

SelfLinux-0.10.0



## Was ist Linux?

Autor: Lugs (*philipp.frauenfelder@swissonline.ch*)

Autor: Heiko Degenhardt (*hede@pingos.org*)

Autor: Nico Golde (*nion@gmx.net*)

Formatierung: Matthias Hagedorn (*matthias.hagedorn@selflinux.org*)

Lizenz: GFDL

## **Inhaltsverzeichnis**

**1 Begriffsklärung**

**2 Kurze Einführung**

**3 Eigenschaften von Linux**

**4 Unterstützte Plattformen**

**5 Voraussetzungen**

**6 Wer benutzt Linux?**



**7 Gründe, Linux zu benutzen**

**8 Wer leistet Support für Linux?**

**9 Warum der Pinguin?**



## 1 Begriffsklärung

Es soll hier zunächst einmal klargestellt werden, was man unter dem Begriff "Linux" eigentlich versteht, und wie dieser Begriff hier im SelfLinux-Tutorial (auch) verwendet wird.


Prinzipiell ist mit dem Begriff "Linux" nur der Kernel gemeint, also der Teil des Systems, der z.B. die Ressourcen für die laufenden Programme und die Anwender verwaltet. Obwohl der Kernel ein essentieller Teil des Systems ist, wäre er ohne zusätzliche Software allerdings nutzlos. Um diesen Kernel effektiv nutzen zu können, ist weitere (teilweise sehr systemnahe) Software notwendig, also das eigentliche Betriebssystem. Derzeit wird in den meisten Fällen die Software des GNU-Projektes ( <http://www.gnu.org>) in Verbindung mit dem Linux-Kernel verwendet. Das GNU-Projekt entwickelt bereits seit 1984 Freie Software ( <http://www.gnu.org/philosophy/free-sw.de.html>), unter anderem das Unix-ähnliche Softwaresystem GNU. Als 1991 die Arbeit am Linux-Kernel begann, war dieses Softwaresystem bereits fast fertig, sodass man auf einen Großteil Freier Software für die Linux-Entwicklung zurückgreifen konnte.

Heute sind beide Komponenten, der Linux-Kernel und die GNU-Betriebssoftware, kaum noch voneinander zu trennen. Beide Teile haben sich gegenseitig vorangebracht und gegenseitig befruchtet.

Korrekterweise müsste man also eigentlich, wenn man das Betriebssystem meint, von "GNU/Linux" sprechen, da hier immer der Kernel (Linux) und die Systemwerkzeuge (GNU) gemeinsam gemeint sind.

Es hat sich allerdings heute eingebürgert, dass umgangssprachlich unter dem Begriff "Linux" die Kombination aus Kernel und Systemwerkzeugen gemeint ist. An vielen Stellen in diesem Tutorial (beginnend bereits im vorliegenden Text) wird dies genauso gehandhabt. Es finden sich also als Bezeichnung für das Betriebssystem gleichbedeutend die Begriffe "Linux" und "GNU/Linux", und für den Kernel "Kernel" oder "Linux-Kernel". Dies soll keinesfalls eine Verunglimpfung der hervorragenden Arbeit des GNU-Projektes sein, sondern trägt eher der Tatsache Rechnung, dass diese Benennung heutzutage von den meisten Anwendern verstanden und benutzt wird. Eine detaillierte Erklärung über die Zusammenhänge zwischen GNU und Linux ist auf  <http://www.gnu.org/gnu/linux-and-gnu.html> zu finden. HINWEIS: Natürlich wird auf GNU/Linux-Systemen nicht nur Software eingesetzt, die unter der GNU GPL steht. Es gibt viele andere Freie Lizenzen, unter denen Software stehen kann. Einen ersten Überblick kann man sich auch hier beim GNU-Projekt auf  <http://www.gnu.org/licenses/licenses.html> verschaffen.

## 2 Kurze Einführung

(GNU)/Linux ist ein hochstabiles, sehr schnelles und voll funktionsfähiges Unix-ähnliches Betriebssystem. Es wird von einer Gemeinschaft von tausenden von Leuten im Internet programmiert und unter der GNU General Public License verteilt, das heißt, es ist frei erhältlich. Es wird weltweit von mehreren Millionen Leuten, Organisationen und Firmen benutzt (aktuelle Zahlen gibt es beim  [Linux Counter](#)).

Linux

Das ist "nur" der Kernel. Ein Kernel ist ein Stück Software, das die Kommunikation zwischen den einzelnen Hardwarekomponenten und den Anwenderprogrammen regelt. Das mag trivial klingen, ist aber eine sehr komplexe Aufgabe. Jedes OS (Operating System) hat einen Kernel, nur werden die wenigsten nach dessen Namen benannt.

Betriebssystem (engl. Operating System)

Ein Kernel allein nützt noch (fast) niemandem etwas. Damit ein Mensch mit ihm arbeiten kann braucht ein

OS weitere, elementare Programme, um funktionieren zu können.

### Shell

Eine Shell ist ein Programm, das zwischen dem Benutzer und dem System arbeitet. Auf der Shell hat man die Möglichkeit, Befehle und Programme auszuführen. Zudem verfügt jede Shell über eine Programmiersprache. Damit können Skripte zur Arbeitserleichterung geschrieben werden. Unter Linux ist die **Bash** (Bourne Again Shell) die Standardshell.

### X-Server

Es ist möglich ein Linux-Betriebssystem (also Kernel und "Systemprogramme") auf einem Plattenplatz von <50 MB unterzubringen (zum Beispiel mit Hilfe einer RAM-Disk, wie sie bei Embedded-Linux-Systemen verbreitet ist). Allerdings steht dann mit Sicherheit nur eine Shell (Befehlszeilen-Prompt) als Benutzerschnittstelle zur Verfügung. Heute möchten nur noch wenige den Komfort einer grafischen Benutzeroberfläche missen, trotzdem ist eine grafische Oberfläche (unter Unix X-Server, X Window System oder X11 genannt) nicht Teil des Betriebssystems, sondern ein Programm, das auf diesem läuft.

### Desktop

Der X-Server bringt einem nicht den gewünschten Arbeitskomfort. Dazu dient ein Desktop wie **KDE** oder **GNOME**. Dieser ermöglicht das Arbeiten, wie man es von anderen grafischen Betriebssystemen her kennt. Man verfügt über eine Arbeitsfläche mit Fenstern und kann Programme über Icons starten.

### Bibliothek

Eine Bibliothek ist eine Sammlung von wiederkehrenden Programmteilen. Da viele Programme gleiche oder ähnliche Teile haben, ist es ineffektiv, diesen Teil immer wieder zu programmieren. Daher werden diese Teile in eine externe Datei ausgelagert. Da mehrere Programme darauf zugreifen, kann die Grösse des einzelnen Programms verkleinert werden. Fehlt die Bibliothek, läuft aber das gesamte Programm nicht mehr. Dies sind dann die berühmten fehlenden Abhängigkeiten.

### Anwendungen

Alles was wir so zum Arbeiten mit dem Computer brauchen, also z.B. eine Office-Software, Internet-Tools, Buchhaltungen, Spiele, Bildbearbeitung, Audio-Software, etc.


### Distribution

Alles oben Genannte zusammengestellt und aufeinander abgestimmt ergibt eine Distribution. Linux-Distributoren sind unabhängige Organisationen, die Software für Linux sammeln und vertreiben.

### 3 Eigenschaften von Linux

- \* Entweder 32- oder 64-bittig, abhängig von der Computer-Architektur.
- \* Multi-User: mehrere Benutzer können gleichzeitig auf derselben Maschine arbeiten.
- \* Multi-Plattform: Linux läuft auf den verschiedensten Prozessoren.
- \* Multi-Processing: SMP (Symmetric Multi-Processing) für bis zu 16 Prozessoren.
- \* Clustering: Cluster von beliebiger Grösse (es gibt schon solche mit 520 Maschinen).
- \* Geschützter Speicher: damit ein Prozess nicht das ganze System zum Absturz bringen kann.
- \* Programme, die bestimmte Teile nur nachladen, wenn sie wirklich gebraucht werden.
- \* Virtueller Speicher mit Paging (das heisst nur Teile von Prozessen können auf die Festplatte ausgelagert werden).
- \* Freier Speicher wird in einem Pool verwaltet, der sowohl Programme als auch Disk-Cache bedient: der gesamte freie Speicher kann zum Cachen verwendet werden.
- \* Verwendung von dynamisch gelinkten Bibliotheken; statische Bibliotheken können selbstverständlich auch verwendet werden.
- \* Größtenteils kompatibel mit Unix (POSIX, System V und BSD) Der Unix98-Standard wird ebenfalls angestrebt (und bald erreicht).
- \* POSIX-kompatible Job-Kontrolle. Realtime Scheduling ist ebenfalls möglich.
- \* Ein nicht fragmentierendes Dateisystem (Extended 2) mit langen Dateinamen (255 Zeichen) und Partitionen von bis zu 4 Terabyte. Es unterstützt Hard- und Softlinks.
- \* Unterstützt zahlreiche Dateisysteme, unter anderem VFAT / FAT32, Windows NT / NTFS, MacOS / HFS, QNX, BSD 4.3 ufs, Coda und Amiga / AFFS.
- \* Bis zu 64 virtuelle Konsolen.
- \* Der ganze Quellcode ist erhältlich, eingeschlossen der ganze Kernel und (fast) alle Treiber.
- \* Schnell, sogar richtig schnell! Und wahrscheinlich der schnellste TCP/IP-Stack der Welt.
- \* Ein Großteil der PC-Hardware wird unterstützt, dazu gehören auch Sound-, Ethernet-, ATM-, Appletalk-, TV-, ISDN- und Multiport-Karten, Joysticks etc.
- \* Viele Netzwerk-Protokolle wie TCP/IP Versionen 4 und 6, IPX/SPX, Token-Ring, Ethertalk, Appletalk etc.
- \* Viele konfigurierbare Netzwerk-Funktionen wie Masquerading (NAT), Tunneling, Forwarding, Routing, Firewalling etc.

## 4 Unterstützte Plattformen

Intel 80386:	PCs mit Intel-, AMD- oder Cyrix-Prozessoren, die zum 80386 kompatibel sind
Intel IA-64:	der neue Intel-Prozessor wird von Beginn an unterstützt.
Intel 8086:	Es gibt ein Projekt mit dem Ziel, Linux auf den 8086 zu portieren. Dieses Vorhaben ist noch nicht ganz abgeschlossen.
Power PC 604:	Power Macintosh und Kompatible. Beachten Sie, dass es zwei Linux-Versionen für den PPC gibt: eine generische und eine, die auf dem Mach-Microkernel aufbaut.
Motorola 68020:	Amiga oder Atari ST mit einem 68020 brauchen einen zusätzlichen Coprozessor für die Speicherverwaltung (PMMU MC 68851) oder einen 68030. Ebenso Sun 3/80 und Classic Macintosh.
Motorola 68000:	MicroLinux und uClinux für 68000-Prozessoren ohne MMU: Palmpilot, Amiga 500, ColdFire, Atari ST und verschiedenste Embedded-Systeme
MIPS R3000:	einige Decstations, möglicherweise Nintendo 64.
ARM:	Acorn Archimedes.
Alpha:	DEC Alpha und Kompatible. Dieser Port hat immer noch Geschwindigkeitsprobleme.
Sparc:	Sun4c und höher. Ebenso UltraSparc-Prozessoren. Das ist einer der schnellsten und stabilsten Ports. Für aktuelle Informationen sei auf  <a href="http://www.ultralinux.org">http://www.ultralinux.org</a> verwiesen.

## **5 Voraussetzungen**

Linux benötigt mindestens 4 MB RAM, beziehungsweise 8 für das X Window System. 128 MB werden empfohlen. Es ist möglich, ein Minimal-System mit weniger als 50 MB Festplatten-Platz zu installieren, aber wir empfehlen 1 GB. Für die CPU ist bei Intel Prozessoren ein 80386sx das Minimum.


## **6 Wer benutzt Linux?**

Viele Privatpersonen benutzen es als Workstation; speziell Studenten, System-Betreuer und Programmierer. Linux wird auch in vielen Firmen eingesetzt, dort vor allem als Server für verschiedene Aufgaben wie Mail, WWW, File-Server, Firewalls oder auch als Application-Server. Einige Firmen benutzen Linux als über das Netz wartbare Thin Clients.

## **7 Gründe, Linux zu benutzen**

Es ist schneller, stabiler und besser skalierbar als so manches kommerzielle System. Anschaffungs- und Unterhaltskosten sind niedrig, was zu einem hervorragenden Preis-/Leistungsverhältnis führt. Die Hardwareunterstützung ist besser als bei den meisten anderen Unix-Systemen. Die Quellen sind verfügbar. Fehler werden sehr schnell behoben, oft innerhalb weniger Stunden nach der Entdeckung.

## **8 Wer leistet Support für Linux?**

Sie erhalten gratis Support von zahlreichen Internet-Newsgroups, Mailinglisten und ungezählten WWW-Seiten. Sie haben die Möglichkeit, dem Programmierer einer Software direkt eine Mail zu schicken. Sie können auch mit Ihrer lokalen Linux oder Unix User Group Kontakt aufnehmen. Wenn Sie kommerziellen Support benötigen, können Sie sich an einen der Distributoren oder eine Firma, die Linux einsetzt und unterstützt, wenden. Eine Liste solcher Firmen erscheint beispielsweise monatlich im Linux-Magazin ( <http://www.linux-magazin.de>).

## 9 Warum der Pinguin?



Der Tux

Viele haben sich sicher schonmal gefragt, warum gerade ein Pinguin das am weitesten verbreitete Symbol für Linux auf der Welt ist.

Alles begann damit, dass eine Linux User Group in Bristol in England für *Linus Torvalds* einen Pinguin gekauft hat. Linus machte sich auf den weiten Weg um das Tier zu besuchen. Als er dort ankam, stellte sich heraus, dass sie ihm keinen Pinguin gekauft, sondern nur in seinem Namen eine Patenschaft für einen Pinguin übernommen hatten. Aber wie kam es jetzt, dass der Pinguin als Symbol für Linux genommen wurde?

Dies war die Idee von *Tove*, seiner Frau. *Linus* versuchte ein Symbol für Linux zu finden, da ihn viele Leute darauf ansprachen, dass es gut wäre, ein eigenes Symbol zu haben. Die anderen Linux Firmen hatten ja zu dem Zeitpunkt auch schon alle eigene Logos. Seine Frau dachte bei der Logosuche an Pinguine, weil Linus im einem Zoo in Australien einmal von einem Pinguin gebissen wurde. Dies geschah beim Versuch, die Pinguine einfach so über ihren Zaun hinweg zu streicheln.

Nach diesem Erlebnis entwickelte *Linus* komischerweise eine Leidenschaft für Pinguine und *Tove* machte den Vorschlag, er solle doch einfach einen Pinguin als Logo nehmen. Linus sprach die Idee des Pinguins nach langem Überlegen mit *Henry Hall* und *Maddog* ab und schließlich entschieden sie sich dafür, dass der Pinguin gut ist.

Als nächstes spornte Linus die Internetgemeinde an, ihm Bilder von Pinguinen zu schicken. Linus gab sich jedoch nicht mit einem x-beliebigen Pinguin ab, nein, sein Pinguin sollte glücklich aussehen, "so als hätte er eine Maß Bier genossen und den besten Sex seines Lebens gehabt". Er wählte die Version von *Larry Ewing*, einem Grafiker, der am Institute for Scientific Computing an der A&M University in Texas arbeitete, aus. Der Pinguin sollte unverwechselbar sein, was er mit *Tux*, so sein Name, auch geschafft hat.

Während normale Pinguine einen schwarzen Schnabel und schwarze Füße haben, sollte das Linux Maskottchen wegen der Einzigartigkeit einen orangen Schnabel und orangefarbene Füße haben, so dass es so aussieht, als sei die Mutter des Pinguins eine Ente.

"Als hätte Daisy Duck sich auf einer Antarktis-Kreuzfahrt vergessen und einen wilden One-Night-Stand mit einem einheimischen Federvieh gehabt".

Nähere Informationen in dem Buch:  
Linus Torvalds und David Diamond



Just for fun.  
Wie ein Freak die Computerwelt revolutionierte  
München 2001  
ISBN: 3-446-21684-7