

Linux Handbuch für Einsteiger

Der leichte Weg zum Linux-Experten

Markus Neumann

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;
detaillierte bibliografische Informationen sind im Internet über <http://dnb.d-nb.de> abrufbar.

©2021 BMU Media GmbH
www.bmu-verlag.de
info@bmu-verlag.de

Lektorat: Lektormeister
Einbandgestaltung: Pro ebookcovers Angie
Druck und Bindung: Wydawnictwo Poligraf sp. zo.o. (Polen)

Taschenbuch-ISBN: 978-3-96645-123-9
Hardcover-ISBN: 978-3-96645-124-6
E-Book-ISBN: 978-3-96645-122-2

Dieses Werk ist urheberrechtlich geschützt.
Alle Rechte (Übersetzung, Nachdruck und Vervielfältigung) vorbehalten. Kein Teil des Werks darf ohne schriftliche Genehmigung des Verlags in irgendeiner Form – auch nicht für Zwecke der Unterrichtsgestaltung- reproduziert, verarbeitet, vervielfältigt oder verbreitet werden.

Dieses Buch wurde mit größter Sorgfalt erstellt, ungeachtet dessen können weder Verlag noch Autor, Herausgeber oder Übersetzer für mögliche Fehler und deren Folgen eine juristische Verantwortung oder irgendeine Haftung übernehmen. Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären.

Linux Handbuch für Einsteiger

Inhaltsverzeichnis

1. Einführung	7
1.1 Was ist das Betriebssystem?	8
1.2 Die Entstehungsgeschichte von Linux.....	12
1.3 Linux und UNIX	16
1.4 Vielfältige Einsatzbereiche für Linux-Betriebssysteme	18
1.5 Der erste Kontakt mit Linux: Ein Live-System erzeugen	20
1.6 Die Linux-Community: So entsteht das Betriebssystem	28
1.7 Übungsaufgaben	30
2. Der Linux-Kernel	33
2.1 Was ist ein Kernel?.....	33
2.2 Die technischen Eigenschaften des Linux-Kernels	37
2.3 Verschiedene Kernel-Versionen	38
2.4 Übungsaufgaben	40
3. Linux-Distributionen	43
3.1 Was ist eine Linux-Distribution?	43
3.2 Verschiedene Arten von Distributionen	49
3.3 Die wichtigsten Linux-Distributionen im Überblick	51
3.4 Eine passende Linux-Distribution auswählen	56
3.5 Übungsaufgaben	60
4. Vorüberlegungen und Vorbereitungsmaßnahmen	63
4.1 Ausschließlich Linux verwenden oder ein Multi-Boot-System implementieren?	63
4.2 Die Partitionen der Festplatte vorbereiten	67
4.3 Weitere Vorbereitungsmaßnahmen	76
4.4 Übungsaufgaben	77
5. Installationsbeispiele für verschiedene Linux-Distributionen	80
5.1 Linux Mint	81
5.2 OpenSUSE	91
5.3 Debian	98
5.4 Manjaro Linux.....	102
5.5 Ubuntu	109
5.6 Übungsaufgaben	116

6.	Die grundlegenden Funktionen von Linux	119
6.1	Das Dateisystem in Linux	119
6.2	Programme und Prozesse	125
6.3	Textverarbeitung und weitere Office-Programme.....	131
6.4	Drucker anschließen und Treiber installieren.....	138
6.5	Texteditor zum Programmieren	142
6.6	Multimedia-Anwendungen	147
6.7	Übungsaufgaben	150
7.	Die Shell: Wichtiges Werkzeug für die Arbeit mit Linux	153
7.1	Was ist eine Shell?	153
7.2	Verschiedene Shells in Linux.....	157
7.3	Dateien und Ordner verwalten.....	163
7.4	Anwender und Zugriffsrechte verwalten	173
7.5	Programme installieren.....	189
7.6	Wildcards.....	194
7.7	Pipes und Filter	202
7.8	Die Befehle cat, head und tail: Inhalte von Dateien anzeigen	209
7.9	Nano Editor, VIM und weitere VI-Editoren über die Shell aufrufen	218
7.10	Informationen mit help, man und info abrufen.....	224
7.11	Übungsaufgaben	233
8.	Shell-Programme erstellen	236
8.1	Was ist ein Shell-Script?	236
8.2	Die ersten Shell Scripts erstellen	239
8.3	Variablen verwenden	248
8.4	Arrays	255
8.5	Einnahmen durch den Anwender aufnehmen.....	259
8.6	Verzweigungen	261
8.7	Schleifen in der Shell ausführen.....	272
8.8	Funktionen in der Shell verwenden	283
8.9	Übungsaufgaben	291
9.	Automatischer Aufruf von Programmen mit cron und at	296
9.1	Cron und crontab: Was versteckt sich hinter diesen Begriffen?.....	297
9.2	Aufgaben mit cron regelmäßig ausführen	298
9.3	Das Programm at als Alternative für eine einmalige Ausführung	303
9.4	Übungsaufgaben	306
10.	Netzwerke in Linux erstellen	309
10.1	Die Vorbereitungsmaßnahmen für die Einrichtung des Netzwerks.....	310
10.2	Dateien mit anderen Computern teilen	318

10.3	Einen Print-Server zum Drucken einrichten.....	324
10.4	Übungsaufgaben	330
11.	Ubuntu aktualisieren	333
11.1	So nehmen Sie die Aktualisierung vor.....	334
11.2	Übungsaufgaben	336
12.	Glossar	340
13.	Index	347

Kapitel 1

Einführung

Jeder Computer benötigt ein *Betriebssystem*. Dieses ist unter anderem dazu notwendig, die *Hardware* zu verwalten, die Ausführung von Anwenderprogrammen zu organisieren und dem Anwender eine geeignete Oberfläche für die Steuerung des Geräts zur Verfügung zu stellen. Ein Betriebssystem ist nicht nur für gewöhnliche PCs erforderlich. Auch Smartphones, Tablets und sogar Smart-TVs, NAS-Speicher und viele weitere programmierbare Geräte benötigen diese Art von Software. Sie hat eine enorme Bedeutung für die Funktionsweise, die Performance und die Handhabung der entsprechenden Produkte.

Es gibt viele unterschiedliche Betriebssysteme. Im Bereich der Desktop-Computer und Laptops ist beispielsweise Windows führend. Die große Mehrheit dieser Geräte verwendet das bekannte Betriebssystem von Microsoft. Im Bereich der mobilen Endgeräte nimmt hingegen Android von Google eine marktbeherrschende Stellung ein. In beiden Bereichen gibt es auch Konkurrenz von Apple: macOS für gewöhnliche Computer und iOS für Smartphones und Tablets. Diese Betriebssysteme zeichnen sich dadurch aus, dass sie sich nur für die Geräte dieses Herstellers eignen. Das führt dazu, dass die Verbreitung deutlich geringer ist.

Dieses Buch beschäftigt sich jedoch mit dem Betriebssystem Linux. Dieses hat einen deutlich geringeren Bekanntheitsgrad. Viele Anwender haben diesen Ausdruck zwar bereits gehört, doch haben sie noch nie bewusst Linux verwendet. Dennoch spielt das Betriebssystem Linux eine enorme Rolle. Im PC-Segment ist es zwar nicht allzu weit verbreitet. Doch ist es für den Betrieb von Servern von sehr großer Bedeutung. Insbesondere Webserver verwenden häufig Linux-Betriebssysteme. Das hat eine herausragende Rolle für die Gestaltung des Internets zur Folge. Hinzu kommt, dass Linux die Grundlage vieler weiterer Betriebssysteme darstellt. Beim Smartphone-Betriebssystem Android handelt es sich beispielsweise um eine Weiterentwicklung von Linux. Daher hat

ein Großteil der Bevölkerung das Betriebssystem Linux doch schon einmal verwendet – auch wenn sich die Anwender darüber meistens überhaupt nicht darüber bewusst sind.

Dieses Buch stellt das Betriebssystem Linux vor. Dabei erfahren Sie, durch welche Eigenschaften es sich auszeichnet, was bei der Verwendung zu beachten ist und wie Sie es installieren. Darüber hinaus erhalten Sie vielfältige Informationen zur Anwendung und zur Konfiguration des Systems.

1.1 Was ist das Betriebssystem?

Da es sich bei Linux um ein Betriebssystem handelt, besteht der erste Schritt für die Annäherung an dieses Thema darin, zu beschreiben, was ein Betriebssystem überhaupt ist und welche Aufgaben es hat. Auf diese Weise fällt es dann leichter, die Eigenheiten von Linux zu verstehen und die Anwendung zu erlernen.

Das Betriebssystem – das entsprechend der englischsprachigen Bezeichnung Operating System häufig auch als OS abgekürzt wird – ist eine Software, die für den Betrieb eines jeden Computers notwendig ist. Sie ist für die Steuerung sowohl der Systemressourcen als auch der Anwenderprogramme zuständig. Diese beiden Aufgaben sind jedoch eng miteinander verknüpft und lassen sich nicht vollkommen unabhängig voneinander betrachten. Darüber hinaus stellt sie eine Oberfläche für den Anwender zur Verfügung, mit deren Hilfe er das Gerät steuern und die gewünschten Funktionen abrufen kann. Die folgenden Abschnitte stellen die wesentlichen Aufgaben des Betriebssystems etwas detaillierter vor.

Prozessor-Management

Der *Prozessor* stellt das zentrale Element des Computers dar. Es ist möglich, diesem einen bestimmten Befehl zuzuführen. Dieser besteht aus einem Binärcode. Dabei kann es sich beispielsweise um eine Anweisung handeln, eine Information aus einem bestimmten Speicherbereich abzurufen oder eine Berechnung durchzuführen. Jeder Befehl,

der an den Prozessor übergeben wird, hat ein ganz bestimmtes Ergebnis zur Folge. Dabei kann es sich um eine Ausgabe über den Bildschirm oder ein bestimmtes Peripheriegerät handeln oder es ist möglich, den entsprechenden Wert im Arbeitsspeicher abzuspeichern.

Das Betriebssystem ist für die Steuerung dieser Prozesse notwendig. Wenn ein *Anwenderprogramm* ausgeführt werden soll, benötigt dieses selbstverständlich Zugriff auf den Prozessor. Hierfür bestehen mehrere Möglichkeiten. Zum einen bietet das Betriebssystem hierfür eine geeignete Schnittstelle an. Ein Programm kann einen sogenannten System Call durchführen. Dabei handelt es sich um eine Anweisung an das Betriebssystem. Dieses führt sie dann auf dem Prozessor aus und gibt das Ergebnis an das Programm zurück. Die meisten Programme nutzen hingegen einen direkten Zugriff auf den Prozessor und werden so unmittelbar ausgeführt. Dennoch ist das Betriebssystem hierbei involviert. Dieses muss der entsprechenden Anwendung die Berechtigung zum Zugriff auf den Prozessor erteilen. Die Organisationsform, die das Betriebssystem hierbei wählt, hat einen großen Einfluss auf die Performance des gesamten Systems. Daher ist diese Software von großer Bedeutung für die Leistung des Rechners.

Speicherverwaltung

Auch beim Zugriff auf den Arbeitsspeicher ist es notwendig, für Ordnung zu sorgen. Das Betriebssystem muss bestimmte Speicherbereiche für die einzelnen Anwendungen reservieren und gewährleisten, dass andere Programme keinen Zugriff darauf erhalten. Würden zwei Programme den gleichen Speicherbereich parallel nutzen, würden sich die Berechnungen gegenseitig beeinflussen, sodass eine korrekte Ausführung nicht möglich wäre.

Darüber hinaus handelt es sich beim *Arbeitsspeicher* um eine begrenzte Ressource. Es ist nicht ungewöhnlich, dass die verschiedenen Anwendungen, die zu einem bestimmten Zeitpunkt auf dem Computer ablaufen, deutlich mehr Speicherplatz einfordern als vorhanden ist. Auch in diesem Fall ist das Betriebssystem gefordert. Es muss genaue Kriterien

aufstellen, nach denen der Speicherplatz zugeteilt wird. Diese Funktion ist ebenfalls für die Performance des gesamten Systems von enormer Bedeutung.

Steuerung von Peripheriegeräten

Ein Computer verfügt in der Regel über zahlreiche *Peripheriegeräte* – beispielsweise über einen Bildschirm, eine Tastatur, eine Maus und einen Drucker. Auch hierbei ist eine Steuerung erforderlich. Für die genauen Details kommen Hardwaredriver zum Einsatz. Doch auch hierbei ist das Betriebssystem für die Organisation erforderlich. Es muss beispielsweise bestimmen, welche Anwendung Zugriff auf welches Peripheriegerät erhält. Daraufhin muss es mit dem Treiber kommunizieren, um eine korrekte Abwicklung der entsprechenden Aufgabe sicherzustellen.

Verwaltung der Dateien

Das Betriebssystem ist auch für die Verwaltung der *Dateien* auf der Festplatte verantwortlich. Es muss hierbei passende Strukturen erstellen – beispielsweise durch Ordner. Hierbei spricht man vom Dateisystem – oder auf Englisch vom File System. Auch in diesem Bereich ist es wichtig, dass das Betriebssystem kontrolliert, welche Anwendung Zugriff auf welche Bereiche erhält. Es muss die entsprechenden Ressourcen zuweisen und wieder freigeben.

Sicherheit

Eine immer wichtigere Aufgabe des Betriebssystems besteht darin, die Sicherheit aufrechtzuerhalten. Dabei ist insbesondere die Kontrolle der Zugriffsrechte wichtig. Das OS muss stets überprüfen, ob eine Anwendung dazu berechtigt ist, auf eine Ressource zuzugreifen. Darüber hinaus fordert es eine Bestätigung durch den Anwender ein, wenn ein neues Programm installiert werden soll. Viele moderne Systeme verfügen außerdem über integrierte Funktionen, die Schadsoftware und andere Angriffe erkennen und abwehren können.

Bereitstellen einer Benutzeroberfläche

Bei den bisher vorgestellten Aufgaben handelte es sich ausschließlich um die Organisation der internen Prozesse. Das ist sicherlich von großer Bedeutung für die Funktionsweise des Computers. Der Anwender kommt damit in der Regel jedoch nicht in Kontakt. Der Berührungspunkt besteht in der Benutzeroberfläche. Ein Betriebssystem muss es dem Anwender erlauben, den Computer zu steuern – beispielsweise ein Programm aufzurufen oder eine Datei zu speichern. Bei frühen Betriebssystemen kam hierfür ausschließlich ein sogenannter Kommandozeileninterpreter zum Einsatz. Dieser ermöglichte es, schriftliche Befehle einzugeben. Diese Form der Steuerung ist jedoch recht umständlich und außerdem erfordert sie eine genaue Kenntnis der möglichen Befehle. Daher verwenden moderne Betriebssysteme in der Regel eine *grafische Benutzeroberfläche* (GUI = Graphical User Interface). Diese ermöglicht eine einfache und intuitive Steuerung per Maus oder bei vielen Geräten auch per Touchscreen.

Hierbei lässt sich bereits auf eine Besonderheit des Betriebssystems Linux hinweisen. Bei den meisten Konkurrenzprodukten – wie etwa Windows oder macOS – sind die Steuerung der internen Prozesse und die grafische Benutzeroberfläche fest miteinander verbunden und stellen eine Einheit dar. Bei Linux ist dies jedoch nicht der Fall. Diese Bezeichnung bezieht sich eigentlich nur auf den sogenannten Linux-Kernel – den Teil der Software, der die internen Prozesse steuert. Darüber hinaus gibt es sogenannte Linux-Desktops. Diese setzen eine grafische Benutzeroberfläche des Linux-Kernels um. Hierbei gibt es jedoch zahlreiche verschiedene Ausführungen. Daher gibt es Linux-Systeme, die auf den Anwender ganz unterschiedlich wirken.

Integrierte Software

Fast jedes moderne Betriebssystem wird mit einer Reihe verschiedener Software-Produkte ausgeliefert – beispielsweise Texteditoren, Programme für die Multimedia-Wiedergabe oder für die Darstellung von Fotos. Diese sind jedoch eigentlich kein Bestandteil des Betriebssystems. Dennoch stellen sie ein wichtiges Merkmal der verschiedenen

Angebote dar, sodass dieser Aspekt in diesem Buch ebenfalls gelegentlich zur Sprache kommen wird.

Das Betriebssystem – die zentrale Steuerungs-Software des Computers

Die Zusammenstellung der Aufgaben des Betriebssystems hat gezeigt, dass diesem eine zentrale Rolle bei der Verwendung eines Computers zukommt. Es ist für die Steuerung aller Prozesse zuständig, die darauf ablaufen. Es bestimmt, welches Programm welche Ressourcen verwenden kann, und sorgt dabei für eine gute Performance des Systems. Es stellt die Schnittstelle zwischen der Hardware auf der einen Seite und den Programmen und dem Anwender auf der anderen Seite dar. Dazu zählt auch die Bereitstellung einer Benutzeroberfläche, die die Steuerung des Systems erlaubt.

1.2 Die Entstehungsgeschichte von Linux

Bevor wir uns den Anwendungsmöglichkeiten von Linux zuwenden, ist es sinnvoll, einen kurzen Blick auf die Entstehung des Betriebssystems zu werfen. Das gibt bereits wertvolle Einblicke, wie Linux aufgebaut ist und für welche Zwecke sich das Betriebssystem verwenden lässt.

Die Entwicklung von Linux geht auf zwei Stränge zurück. Zum einen gibt es das *GNU-Project*. Dieses war bereits seit Anfang der 80-er Jahre darum bemüht, ein freies UNIX-ähnliches Betriebssystem zu gestalten. Zum anderen hat der finnische Informatiker Linus Torvalds zu Beginn der 90er Jahre ein privates Projekt gestartet und dabei die wesentlichen Bestandteile der ersten Linux-Version entwickelt. Aus der Kombination dieser beiden Stränge entstand das heute bekannte Betriebssystem Linux.

Das GNU-Project

Um die chronologische Reihenfolge der beiden Projekte beizubehalten, beginnen wir mit der Vorstellung des GNU-Projects. Dieses entstand

1983 unter der Federführung von Richard Stallman – einem US-amerikanischen Informatiker und Aktivisten für freie Software. Eines der wesentlichen Ziele dieses Projekts bestand darin, ein freies Betriebssystem zu entwickeln, das ein ähnliches Verhalten wie das Betriebssystem UNIX aufweist.

Dieses Ziel konnte das GNU-Projekt jedoch bis heute nicht erreichen. Die Entwicklung dieses neuen Betriebssystems verlief sehr langsam. Dennoch entstanden dabei zahlreiche wichtige Software-Produkte. Besonders erwähnenswert ist hierbei die GNU Compiler Collection (GCC). Diese enthält freie Compiler für zahlreiche Programmiersprachen – insbesondere für C. Sie kam bereits für die Erstellung vieler Millionen Computerprogramme zum Einsatz und hatte einen großen Einfluss auf die Entwicklung der modernen Informatik.

1

Private Terminal-Emulation von Linus Torvalds

Die zweite Grundlage für das Betriebssystem Linux entstand Anfang der 90er Jahre. Der finnische Informatik-Student Linus Torvalds entwickelte damals eine sogenannte Terminal-Emulation. Dabei handelt es sich um ein Programm, das die wesentlichen Funktionen eines UNIX-Terminals nachstellt. Dabei handelt es sich um einen Bestandteil dieses Betriebssystems, der es erlaubt, es mithilfe schriftlicher Kommandos zu steuern. Das *Terminal* spielt eine entscheidende Rolle für die Verwendung von UNIX und für die Gestaltung von Software für dieses Betriebssystem.

Der Grund dafür bestand darin, dass Torvalds dieses UNIX-Terminal für sein Studium an der Universität nutzte. Das dortige Rechenzentrum verwendete, wie zu dieser Zeit üblich, UNIX-Server. Torvalds selbst nutzte auf seinem privaten Rechner das Betriebssystem Minix – ein unixoides Betriebssystem, das ausschließlich für Bildungszwecke eingesetzt wurde.

Die Terminal-Emulation war notwendig, um sich per Internet mit dem Computer seiner Universität zu verbinden. Minix stellte diese Funktion zwar zur Verfügung, allerdings wies das Programm zahlreiche

Schwachstellen auf. Das führte zu dem Problem, dass es für die Studenten damals sehr schwierig war, die Anwendungen, die sie für die Universität gestalteten, auf ihren privaten Rechnern nachzustellen. Torvalds schrieb den Emulator, um dieses Problem zu beheben und, wie er selbst sagte, um seinen Computer besser zu verstehen.

Im Laufe der Zeit entstand daraus jedoch ein immer umfangreicheres Programm, das immer mehr Ähnlichkeiten mit einem Betriebssystem aufwies. Als sich Torvalds darüber bewusst wurde, veröffentlichte er am 25. August 1991 im Diskussionsforum Usenet ein Posting, das enorme Auswirkungen auf die moderne Informatik hatte. Darin kündigte er an, dass er ein privates Betriebssystem als Alternative zu Minix erstellt habe und bat andere Entwickler um Vorschläge für Funktionen, die dabei umgesetzt werden sollten. Auf diese Weise wurde bereits eine beträchtliche Anzahl an Entwicklern auf dieses Projekt aufmerksam. Diese machten Vorschläge für die Verbesserung der ersten Entwürfe und unterstützten Torvalds bei der Umsetzung.

Torvalds verwendete für sein Betriebssystem ursprünglich die Bezeichnung Freax. Diese Wortschöpfung stellte eine Kombination aus den Begriffen „Freak“ und „Free“ dar – mit dem abschließenden „X“, das die Verbindung zu UNIX symbolisiert. Die Bezeichnung Linux in Anspielung an seinen eigenen Vornamen ist Torvalds ebenfalls bereits in den Sinn gekommen. Er hatte sie jedoch verworfen, da er sie als zu egozentrisch empfand. Als er das Betriebssystem jedoch auf dem Server der Universität in Helsinki veröffentlichte, um es auch anderen Menschen zur Verfügung zu stellen, entschloss sich der hierfür zuständige Mitarbeiter Ari Lemmke dazu, die Bezeichnung Linux zu verwenden. Diese Änderung führte er ohne Absprache mit Torvalds durch. Dieser akzeptierte die Entscheidung jedoch nachträglich und musste zugeben, dass dieser Name deutlich eingängiger ist.

Veröffentlichung unter der GNU-Lizenz

Zunächst handelte es sich bei Linux und GNU um eigenständige Entwicklungen. Allerdings verwendete Torvalds bei der Gestaltung des Betriebssystems bereits GNU-Software. Torvalds gestaltete Linux zwar als

freie Software, doch verbreitete er das Betriebssystem unter einer eigenen Lizenz, die die kommerzielle Verwendung ausschloss.

Von großer Bedeutung für die Entstehung des neuen Betriebssystems war die Veröffentlichung von Linux unter der GNU General Public License (GPL). Das hatte wichtige Auswirkungen auf die verwendete Technik. Bei Linux handelt es sich lediglich um den Systemkern. Die Annäherung an das GNU-Projekt erlaubte es jedoch, die GNU-Software darauf aufzusetzen und auf diese Weise ein komplettes Betriebssystem zu erstellen.

Darüber hinaus erlangte das Betriebssystem auf diese Weise deutlich größere Bekanntheit. Durch die Veröffentlichung unter GNU GPL wurden Programmierer in aller Welt auf das neue Projekt aufmerksam. Das wirkte sich nicht nur positiv auf die Verbreitung aus. Darüber hinaus entstand auf diese Weise eine große Community, die an der Entwicklung mitwirkte. Torvalds sagte rückblickend zu diesem Schritt: „Making Linux GPL'd was definitely the best thing I ever did.“ (Interview mit Hiroo Yamagata, August 1997).

Trotz dieser Entwicklung nutzten während der Anfangszeit fast ausschließlich Informatik-Studenten und professionelle Programmierer Linux. Der Grund dafür bestand zum einen darin, dass der Installationsprozess damals ausgesprochen kompliziert war. Es war in der Regel notwendig, den Code in einem Rechenzentrum in einer Universität oder in einem Unternehmen über einen FTP-Server herunterzuladen, um diesen anschließend auf bis zu 30 Disketten zu überspielen. Zum anderen brachte Linux diesem Anwenderkreis den größten Nutzen. Da an Universitäten und in größeren Unternehmen fast ausschließlich UNIX-Server zum Einsatz kamen, arbeiteten die Programmierer normalerweise mit diesem System. Das führte jedoch dazu, dass sie ihre Programme auf ihrem privaten PC nicht ausprobieren konnten, da hierbei ein anderes Betriebssystem zum Einsatz kam. Erst mit der Entwicklung von Linux, das viele Gemeinsamkeiten mit UNIX aufweist, wurde dies möglich.

Bis Linux massentauglich wurde, sollten noch einige Jahre vergehen. Doch entstanden immer neue Linux-Distributionen, die die Installation und die Anwendung deutlich erleichterten. Mittlerweile stellt die

Verwendung dieses Betriebssystems auch für Nicht-Informatiker kein großes Problem mehr dar.

1.3 Linux und UNIX

In den bisherigen Abschnitten wurde mehrfach das Betriebssystem *UNIX* erwähnt. Das zeigt bereits, dass dieses einen großen Einfluss auf die Entwicklung von Linux hatte. Häufig werden die Begriffe Linux und UNIX sogar synonym zueinander verwendet – obwohl dies eigentlich nicht korrekt ist. Aufgrund der Verbindungen zwischen den beiden Betriebssystemen ist es aber dennoch sinnvoll, einen kurzen Blick auf UNIX zu werfen und die Gemeinsamkeiten und Verschiedenheiten zu Linux herauszuarbeiten.

UNIX – ein wichtiges Betriebssystem der 70-er und 80-er Jahre

Die Geschichte von UNIX beginnt in den frühen 60-er Jahren. Damals schlossen sich die Unternehmen General Electrics und AT & T Bell Labs mit dem Massachusetts Institute of Technology (MIT) zusammen, um ein gemeinsames Betriebssystem zu entwickeln. Dieses sollte den Namen Multics erhalten. Eines der wesentlichen Ziele bestand dabei darin, dass dieses mehrbenutzerfähig sein sollte. Allerdings war der gewählte Ansatz ausgesprochen komplex, sodass es bei diesem Projekt nur zu geringen Fortschritten kam.

Aus diesem Grund beschloss AT & T Bell Labs, aus diesem Projekt auszusteigen und ein eigenes Betriebssystem zu entwickeln. Zu der Gruppe, die das gemeinschaftliche Projekt verließ, gehörten unter anderem Ken Thompson und Dennis Ritchie – zwei der einflussreichsten Informatiker des 20. Jahrhunderts. Unter ihrer Federführung entstand ein neues Projekt, das im Prinzip die gleichen Ziele verfolgte wie Multics, aber wesentlich einfacher aufgebaut war. Das Ergebnis daraus war das Betriebssystem UNIX, das im August 1969 erschien.

Die erste Version war in Assemblersprache geschrieben. Dabei handelt es sich um eine Programmiersprache, die ein sehr niedriges Abstrak-

tionsniveau aufweist. Das bedeutet, dass sich die Befehle genau an den Kommandos im Binärcode orientieren, die dem Prozessor zugeführt werden. Das hat wiederum zur Folge, dass der Programmcode genau auf die verwendete Hardware abgestimmt sein muss. Aus diesem Grund war das erste UNIX-Betriebssystem lediglich auf dem Rechner PDP-7 lauffähig, für den es entwickelt wurde.

Dennis Ritchie entwickelte später die Programmiersprache C, die bis heute zu den beliebtesten Programmiersprachen zählt. Thompson und Ritchie verwendeten diese daraufhin auch für die Entwicklung ihres Betriebssystems. Das hatte zur Folge, dass sich dieses deutlich leichter auf andere Rechnerarchitekturen übertragen ließ.

UNIX kam in den 70-er Jahren hauptsächlich in Universitäten zum Einsatz. Zu diesem Zeitpunkt war das Betriebssystem frei verfügbar und der Quellcode offen zugänglich. Ab den 80-er Jahren begann AT & T mit der kommerziellen Nutzung der Software. Fortan fielen erhebliche Lizenzgebühren für die Nutzung an. Außerdem war es nicht mehr erlaubt, den Quellcode zu veröffentlichen. Das stellte insbesondere für den Lehrbetrieb an den Universitäten eine erhebliche Einschränkung dar. Hier kam der UNIX-Quellcode häufig zum Einsatz.

Linux als UNIX-ähnliches System

UNIX war bereits seit den 70-er Jahren das gebräuchlichste Betriebssystem für die Server an Universitäten in aller Welt. Durch die Änderung der Lizenzierung zu Beginn der 80-er Jahre war die Verwendung des Quellcodes in den Vorlesungen jedoch nicht mehr möglich. Darüber hinaus machten es die hohen Lizenzgebühren für die meisten Studenten und Studentinnen unmöglich, ein UNIX-System auf ihren privaten Computern zu installieren. Aus diesem Grund war es schwierig, die Programme, die für die Universität entstanden, in einem privaten Umfeld nachzustellen.

Ein erster Schritt, um eine Alternative zum proprietären UNIX zu erstellen, bestand darin, einen allgemeinen Standard für Betriebssysteme zu

entwickeln. Treibende Kraft hinter diesem Bestreben war Richard Stallman, der auch für das GNU-Projekt verantwortlich war. Das führte zur Verabschiedung des Standards POSIX im Jahre 1988. Dieser beschreibt einheitliche Anforderungen an die Schnittstelle, die für die Kommunikation zwischen dem Betriebssystem und der Anwendersoftware zuständig ist. POSIX orientiert sich an den Funktionen von UNIX. Das bedeutet, dass sich Programme, die für UNIX geschrieben sind, auch für andere Systeme kompilieren lassen, die POSIX umsetzen.

Das dominierende Betriebssystem für private Computer war zu dieser Zeit jedoch MS-DOS. Dieses beachtet jedoch den POSIX-Standard nicht. Das führte zu der beschriebenen Inkompatibilität zwischen den privaten Rechnern und den Servern an der Universität.

Linux schloss diese Lücke. Es erfüllte zahlreiche Anforderungen, die viele Studenten der damaligen Zeit an ihr Betriebssystem stellten: Es war kostenfrei nutzbar, quellenoffen, für die Verwendung auf dem PC geeignet und POSIX-konform. Daraus geht auch die Verbindung zwischen Linux und UNIX hervor. Bei Linux handelt es sich zwar um ein selbstständiges Betriebssystem, das von Grund auf neu entwickelt wurde und das praktisch keinen UNIX-Code verwendet. Da beide Systeme jedoch dem POSIX-Standard entsprechen, ist eine Kompatibilität zwischen den Systemen gegeben.

Die Ähnlichkeiten zwischen UNIX und Linux beschränken sich jedoch nicht auf die Beachtung des POSIX-Standards. Darüber hinaus verwendet Linux zahlreiche Tools, deren Verwendung identisch zu UNIX-Systemen ist. In erster Linie ist hierbei die Shell zu nennen, die im weiteren Verlauf dieses Buchs noch ausführlicher besprochen wird. Hierbei handelt es sich um das zentrale Steuerungswerkzeug der beiden Betriebssysteme. Die Verwendung ist unter UNIX und Linux beinahe identisch.

1.4 Vielfältige Einsatzbereiche für Linux-Betriebssysteme

Linux kommt in vielen verschiedenen Bereichen zum Einsatz. Ursprünglich wurde das Betriebssystem für die Verwendung auf einem

Desktop-Computer entwickelt. Es ist hierbei möglich, Linux als alleiniges Betriebssystem zu installieren. Sehr beliebt ist jedoch auch die Einrichtung eines *Multi-Boot-Systems*. Dieses ermöglicht es, Linux neben einem anderen Betriebssystem zu verwenden – beispielsweise neben Windows oder macOS. Obwohl die Verwendung auf dem Desktop-Computer dem ursprünglichen Einsatzzweck von Linux entspricht, ist der Marktanteil hierbei nur recht gering. Laut Erhebungen von netmarketshare.com liegt der Anteil von Linux für Desktop-PCs und Laptops aktuell (August 2020) bei 2,69 Prozent. Allerdings ist hierbei eine erhebliche Steigerung zu beobachten. Vor etwa einem Jahrzehnt lag der Anteil noch unter einem Prozent.

Besonders beliebt ist die Verwendung von Linux für Server. Dieses Betriebssystem ist sehr stabil, einfach zu warten und es eignet sich für beinahe alle Server-Architekturen. Daher erfüllt es die wesentlichen Anforderungen an ein Server-Betriebssystem. Außerdem ist es modular aufgebaut, sodass sich damit auch sehr gut dedizierte Server einrichten lassen. Darüber hinaus spielt es auch hierbei sicherlich eine Rolle, dass für die Verwendung von Linux keine Lizenzkosten anfallen. Verlässliche Nutzungsstatistiken sind in diesem Bereich nur schwer erhältlich. Viele Beobachter gehen jedoch davon aus, dass mindestens 60 Prozent aller Webseiten auf Linux-Servern gehostet werden.

Auch in anderen Bereichen ist Linux von enormer Bedeutung. Dabei sind insbesondere Smartphones und Tablets zu nennen. Das Betriebssystem Android, das einen Marktanteil von rund 70 Prozent in diesem Bereich aufweist, verwendet beispielsweise einen Linux-Kernel. Allerdings ist diese Verwandtschaft für den Anwender kaum noch zu bemerken. Android verzichtet auf zahlreiche GNU-Tools, die den typischen Charakter gewöhnlicher Linux-Distributionen prägen und die eine starke Verbindung zu UNIX aufweisen. Aufgrund dieser deutlichen Unterschiede wird das Android-Betriebssystem in diesem Buch auch nicht weiter behandelt.

Besonders beeindruckend ist die Position von Linux im Bereich der Supercomputer. Nach Angaben der Organisation TOP500, die Statistiken zu den 500 leistungsstärksten Computern der Welt erhebt, nutzen

derzeit (Stand 2020) ausnahmslos alle hier gelisteten Rechner Linux als Betriebssystem. Allerdings handelt es sich hierbei um Versionen mit einigen Abwandlungen, die auf die spezifischen Architekturen dieser Supercomputer ausgerichtet sind.

1.5 Der erste Kontakt mit Linux: Ein Live-System erzeugen

Eine wichtige Rolle nimmt Linux auch im Bereich der Live-Systeme ein. Bei einem Live-System handelt es sich um ein Betriebssystem, das keine feste Installation erforderlich macht. Stattdessen wird es auf einem externen bootfähigen Medium abgespeichert. Heutzutage kommt hierfür meistens ein USB-Stick zum Einsatz. Früher kamen hierfür jedoch häufig DVDs, CDs und Disketten zum Einsatz. Während viele andere Betriebssysteme keine Live-Version anbieten, hält Linux hierfür unzählige Möglichkeiten bereit.

Allerdings stellt sich noch die Frage, welche Vorteile es bietet, ein Live-System zu verwenden. Dieses bietet viele Anwendungsmöglichkeiten. In der Praxis kommt es häufig zum Einsatz, wenn Probleme mit dem eigentlich installierten Betriebssystem auftreten. Wenn sich dieses nicht mehr starten lässt, stellt das Live-System eine gute Möglichkeit dar, um den Computer dennoch in Betrieb zu nehmen, ohne gleich ein neues System zu installieren, das alle bisherigen Daten überschreiben würde. Das ist insbesondere für die Datenrettung sehr praktisch. Auch wenn es zu einem Befall mit Malware gekommen ist, kann dies hilfreich sein. Das Live-System bietet Zugriff auf alle Dateien der Festplatte – selbst auf die Bereiche, die in der eigentlichen Betriebssystem-Umgebung unter einem besonderen Schutz stehen. Da sich die Schadsoftware häufig in diesen Bereichen einnistet, ist es deutlich einfacher, sie zu entfernen. Auch hinsichtlich der Sicherheit kann die Verwendung eines Live-Systems sinnvoll sein. Sollten Sie etwa sensible Informationen an einem Rechner bearbeiten, auf den auch andere Personen Zugriff haben, ist es möglich, dass einer dieser Nutzer eine Schadsoftware installiert hat, um die Daten auszuspähen. Durch die Verwendung eines Live-Systems stellen Sie jedoch sicher, dass keine anderen Prozesse auf dem Computer ablaufen.

Um die ersten Kapitel, die in erster Linie Grundlagenwissen vermitteln und kaum praktische Bestandteile beinhalten, etwas aufzulockern, wollen wir nun die Verwendung eines solchen Live-Systems ausprobieren. Hierfür müssen wir eine passende Linux-Distribution auswählen. Was das genau ist, wird in Kapitel 3 noch ausführlich beschrieben. An dieser Stelle sei lediglich gesagt, dass es sich hierbei um unterschiedliche Ausführungen des Linux-Betriebssystems handelt, die sich durch verschiedene Anwendungsmöglichkeiten und Gestaltungsweisen auszeichnen.

Zahlreiche Linux-Distributionen bieten die Möglichkeit, ein Live-System zu erstellen. Wir entscheiden uns bei unserem Beispiel für Knoppix. Hierbei handelt es sich um eine der ersten Linux-Distributionen, die ein Live-System anboten. Diese Software gelangte bei diesem speziellen Anwendungszweck zu einer sehr hohen Beliebtheit und trug erheblich zur Verbreitung von Linux-Live-Systemen bei. Obwohl mittlerweile auch eine feste Installation möglich ist, kommt Knoppix bis heute fast ausschließlich als Live-System zum Einsatz.

Der erste Schritt hierfür besteht darin, das Betriebssystem Knoppix herunterzuladen. Dafür bietet es sich an, die Seite des Entwicklers aufzusuchen und dort die Download-Möglichkeiten abzurufen: <https://www.knopper.net/knoppix-mirrors/index-en.html>. Das zeigt, dass hierbei eine große Auswahl besteht. Nun ist es möglich, eine dieser Download-Möglichkeiten aufzurufen. Wir entscheiden uns für den folgenden Link: <http://ftp.uni-kl.de/pub/linux/knoppix-dvd/>. Selbstverständlich wäre es jedoch auch möglich, eine andere Alternative auszuwählen. Es ist empfehlenswert, hierbei stets einen Link auszuwählen, der mit DVD gekennzeichnet ist. Die Version für CDs wird bereits seit längerer Zeit nicht mehr aktualisiert. Sollten Sie keinen FTP-Client installiert haben, ist es außerdem ratsam, einen der HTTP-Download-Links auszuwählen. Die folgende Abbildung zeigt, dass hier nun viele verschiedene Dateien zum Download bereitstehen.

File Name	Date	Time	Size
..			
./			
CD/	17-Sep-2013	21:20	-
md5-oid/	23-Nov-2019	07:24	-
KNOPPIX_V8.6-2019-08-08-DE.iso	10-Aug-2019	13:07	4G
KNOPPIX_V8.6-2019-08-08-DE.iso.md5	17-Aug-2019	14:19	65
KNOPPIX_V8.6-2019-08-08-DE.iso.md5.asc	17-Aug-2019	14:27	309
KNOPPIX_V8.6-2019-08-08-DE.iso.sha1	17-Aug-2019	14:20	73
KNOPPIX_V8.6-2019-08-08-DE.iso.sha1.asc	17-Aug-2019	14:27	317
KNOPPIX_V8.6-2019-08-08-DE.iso.sha256	17-Aug-2019	14:21	97
KNOPPIX_V8.6-2019-08-08-DE.iso.sha256.asc	17-Aug-2019	14:27	341
KNOPPIX_V8.6-2019-08-08-EN.iso	17-Aug-2019	08:50	4G
KNOPPIX_V8.6-2019-08-08-EN.iso.md5	17-Aug-2019	14:23	65
KNOPPIX_V8.6-2019-08-08-EN.iso.md5.asc	17-Aug-2019	14:27	309
KNOPPIX_V8.6-2019-08-08-EN.iso.sha1	17-Aug-2019	14:25	73
KNOPPIX_V8.6-2019-08-08-EN.iso.sha1.asc	17-Aug-2019	14:27	317
KNOPPIX_V8.6-2019-08-08-EN.iso.sha256	17-Aug-2019	14:27	97
KNOPPIX_V8.6-2019-08-08-EN.iso.sha256.asc	17-Aug-2019	14:27	341
KNOPPIX_V8.6.1-2019-10-14-DE.iso	22-Nov-2019	20:35	4G
KNOPPIX_V8.6.1-2019-10-14-DE.iso.md5	22-Nov-2019	21:06	67
KNOPPIX_V8.6.1-2019-10-14-DE.iso.md5.asc	22-Nov-2019	21:10	311
KNOPPIX_V8.6.1-2019-10-14-DE.iso.sha1	22-Nov-2019	21:08	75
KNOPPIX_V8.6.1-2019-10-14-DE.iso.sha1.asc	22-Nov-2019	21:10	319
KNOPPIX_V8.6.1-2019-10-14-DE.iso.sha256	22-Nov-2019	21:09	99
KNOPPIX_V8.6.1-2019-10-14-DE.iso.sha256.asc	22-Nov-2019	21:10	343
KNOPPIX_V8.6.1-2019-10-14-EN.iso	22-Nov-2019	14:44	4G
KNOPPIX_V8.6.1-2019-10-14-EN.iso.md5	22-Nov-2019	20:49	67
KNOPPIX_V8.6.1-2019-10-14-EN.iso.md5.asc	22-Nov-2019	20:52	311
KNOPPIX_V8.6.1-2019-10-14-EN.iso.sha1	22-Nov-2019	20:50	75
KNOPPIX_V8.6.1-2019-10-14-EN.iso.sha1.asc	22-Nov-2019	20:52	319
KNOPPIX_V8.6.1-2019-10-14-EN.iso.sha256	22-Nov-2019	20:52	99
KNOPPIX_V8.6.1-2019-10-14-EN.iso.sha256.asc	22-Nov-2019	20:52	343
dbg-1-dvd-860.txt	17-Aug-2019	13:44	548K
dbg-1-dvd-861.txt	23-Nov-2019	07:28	548K
knoppix-cheatcodes.txt	17-Aug-2019	15:27	10K
packages-dvd.txt	23-Nov-2019	07:28	548K
packages.txt	23-Nov-2019	07:28	548K

Abb. 1.1 Die Download-Möglichkeiten für Knoppix

Hier ist es nun notwendig, eine Datei mit der Endung .iso auszuwählen – ohne weitere Zusätze. Doch stehen zahlreiche Möglichkeiten zur Auswahl, die diese Anforderung erfüllen. Dabei handelt es sich um unterschiedliche Versionen. Wir entscheiden uns für die aktuellste Ausführung und laden diese herunter.

Um einen bootfähigen USB-Stick zu erhalten, ist ein weiteres Programm notwendig. Auch hierfür gibt es mehrere Möglichkeiten. Wir entscheiden uns für den Universal USB Installer, weil dieser zum einen gratis erhältlich und zum anderen einfach zu verwenden ist. Er steht unter dem folgenden Link bereit: <https://www.pendrivelinux.com/universal-usb-installer-easy-as-1-2-3/>.

Nun müssen wir dieses Programm herunterladen und ausführen. Die folgende Abbildung zeigt das Fenster, das sich nach der Zustimmung zu den Lizenzbedingungen öffnet. Unter Step 1 ist es nun notwendig, die Distribution KNOPPIX auszuwählen. Sie befindet sich unter der Überschrift „Security and Penetration Testing“ beinahe ganz am Ende der Liste. Unter Step 2 ist es nun notwendig, die iso-Datei auszuwählen, die wir zuvor heruntergeladen haben. Außerdem müssen Sie noch den USB-Stick angeben, auf dem Sie das System installieren möchten. Beachten Sie hierbei, dass bei diesem Prozess alle Daten, die zuvor auf

dem USB-Stick gespeichert waren, verlorengehen. Falls Sie hier wichtige Dateien aufbewahren, sollten Sie diese zuvor an einem anderen Ort abspeichern. Nachdem Sie die Auswahl bestätigt haben, kommt es zu einem Installationsprozess, der einige Minuten dauern kann.

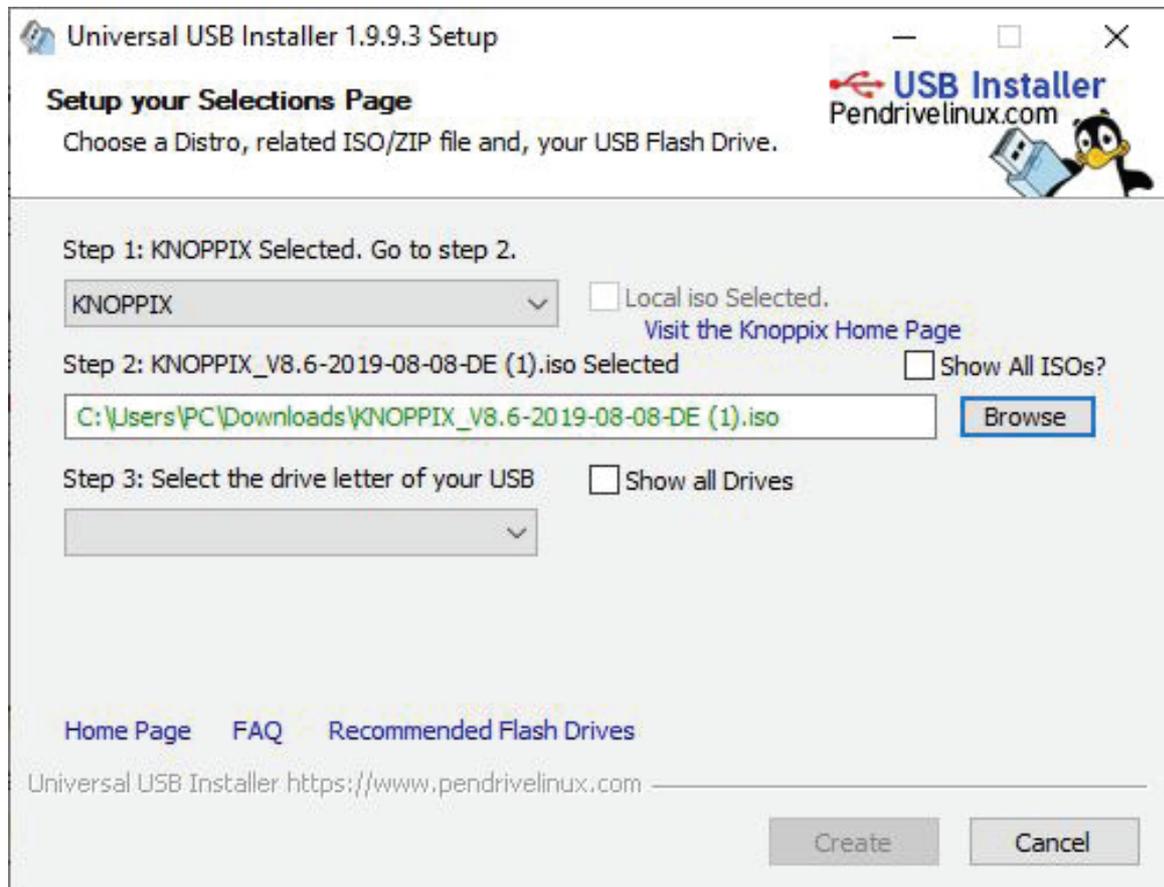


Abb. 1.2 Einen bootfähigen USB-Stick erstellen

Nun können Sie Ihr erstes Linux-Betriebssystem ausprobieren. Dazu müssen Sie den soeben erstellten USB-Stick in den Computer einführen und diesen daraufhin erneut starten. In der Regel erscheint nun nicht das gewohnte Betriebssystem, sondern KNOPPIX. Sollte dies nicht der Fall sein, liegt dies in der Regel daran, dass der USB-Port an Ihrem Computer nicht zum Booten vorgesehen ist. Um dies zu ändern, müssen Sie beim Anschalten zum Boot-Menü wechseln. Die folgenden Abschnitte zeigen, wie Sie dabei vorgehen. Sollte Ihr Computer das Live-System bereits ordnungsgemäß gestartet haben, können Sie diesen Bereich überspringen.

Bei Computern, die mit *BIOS* arbeiten, ist es hierfür notwendig, direkt nach dem Start eine bestimmte Taste oder Tastenkombination zu drücken. Um welche Tasten es sich dabei handelt, ist jedoch von Gerät zu Gerät unterschiedlich. Häufig handelt es sich hierbei um die Entf-Taste (auf Englisch Del) oder um die F2-Taste. Es bestehen jedoch noch weitere Möglichkeiten. Normalerweise erscheint jedoch direkt nach dem Starten eine kurze Nachricht, die angibt, welche Taste hierfür notwendig ist wie etwa: „Press to enter setup“.

Die meisten neueren Rechner arbeiten jedoch mittlerweile mit einer *UEFI*-Firmware. Hierbei ist es notwendig, zunächst Windows wie gewohnt zu starten. Daraufhin bestehen zwei verschiedene Möglichkeiten. Zum einen können Sie bei gedrückter Umschalt-Taste (Shift) auf die Schaltfläche „Neu Starten“ klicken. Die andere Alternative besteht darin, die Windows-Einstellungen über das Zahnrad-Symbol im Start-Menü aufzurufen. Wählen Sie daraufhin „Update und Sicherheit“ und anschließend „Wiederherstellung“ aus. Unter der Überschrift „Erweiterter Start“ entdecken Sie nun die Schaltfläche „Jetzt neu starten“. Bei beiden vorgestellten Möglichkeiten erscheint anschließend ein blaues Fenster mit mehreren Optionen. Hierbei müssen Sie zunächst „Problembehandlung“ und anschließend „Erweiterte Optionen“ und abschließend „UEFI Firmwareeinstellungen“ auswählen.

Unabhängig davon, ob Sie der Anweisung für Rechner mit BIOS oder mit UEFI gefolgt sind, gelangen Sie im nächsten Schritt zum Boot-Menü. Achten Sie dabei darauf, dass bei diesem Schritt der USB-Stick bereits eingelegt ist. Bei manchen Rechnern ist dies notwendig, damit diese den USB-Port als bootfähiges Laufwerk erkennen.

Das Boot-Menü kann - je nach Gerät - einige Unterschiede aufweisen. Allerdings sind stets konkrete Anweisungen vorhanden, wie Sie das System steuern können. Im Folgenden wird eine Möglichkeit dafür vorgestellt, wie Sie die Einstellungen vornehmen können. Es ist aber möglich, dass Ihr Rechner davon in manchen Punkten abweicht. In diesem Fall müssen Sie die Anweisungen im Boot-Menü beachten.

Über die Pfeiltasten müssen Sie zunächst zum Reiter „Boot“ navigieren. Hier erscheint eine Zeile mit der Beschriftung „USB Boot“. Sollte diese bereits als „Enabled“ ausgezeichnet sein, sind keine weiteren Aktionen notwendig. Sollte dies nicht zutreffen, müssen Sie sie mit der Enter-Taste auswählen und daraufhin den entsprechenden Wert vorgeben.

Weiter unten erscheint die Überschrift „Boot Priority Order“. Hier sind nun verschiedene Laufwerke angegeben, die zum Booten infrage kommen. Hier müssen Sie dafür sorgen, dass das USB-Laufwerk in der Liste vor der Festplatte erscheint. Um die einzelnen Laufwerke zu verschieben, kommen häufig die F5- und die F6-Taste zum Einsatz. Nachdem Sie diese Einstellung vorgenommen haben, können Sie „Exit“ und daraufhin „Exit Saving Changes“ auswählen, um die entsprechenden Einstellungen zu übernehmen.

Wenn Sie daraufhin den Computer bei eingelegtem USB-Stick neu starten, sollte nun das Linux Live-System erscheinen. Abbildung 3 zeigt, wie dieses bei der Version 8.6 aussieht.

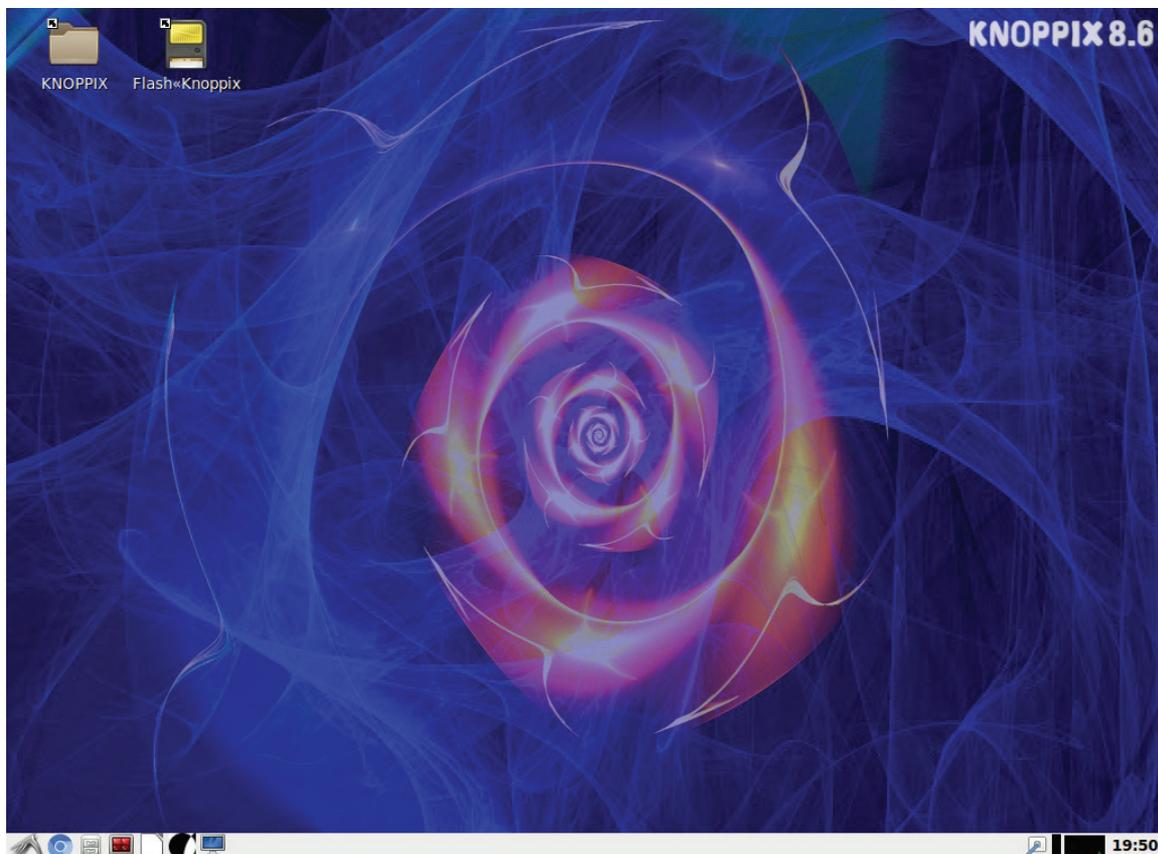


Abb. 1.3 Die Oberfläche von KNOPPIX

Damit haben Sie bereits zum ersten Mal ein Linux-Betriebssystem gestartet. Um zu zeigen, was damit alles möglich ist, können Sie einmal auf den Ordner mit der Beschriftung „KNOPPIX“ klicken, der sich links oben auf der Desktop-Oberfläche befindet. Dessen Inhalt ist in Abbildung 4 zu sehen.

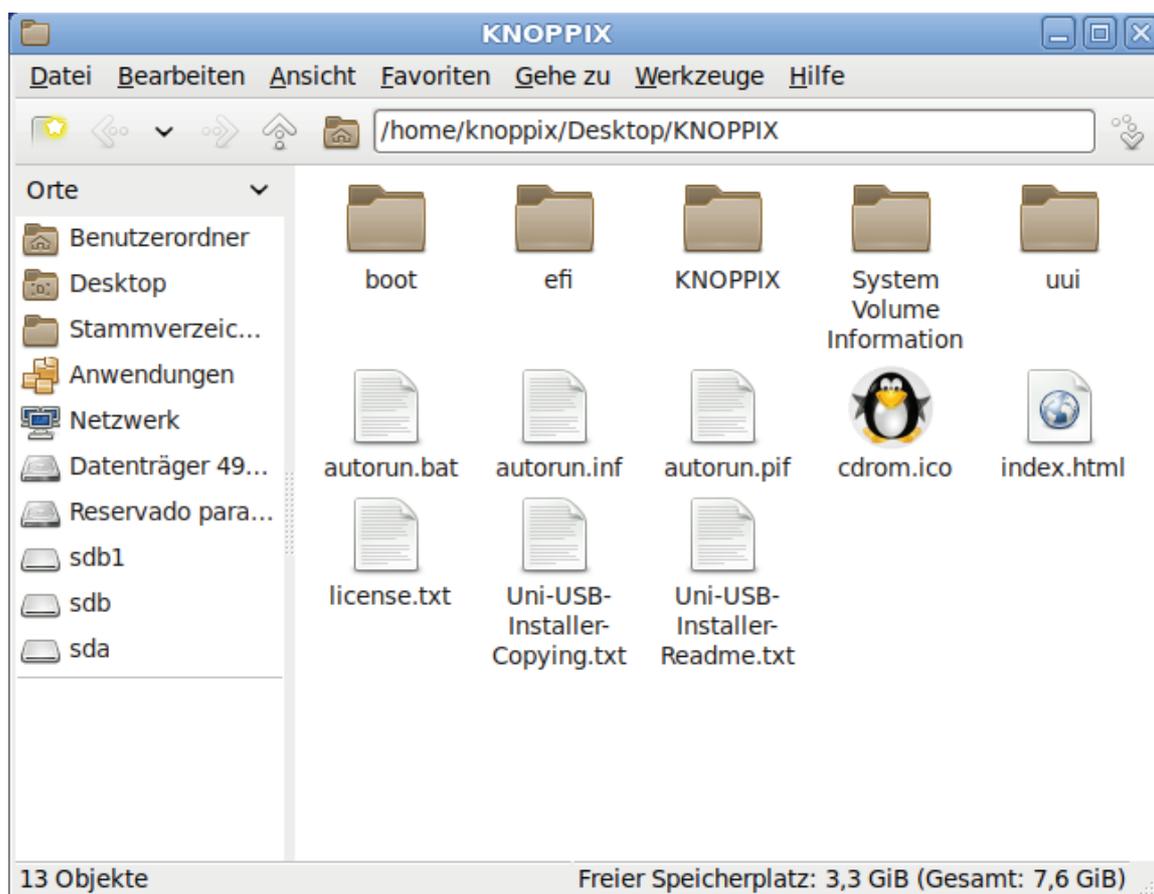


Abb. 1.4 Die Dateien der Festplatte

Wenn Sie hier in der linken Spalte auf den Begriff „Datenträger“ mit dem Festplattensymbol klicken, gelangen Sie zu den Dateien, die auf der Festplatte des Computers abgespeichert sind. Das bedeutet, dass Sie dadurch auf alle Inhalte zugreifen können, die sich auf Ihrem PC befinden. Das ist beispielsweise für die Datenrettung bei einem defekten Betriebssystem ausgesprochen hilfreich.

Im nächsten Schritt lernen wir die Programme kennen, die KNOPPIX zu bieten hat. Klicken Sie hierfür auf das Symbol in der linken unteren Ecke. Daraufhin öffnet sich ein Auswahlm Menü, das die unter KNOPPIX

verfügbaren Programme anzeigt. Hier können Sie nun einmal die verschiedenen Optionen ausprobieren. In Abbildung 5 sind beispielsweise die Angebote für Büro-Software zu sehen. Darüber hinaus gibt es auch verschiedene Spiele, Entwicklungswerkzeuge, Grafik-Programme, Internetanwendungen und vieles mehr.

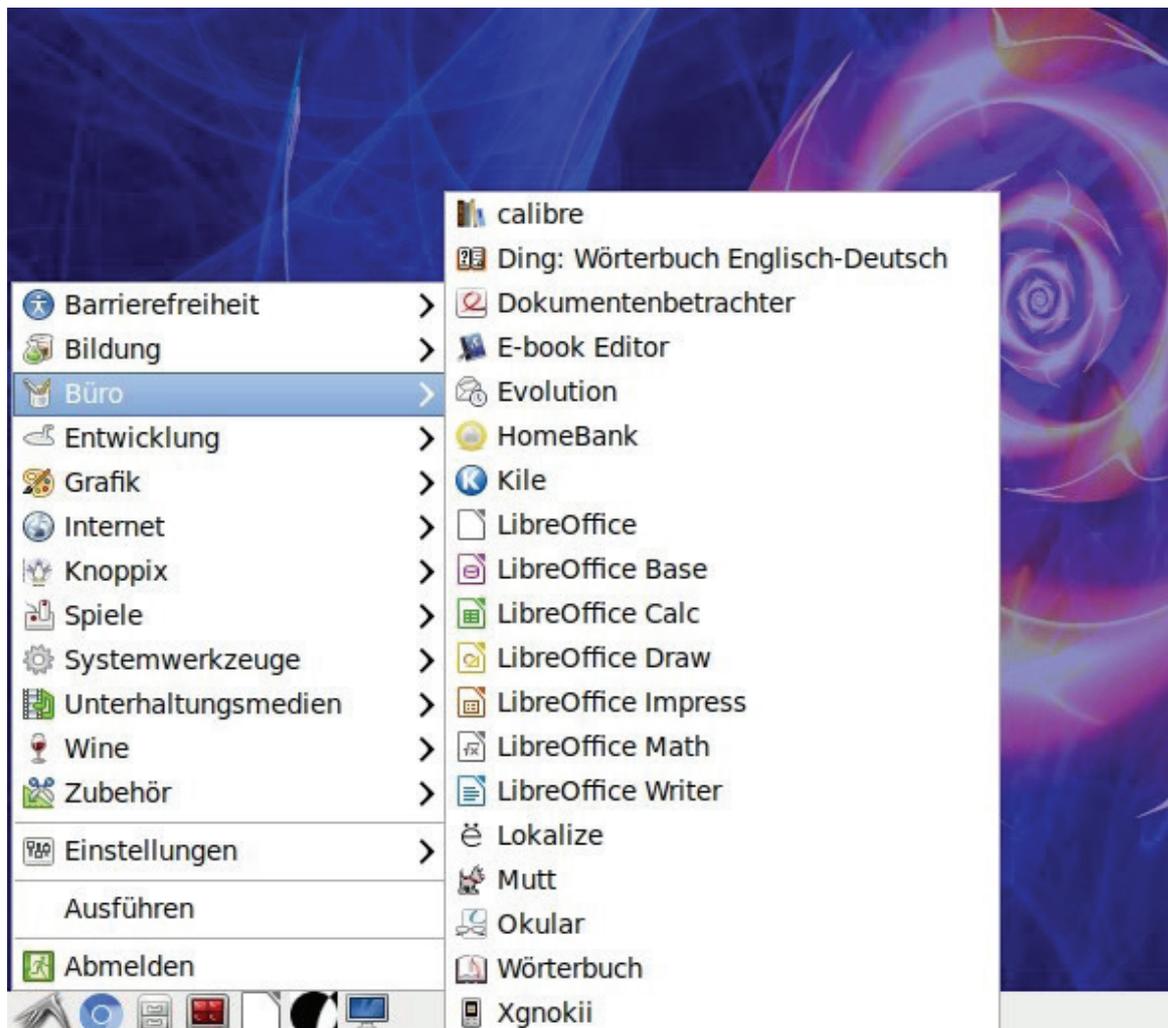


Abb. 1.5 Vielfältige Programme unter Linux

Das zeigt bereits einen der wesentlichen Vorteile bei der Verwendung von Linux. Hierbei stehen Ihnen unzählige Programme bereits vorinstalliert zur Auswahl. Diese können Sie vollkommen kostenlos nutzen. Nicht einmal der Aufwand für die Installation ist notwendig.

Zum Abschluss dieses ersten kleinen Rundgangs klicken wir in diesem Menü einmal auf „Systemwerkzeuge“ und anschließend auf „Terminal“.

Daraufhin öffnet sich das Programm, das in Abbildung 6 zu sehen ist. Hierbei handelt es sich um eine Bash-Shell. Die Shell stellt eines der wesentlichen Steuerungswerkzeuge unter Linux dar. Die Funktionsweise wird an dieser Stelle noch nicht genauer erklärt. Das folgt noch im weiteren Verlauf dieses Buchs. Doch können Sie sich bereits einmal mit diesem Werkzeug vertraut machen, da es später noch häufig zum Einsatz kommen wird.

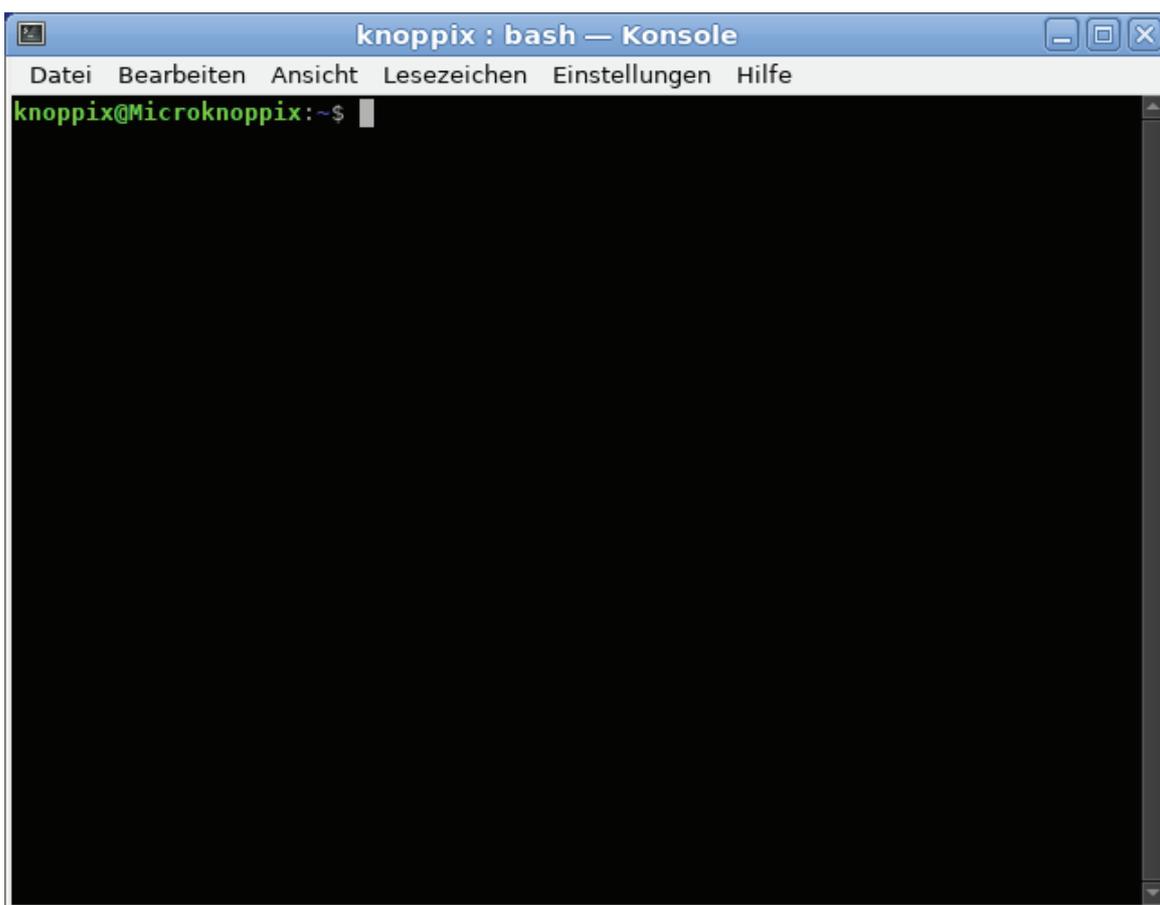


Abb. 1.6 Die Oberfläche der Bash-Shell

1.6 Die Linux-Community: So entsteht das Betriebssystem

Zum Abschluss des Einleitungs-Kapitels soll noch ein kurzer Blick darauf geworfen werden, wie die Entwicklung von Linux abläuft. Wie bei fast jeder anderen Software auch, kommt es hierbei zu einer ständigen Weiterentwicklung. Das ist nicht nur notwendig, um das Betriebssystem immer wieder an den aktuellen Stand der Technik anzupassen.

Darüber hinaus ist dies erforderlich, um die Funktionen zu verbessern und um Fehler zu beseitigen.

Bei dieser Betrachtung ist es wichtig, zwischen dem Linux-Kernel und den einzelnen Linux-Distributionen zu unterscheiden. Es wurde bereits dargestellt, dass ein Betriebssystem in zwei Bereiche aufgeteilt ist: in den Kernel, der die zentralen Funktionen des Computers steuert, und in die einzelnen Distributionen, die eine Oberfläche für die Anwender und die *Systemprogramme* bereitstellen. Während die Entwicklung des Kernels einen einheitlichen Prozess darstellt, der Auswirkungen auf alle Ausführungen von Linux hat, handelt es sich bei der Entwicklung der Distributionen um unabhängige Projekte.

Obwohl es sich beim Linux-Kernel um ein einheitliches System handelt, gibt es für die Entwicklung kaum feste Strukturen. Jeder Programmierer, der daran Interesse hat, kann hier mitwirken. Die Kommunikation der Linux-Community findet größtenteils per E-Mail statt. Diese Form wird insbesondere von Linus Torvalds unterstützt, da sie - seiner Meinung nach - verhindert, dass unterschiedliche Meinungen direkt aufeinanderprallen. Das soll dazu beitragen, Konflikte in der Entwicklungsgemeinschaft zu vermeiden. Die Mailing-Listen stehen jedem offen. Hier ist es möglich, eigene Vorschläge einzubringen und die Ideen anderer Entwickler zu bewerten. Darüber hinaus ist es problemlos möglich, selbst an der Entwicklung des Linux-Kernels mitzuwirken. Jeder Programmierer kann eigene Programme einfügen, um den Kernel zu verbessern. Bei der Linux-Community handelt es sich um eine große Gemeinschaft, an der viele Programmierer aus aller Welt teilnehmen. Diese erhalten für ihre Tätigkeit keinerlei Bezahlung. Das ist eine wichtige Voraussetzung dafür, dass das Betriebssystem kostenlos zur Verfügung steht.

Aufgrund der großen Gemeinschaft kommt es zu vielfältigen Verbesserungsvorschlägen. Zahlreiche Entwickler reichen Programmteile ein, die in den Linux-Kernel aufgenommen werden sollen. Da nicht alle Vorschläge sinnvoll sind, ist ein Entscheidungsgremium notwendig, das darüber urteilt, welche Ideen in den offiziellen Linux-Kernel aufgenommen werden. Diese Funktion übt Linus Torvalds selbst aus – mit

der Unterstützung einer kleinen Gruppe aus Programmierern, die sich bei der Entwicklung des Linux-Kernels bereits verdient gemacht haben.

Ein ganz anderes Bild ergibt sich bei den einzelnen Linux-Distributionen. Hierbei gibt es viele verschiedene Entwicklungsmodelle. In vielen Fällen sind Unternehmen für die Gestaltung zuständig, die eine kommerzielle Absicht verfolgen. Doch gibt es auch in diesem Bereich Gruppen, die unentgeltlich arbeiten und an der Verbreitung von Open-Source-Software interessiert sind. Häufig arbeiten die Entwickler der einzelnen Distributionen auch an der Gestaltung des Linux-Kernels mit. Das gibt ihnen die Möglichkeit, die Funktionen umzusetzen, die sie für ihre jeweilige Distribution für wichtig erachten. Oftmals kommt es aber auch vor, dass diese Zusätze nicht ihren Weg in den offiziellen Linux-Kernel finden. Da es sich hierbei um Open-Source-Software handelt, kann jeder Anwender den Quellcode nach seinen Bedürfnissen anpassen. Daher kommt es immer wieder vor, dass einzelne Distributionen nicht mit dem offiziellen Linux-Kernel arbeiten, sondern leichte Abwandlungen verwenden.

1.7 Übungsaufgaben

Am Ende der einzelnen Kapitel entdecken Sie immer eine oder mehrere Übungsaufgaben. Diese dienen dazu, das erworbene Wissen zu vertiefen und zu erweitern. Bei den Kapiteln, die sich mit praktischen Anwendungen befassen, sollte es möglich sein, die Aufgabe mit dem im jeweiligen Kapitel erworbenen Wissen zu lösen. In Kapiteln, die theoretische Inhalte oder Grundlagenwissen vermitteln, handelt es sich hingegen um weiterführende Inhalte, die nicht im Text zur Sprache kommen. Die Antworten können Sie dann über eine kurze Internetrecherche finden.

Dabei ist jeweils eine kurze Musterlösung angegeben. Das bedeutet jedoch nicht, dass Ihre Antwort genau gleich aussehen muss. Häufig führen in der Informatik mehrere Lösungswege ans Ziel. Die Musterlösung dient lediglich zum Abgleich mit Ihrer eigenen Lösung. Für das erste Kapitel sollen Sie die folgende Aufgabe erledigen:

Ermitteln Sie die wesentlichen Gründe, die für die Nutzung von Linux sprechen.

Musterlösung:

Unter anderem sprechen die folgenden Gründe für die Verwendung von Linux:

- ▶ Kostenfreie Nutzung möglich
- ▶ Quellenoffene Software – optimal für Lernzwecke geeignet
- ▶ Zahlreiche Entwicklungswerkzeuge für Computerprogramme enthalten
- ▶ Hohe Sicherheit
- ▶ Auch für Computer mit geringer Hardware-Ausstattung geeignet
- ▶ Dauerhaft aktuelles Betriebssystem durch ständige Weiterentwicklung
- ▶ Betriebssystem mit guter Performance
- ▶ Einfache Automatisierbarkeit von Routinen, um wiederkehrende Aufgaben automatisch ausführen zu lassen

Alle Programmcodes aus diesem Buch sind als PDF zum Download verfügbar. Dadurch müssen Sie sie nicht abtippen:

<https://bmu-verlag.de/books/linux/>



Außerdem erhalten Sie die eBook Ausgabe zum Buch im PDF Format kostenlos auf unserer Website:



<https://bmu-verlag.de/books/linux/>
Downloadcode: siehe Kapitel 11