

Linux Einstieg

Linux ist ein freies und quelloffenes Betriebssystem, das für die Verwendung auf Computern, Servern, mobilen Geräten und anderen Geräten konzipiert ist. Es wurde ursprünglich von Linus Torvalds entwickelt und wird seitdem von einer weltweiten Gemeinschaft von Programmierern ständig weiterentwickelt. Linux ist ein zuverlässiges, sicheres und robustes Betriebssystem und wird wegen seiner Flexibilität und Kosteneffizienz häufig im professionellen Bereich eingesetzt. Es wird auch häufig als Alternative zu proprietären Betriebssystemen wie Windows verwendet.

Das zentrale Kernstück des Betriebssystems ist der Linux-Kernel. Er ist verantwortlich für das Verwalten der Ressourcen des Computers, das Ausführen von Programmen, das Verarbeiten von Anweisungen und vieles mehr. Es ist das Herzstück des Betriebssystems, das alle anderen Komponenten zusammenführt.

Wie unter Windows auch, können unter Linux Anwenderprogramme ausgeführt werden. Ausführbare Windows-Programme und Linux-Programme sind hierbei in erster Linie nicht miteinander kompatibel, da die zugrundeliegende Architektur unterschiedlich ist. Viele Programme werden jedoch zwischenzeitlich auch speziell für Linux bereitgestellt.

Während bei Windows viele Programme aus dem Internet heruntergeladen und installiert werden, hat sich unter Linux die Software-Verwaltung über sogenannte Paketmanager etabliert. Paketmanager wie **apt** und **yum** ermöglichen es den Benutzern, Programme aus einer Vielzahl von Quellen zu installieren und zu verwalten. Die Installation kann dabei direkt über das Terminal vorgenommen werden. Die Programme müssen somit nicht manuell heruntergeladen werden um diese zu installieren oder aktualisieren.

Die Shell

Ein ganz spezielles Anwenderprogramm ist die **Shell** - die "Benutzeroberfläche". Es existieren viele verschiedene Shells - in diesem Modul werden wir mit der Bash (**B**ourne **a**gain **s**hell) arbeiten. Diese ist die Standardshell auf Linuxsystemen. Alle Shells stellen dem Benutzer eine Kommandozeile zur Verfügung, mit der Befehle eingegeben werden können, die direkt als Systemaufrufe an das Betriebssystem weitergeleitet werden.

 or type unknown

Es gibt auch grafische Shells für Linux. Dies sind wie bereits von Windows bekannt, Desktop-Umgebungen welche mit der Maus bedient werden können. Bekannte Vertreter sind hierbei Gnome, Unity, KDE, oder Cinnamon.

Die grafischen Shells spielen in diesem Modul keine Rolle. Da Server oftmals ohne Bildausgabe ausgestattet sind, wird auf Server-Umgebungen bevorzugt die Text-Konsole verwendet.

Dateisystem

Wie Windows arbeitet Linux ebenfalls mit Dateien. Im Unterschied zu Windows, hat aber nicht jedes Laufwerk einen eigenen Dateibaum, sondern es existiert ein einziger Dateibaum wo auch die Laufwerke mit eingebunden werden.

 or type unknown

Das Root-Verzeichnis /

Das Root-Verzeichnis ist das oberste Verzeichnis im Linux Dateibaum. Es ist eine Partition, in der das Root-Verzeichnis eingehängt ist. Weiter Laufwerke und Partition werden an beliebigen Stellen im Dateibaum eingehängt (mounted).

Im Wurzelverzeichnis / darf sich neben den weiteren Verzeichnissen /bin /etc /home /opt /tmp /usr und /var nur eine einzige normale Datei befinden. Dies ist der Linux-Kernel. Heutzutage ist der Linux-Kernel jedoch meistens im Verzeichnis /boot ausgelagert.

/bin

In diesem Verzeichnis liegen sogenannte Binaries, also binäre Programme, jedoch nur jene, die für die wichtigsten Arbeiten am System benötigt werden. Alle User haben in diesem Verzeichnis Lesezugriff.

/boot

Hier liegt der Linux-Kernel und alle Dateien, die er zum Starten benötigt. Nur der root-Benutzer hat auf dieses Verzeichnis Zugriff.

/dev

Hier befinden sich die Gerätedateien. Dies sind Schnittstellen zum Kernel, die ein bestimmtes Gerät bezeichnen. Unter Linux kann fast jede Hardware mit einer solchen Gerätedatei angesprochen werden. Wichtig ist, dass diese Dateien keinen physischen Platz auf der Festplatte beanspruchen. Sie existieren nur virtuell.

/etc

Alle systemweit gültigen Konfigurationsdateien (Netzwerk, Bootmanager, Systemstartskripte, u.s.w) werden hier gespeichert. Auch die Dateien für die Benutzerverwaltung (z.B. "passwd" und "shadow" mit den Benutzern und deren Passwörtern) sind hier abgelegt.

Viele Prozesse müssen in der Lage sein, diese Dateien zu lesen, um bestimmte Informationen zu erhalten. Daher muss das Verzeichnis für alle Benutzer lesbar sein.

/home

Jeder dem Betriebssystem bekannte Benutzer hat sein eigenes Verzeichnis, welches in der Regel denselben Namen wie der Benutzer trägt. Wir nennen solche Verzeichnisse die "Heimatverzeichnisse" der Benutzer. Sie befinden sich alle (mit Ausnahme des "root"-Verzeichnisses des Systemadministrators) im Verzeichnis /home. Nach dem Einloggen (Anmeldung am System mit Benutzernamen und Passwort) befinden sich alle Benutzer immer in ihrem Home-Verzeichnis.

In der Regel darf ein Benutzer in seinem eigenen Home-Verzeichnis alles tun - auch Dateien anlegen, löschen und ändern. Ausserhalb seines Verzeichnisses hat ein normaler Benutzer in der Regel nur Leserechte - er darf dort keine Dateien ändern, anlegen oder löschen.

/root

Dies ist das Heimatverzeichnis des Systemadministrators "root". Aber warum befindet sich dieses Verzeichnis nicht im Verzeichnis /home? Das ist ganz einfach: /home wird oft als eigene Partition eingerichtet. Diese Partition wird aber nur beim Systemstart eingebunden. Sollte ein Fehler auftreten, muss der Systemadministrator in der Lage sein, das System zu starten, ohne alle Partitionen einzubinden. Daher muss auf dem Root-Verzeichnis bereits ein funktionierendes System vorhanden sein. Das Home-Verzeichnis ist jedoch ein wesentlicher Bestandteil eines funktionierenden Systems. Deshalb befindet sich das Home-Verzeichnis von "root" direkt auf dem Root-Verzeichnis und in der gleichen Partition.

In frühen Unix-Systemen war das Home-Verzeichnis von "root" tatsächlich das Root-Verzeichnis, aber da ein Home-Verzeichnis immer viele Dateien enthält, wurde dies zu unübersichtlich und sein Home-Verzeichnis wurde in das /root-Verzeichnis verschoben.

/sbin

Wie in /bin gibt es auch hier Binärdateien, allerdings solche, die hauptsächlich vom Systemadministrator verwendet werden - sozusagen "Systembinärdateien".

/tmp

In diesem Verzeichnis werden, wie der Name schon sagt, temporäre Dateien gespeichert, die nach der Benutzung nicht mehr benötigt werden. Alle Benutzer müssen hier schreiben können, aber es ist kein guter Ort, um die eigenen Daten sicher aufzubewahren. Der gesamte Inhalt wird normalerweise bei einem Neustart des Systems gelöscht.

/usr

Dieses Verzeichnis enthält alle wichtigen Programme, die das System anbietet. Dabei steht "usr" nicht für "user", wie oft fälschlicherweise angenommen wird, sondern für "Unix System Resources".

Analog zum Stammverzeichnis finden wir die Unterverzeichnisse **/usr/bin**, **/usr/lib** und **/usr/sbin**, deren Bedeutung der des Stammverzeichnisses entspricht.

- **/usr/local** dient der Trennung von distributionseigenen und fremden, zusätzlich installierten Programmen. Unter **/usr/local** finden wir die gleichen Verzeichnisse wie im Verzeichnis **/usr** selbst. Selbstkompilierte Programme sollten immer in **/usr/local** und nie direkt in **/usr** installiert werden und die Konfigurationsskripte der Programme sehen dies in der Regel als Standard vor.
- **/usr/share** enthält architekturunabhängige Daten, die verschiedene Programme zum Betrieb benötigen und die nicht verändert werden.
- **/usr/src** enthält den Quellcode für alle Programme des Standardsystems.

/var

Dieses Verzeichnis enthält Daten, die oft von Programmen neu geschrieben werden müssen. Ursprünglich waren diese Daten auch unter **/usr** zu finden, aber der Wunsch, **/usr** schreibgeschützt mounten zu können, machte eine Trennung notwendig. Hier werden Mails auf einem Mailserver zwischengespeichert (**/var/spool**) und alle Programme schreiben hier ihre Log-Dateien (**/var/log**).

Daneben gibt es noch einige andere Verzeichnisse und Unterverzeichnisse, die zum Teil leer sind oder nur symbolische Links auf andere Verzeichnisse enthalten und oft aus Kompatibilitätsgründen existieren.

/opt

Nachdem immer mehr grosse Programmpakete installiert wurden, kam die Überlegung auf, diese nicht mehr in das Verzeichnis **/usr** zu legen, sondern ein eigenes Verzeichnis dafür anzulegen. Tatsächlich nutzen aber nur wenige Programme diese Möglichkeit von sich aus und auf vielen Systemen bleibt dieses Verzeichnis leer.

Revision #4

Created 17 October 2022 21:11:31 by Admin

Updated 1 December 2022 01:43:33 by Admin