings auch einen technischen Hintergrund.

Auf virtuellen Servern, die oft als Einstiegssysteme von »Neulingen« gewählt werden, ist – je nach eingesetztem Virtualisierungsverfahren – die Verwendung eigener Systemkerne (sog. Kernel) und damit der Betrieb von Nicht-Linux-Systemen auf dem Host nicht möglich.

Im Falle von dedizierten Servern liegt die geringere Verbreitung eher an dem aus Anbietersicht gefühlt niedrigeren Interesse und dem damit als unnötig empfundenen Aufwand, Installationsroutinen anzupassen. Oft fehlt es aber auch einfach an entsprechend geschultem Personal. Dennoch haben sich mittlerweile große deutsche Anbieter dazu entschlossen, ihren Kunden auch FreeBSD zur Installation anzubieten.

Für die Wahl von FreeBSD als Betriebssystem sprechen – abgesehen von dem sehr umfangreichen Handbuch, das in viele Sprachen übersetzt ist – vor allem die teils einzigartigen Eigenschaften. Wie sich diese im Einzelnen bemerkbar machen, wirst Du im Laufe dieses Buch noch erfahren.

- FreeBSD ist **konsistent** aufgebaut und somit strikt in Basissystem und Zusatzsoftware getrennt, was die Systempflege und den Einstieg in die UNIX-Welt erheblich vereinfacht.
- Mit dem sogenannten **Portssystem** steht ein sehr umfangreicher Katalog an Software zur Verfügung (über 22.000 Ports).
- Es gibt nur **ein FreeBSD** und keine Vielzahl an Distributionen, die alle unterschiedlich zu bedienen sind.
- Mit **Jails**, einer Art Weiterentwicklung von `chroot`, lässt sich ein FreeBSD-System ohne Virtualisierungsoverhead in mehrere isolierte Subsysteme unterteilen.
- Das vom OpenBSD-Projekt adaptierte Firewallsystem **pf** ist ein sehr leistungsfähiges und gleichzeitig einfach zu konfigurierendes Sicherheitsfeature.

Gerade der erste Punkt ist für Dich wichtig, wenn Du zuvor noch nicht oder wenig mit *NIX-Systemen gearbeitet hast, die alle einer ähnlichen Verzeichnisstruktur folgen.

Ob alle diese Punkte auch von Dir als Vorteil wahrgenommen werden, wirst Du nach dem Durcharbeiten dieses Buchs selbst entscheiden können, da wir alle angesprochenen Punkte gemeinsam behandeln werden.

Netcraft Ltd.: https://ssl.netcraft.com/ssl-sample-report/CMatch/oscnt_all

The FreeBSD Project: http://www.freebsd.org/

FreeBSD Handbuch: http://freebsd.org/doc/de/books/handbook/

1.3 Das Ziel dieses Buchs

Wie ich im vorherigen Abschnitt bereits verraten habe, verfügt FreeBSD über ein hervorragendes Handbuch, und auch die meiste Software, die wir in diesem Buch betrachten werden, ist sehr gut dokumentiert.

Dem Problem, dass Dokumentationen oft lediglich eine Auflistung an Konfigurationsparametern darstellen, wird auf zahlreichen Webseiten mit schrittweisen Anleitungen (sog. Tutorials oder How-To's) begegnet.

Prinzipiell wäre damit bereits alles irgendwo schriftlich festgehalten und müsste lediglich nachgelesen werden. Leider werden in diesen Anleitungen sehr oft nur einzelne Aspekte betrachtet, also entweder wie ein Webserver konfiguriert oder ein Datenbank-Server aufgesetzt wird. Welche Überlegungen sich der Autor aber zum gesamten Systemaufbau gemacht hat, geht dabei oft verloren.

Zudem sind solche Anleitungen häufig unvollständig und eher als Notizen für den Autor gedacht. Ein sicheres System kann allerdings nur mit einem Kon-

zept entstehen, da sonst blind Programme installiert werden, die vermutlich gar nicht benötigt werden und somit ein gewisses Risiko für die Systemsicherheit darstellen.

Das Ziel dieses Buchs ist es daher zum einen, diese Mängel zu beseitigen und Dir eine Sammlung durchgängiger Anleitungen in die Hand zu geben. Zum anderen wollen wir an geeigneter Stelle über den Tellerrand hinaus blicken, sodass Du etwas über die Hintergründe und das Drumherum erfährst.

1.4 Neuerungen in FreeBSD 9

Die Version 9 ist das nächste große Release von FreeBSD und bringt neben Geschwindigkeitsverbesserungen, verbesserter Hardwareunterstützung (beispielsweise USB 3.0) und Fehlerbehebungen auch einige neue Funktionen mit. Da vermutlich nicht alle für Dich relevant sind, stelle ich Dir nur die interessantesten kurz vor.

Hinweis: Eine vollständige Auflistung der Neuerungen findest Du in den Release-Notes auf der Webseite von FreeBSD.

1.5 Der neue Installer

Der bisherige grafische Installer `sysinstall` wurde durch eine Neuentwicklung mit dem Namen `bsdinstall` ersetzt. Dieser hat weniger Abhängigkeiten, ist erweiterbar und lässt sich leicht automatisieren. Zudem bringt er einige neue Funktionen mit, die mit `sysinstall` nur schwer zu realisieren gewesen wären.

Das neue Installationsprogramm bietet beispielsweise die Unterstützung von GUID-Partitionstabellen (kurz: GPT). Diese haben gegenüber den klassischen Master Boot Records einerseits den Vorteil, dass sie sich bei Beschädigungen leichter wiederherstellen lassen, da sie zweimal auf der Festplatte hinterlegt sind. Andererseits unterstützen GPT-Partitionen Festplatten mit einer Größe von bis zu 8192 Exabyte (entspricht 8.589.934.592 Terabyte), die sich in bis zu 128 Partitionen aufteilen lassen.

Wie der neue Installer aussieht und funktioniert, sehen wir im nächsten Kapitel, wenn wir FreeBSD installieren.

1.6 High Availability Storage

Der High Availability Storage (kurz: HAST) ist eine der spannendsten Neuerungen in FreeBSD 9. Hierbei handelt es sich um einen Hochverfügbarkeitsspeicher, der auf einer Client/Server-Architektur beruht.

Vergleichbar mit einem RAID1 über ein Netzwerk, schickt der Server Schreibzugriffe an den konfigurierten und verfügbaren Client. Fällt nun der Server aus oder treten Lese- bzw. Schreibfehler auf, übernimmt der Client die Rolle des Servers.

Am sinnvollsten ist der Einsatz von HAST in Verbindung mit UCARP, das die Nutzung einer gemeinsamen IP-Adresse auf mehreren Systemen erlaubt und so die schnelle Neuverteilung der Rollen im Netzwerk ermöglicht.

Hinweis: HAST unterstützt derzeit jeweils nur einen Client.

Bei der Implementierung von HAST wurde darauf geachtet, dass bestehende Programme nicht speziell für den Einsatz in Verbindung mit diesem Speichersystem angepasst werden müssen. Nach der erfolgreichen Konfiguration gibt sich HAST als Festplatte zu erkennen und lässt sich an jeder beliebigen Stelle im Dateisystem einhängen.

HAST: http://wiki.freebsd.org/HAST

UCARP: http://www.ucarp.org/project/ucarp

1.7 Ressourcenbeschränkungen

Eine Funktion, auf die lange gewartet wurde, ist die Beschränkung der Ressourcenverwendung von Prozessen, Benutzern, Jails etc. Die bisherigen Lösungen hatten entweder Schwächen in der Handhabung oder waren nicht so flexibel wie gewünscht.

Mit den neuen Ressourcencontainern (kurz: RCTL) sind die alten Lösungen Geschichte. RCTL speichert Beschränkungen und für deren Anwendung erforderliche Kriterien in einer zentralen Datenbank. Die darin enthaltenen Datensätze beschreiben, ab wann und für wen oder was die definierten Grenzen gelten sollen. Neue Regelsätze können auch zur Laufzeit definiert werden, ohne dass ein Neustart des zu überwachenden Prozesses oder ein Abmelden des betroffenen Benutzers nötig ist. Besonders interessant ist der Einsatz in Verbindung mit Jails, den wir später noch kennenlernen werden.

RCTL: http://wiki.freebsd.org/Hierarchical_Resource_Limits

1.8 Kernel-Dumps auf andere Systeme

Mit der Version 9 hält auch Netdump Einzug in FreeBSD. Netdump ist ein Framework, mit dessen Hilfe sich im Falle von Kernelfehlern Speicherabbilder (sogenannte Dumps) auf ein entferntes System übertragen lassen.

Diese Abbilder sind bei der Fehleranalyse und Ursachenforschung interessant und sollten nicht verloren gehen. Gerade bei verteilten Systemen oder für Netzwerkkomponenten, die ohne großen Festplattenspeicher auskommen müssen (beispielsweise Router oder Firewalls), ist diese Funktion sehr hilfreich.

Netdump: http://permalink.gmane.org/gmane.os.freebsd.current/128178

1.9 Das Sandbox-Framework Capiscum

Capiscum ist ein vergleichsweise junges Framework zur Isolation von Prozessen. Hier wird nicht nur das Basisverzeichnis verlegt, so wie das beim weitverbreiteten `chroot` der Fall ist, es können auch Speicherbereiche und der Zugriff auf Systemkomponenten beschränkt werden.

Der Nachteil an Capiscum ist allerdings, dass es Anpassungen an der zu beschränkenden Software erfordert. Es ist jedoch zu erwarten, dass in naher Zukunft immer mehr Programme eine optionale Unterstützung hierfür bereithalten werden, da das Framework ab FreeBSD 9 Bestandteil des Systems ist.

Capiscum: http://www.cl.cam.ac.uk/research/security/capsicum/