

Dieses Buch erlaubt Ihnen, Ihre Fähigkeiten auf der Linux-Kommandozeile weiterzuentwickeln und auszubauen, damit Sie schneller, geschickter und effizienter arbeiten können.

Wenn Sie so sind wie die meisten Linux-Anwender, dann haben Sie die ersten Schritte mit der Kommandozeile während der Arbeit gemacht. Vielleicht haben Sie auch ein Buch zur Einführung gelesen oder Dinge während der Installation von Linux einfach ausprobiert. Ich schrieb dieses Buch, um Ihnen dabei zu helfen, weiterzugehen und das Niveau Ihrer Kenntnisse und Fertigkeiten auf der Linux-Kommandozeile zu verbessern. Das Buch ist voller Techniken und Konzepte, die hoffentlich Ihren Umgang mit Linux umkrepeln und Ihre Produktivität ankurbeln. Es führt Sie also über die reinen Grundlagen hinaus.

Eine Kommandozeile ist die einfachste aller Benutzerschnittstellen, sie stellt aber auch die größte Herausforderung dar. Sie ist einfach, weil sie Ihnen nichts weiter präsentiert als einen Prompt, der darauf wartet, dass Sie irgendeinen Befehl ausführen:¹

```
$
```

Sie ist eine Herausforderung, weil alles, was nach dem Prompt kommt, Ihrer Verantwortung unterliegt. Es gibt keine freundlichen Icons, Buttons oder Menüs, die Sie führen und leiten. Stattdessen ist jeder Befehl, den Sie eintippen, ein kreativer Akt. Das gilt für einfache Befehle wie das Auflisten von Dateien:

```
$ ls
```

genauso wie für aufwendigere Befehle wie diesen hier:

```
$ paste <(echo {1..10}.jpg | sed 's/ /\n/g') \  
        <(echo {0..9}.jpg | sed 's/ /\n/g') \  
        | sed 's/^/mv /' \  
        | bash
```

¹ In diesem Buch wird der Linux-Prompt (oder die Eingabeaufforderung, wie man den Prompt auch nennt) als Dollarzeichen dargestellt. Ihr Prompt kann auch anders aussehen.

Falls Sie gerade mit offenem Mund auf diesen Befehl starren und denken: »Was zum Henker ist *das?*« oder »Ich würde niemals so einen komplexen Befehl benutzen«, dann ist dieses Buch genau richtig für Sie.²

Was Sie lernen werden

Dieses Buch wird Sie bei drei wichtigen Fertigkeiten schneller und effektiver machen:

- Auswählen oder Konstruieren von Befehlen zum Lösen eines vorliegenden geschäftlichen Problems.
- Effizientes Ausführen dieser Befehle.
- Navigieren durch das Linux-Dateisystem.

Am Ende werden Sie verstehen, was hinter den Kulissen passiert, wenn Sie einen Befehl ausführen, sodass Sie die Ergebnisse besser vorhersagen können (und keine abergläubischen Vorstellungen entwickeln). Sie werden ein Dutzend unterschiedlicher Methoden sehen, um Befehle aufzurufen, und lernen, wann Sie welche von ihnen am besten einsetzen. Sie werden außerdem praktische Tipps erhalten und Tricks kennenlernen, mit denen Sie Ihre Produktivität steigern können, wie etwa:

- das schrittweise Aufbauen komplexer Befehle aus einfachen, um echte Probleme zu lösen, wie etwa das Verwalten von Passwörtern oder das Generieren von 10.000 Testdateien,
- das intelligente Organisieren Ihres Home-Verzeichnisses, damit Sie Zeit sparen und nicht lange nach Dateien suchen müssen,
- das Umwandeln von Textdateien und Abfragen dieser Dateien, wie etwa Datenbanken, um geschäftliche Ziele zu erreichen,
- das Kontrollieren der Point-and-click-Funktionen von Linux von der Kommandozeile aus, wie etwa das Kopieren und Einfügen mithilfe der Zwischenablage und das Beziehen und Verarbeiten von Webdaten, ohne die Hände von der Tastatur nehmen zu müssen.

Vor allem lernen Sie ganz allgemein die besten Vorgehensweisen, damit Sie ganz unabhängig von den Befehlen, die Sie einsetzen, erfolgreicher beim täglichen Einsatz von Linux werden und auf dem Arbeitsmarkt konkurrenzfähiger sind. Dies ist das Buch, das ich mir gewünscht hätte, als ich Linux gelernt habe.

Was dieses Buch nicht ist

Dieses Buch dient nicht der Optimierung Ihres Linux-Computers, damit dieser effizienter läuft. Es macht *Sie* effizienter im Umgang mit Linux.

² Sie lernen den Zweck dieses mysteriösen Befehls in Kapitel 8 kennen.

Dieses Buch ist auch keine allumfassende Referenz für die Kommandozeile – es gibt Hunderte von Befehlen und Funktionen, die ich nicht erwähne. Hier geht es um Fachwissen. Das Buch lehrt eine sorgfältig ausgewählte Menge an Kommandozeilenwissen in einer praktischen Anordnung und Reihenfolge, um Ihre Fertigkeiten auszubauen. Falls Sie eine Art Referenz wünschen, probieren Sie es mit meinem Taschenbuch »Linux – die wichtigen Befehle kurz & gut« (O'Reilly).

Zielgruppe und Voraussetzungen

Ich gehe davon aus, dass Sie Erfahrungen mit Linux haben; *dieses Buch ist keine Einführung*. Stattdessen ist es für Benutzerinnen und Benutzer gedacht, die ihre Kenntnisse und Fertigkeiten auf der Kommandozeile verbessern wollen, wie Studenten, Systemadministratoren, Softwareentwickler, Sicherheitsverantwortliche, Testingenieure und ganz allgemein Linux-Enthusiasten. Auch erfahrene Linux-Anwender dürften hier noch fündig werden, vor allem wenn sie den Umgang mit Befehlen eher durch Ausprobieren gelernt haben und ihr Verständnis von den dahinterliegenden Konzepten verbessern wollen.

Um wirklich einen Nutzen aus diesem Buch zu ziehen, sollten Sie bereits mit den folgenden Themen vertraut sein (falls Sie es nicht sind, finden Sie in Anhang A Informationen zum Auffrischen Ihrer Erinnerung):

- Anlegen und Bearbeiten von Textdateien mit einem Texteditor wie `vim (vi)`, `emacs`, `nano` oder `pico`.
- Grundlegende Befehle zum Umgang mit Dateien, wie `cp` (Kopieren), `mv` (Verschieben oder Umbenennen), `rm` (Entfernen oder Löschen) und `chmod` (Ändern der Dateiberechtigungen).
- Grundlegende Befehle zum Betrachten von Dateien, wie `cat` (Betrachten einer ganzen Datei) und `less` (seitenweises Betrachten).
- Grundlegende Verzeichnisbefehle, wie `cd` (Wechseln des Verzeichnisses), `ls` (Auflisten der Dateien in einem Verzeichnis), `mkdir` (Anlegen eines Verzeichnisses), `rmdir` (Entfernen eines Verzeichnisses) und `pwd` (Anzeigen Ihres aktuellen Verzeichnisnamens).
- Grundlagen zu Shell-Skripten: Speichern von Linux-Befehlen in einer Datei, Ausführbarmachen einer Datei (mit `chmod 755` oder `chmod +x`) und Ausführen der Datei.
- Betrachten der in Linux enthaltenen Dokumentation, der sogenannten Manpages, mit dem Befehl `man` (Beispiel: `man cat` zeigt die Dokumentation des Befehls `cat` an).
- Superuser werden mithilfe des Befehls `sudo`, sodass Sie vollständigen Zugriff auf Ihr Linux-System erhalten (Beispiel: `sudo nano /etc/hosts` bearbeitet die Systemdatei `/etc/hosts`, die vor normalen Benutzern geschützt ist).

Wenn Sie darüber hinaus auch noch gebräuchliche Kommandozeilenfunktionen wie das Pattern Matching (Musterabgleich) für Dateinamen (mit den Symbolen * und ?), die Umleitung von Eingaben/Ausgaben (< und >) sowie Pipes (|) kennen, sind Sie gut für alles Weitere gerüstet.

Ihre Shell

Ich gehe davon aus, dass Ihre Linux-Shell `bash` ist, die Standard-Shell in den meisten Linux-Distributionen. Immer wenn ich von der »Shell« schreibe, meine ich `bash`. Die meisten Ideen, die ich vorstelle, gelten auch für andere Shells, wie `zsh` oder `dash`; in Anhang B finden Sie Unterstützung beim Übersetzen der Beispiele aus diesem Buch für andere Shells. Ein Großteil des Materials funktioniert auch problemlos und unverändert im Terminal des Apple Mac, das standardmäßig `zsh` einsetzt, obwohl man auch `bash` ausführen kann.³

Konventionen in diesem Buch

Folgende typografische Konventionen kommen in diesem Buch zum Einsatz:

Kursiv

Kennzeichnet neue Begriffe, URLs, E-Mail-Adressen, Dateinamen und Dateierweiterungen.

Nichtproportionalschrift

Wird für Programmlistings verwendet und kennzeichnet beim Einsatz innerhalb eines Absatzes Programmelemente wie Variablen- oder Funktionsnamen, Datenbanken, Datentypen, Umgebungsvariablen, Anweisungen und Schlüsselwörter.

Nichtproportionalschrift fett

Zeigt Befehle oder anderen Text, der wörtlich vom Anwender eingetippt werden sollte. Wird außerdem gelegentlich in Befehlsausgaben eingesetzt, um interessante Stellen hervorzuheben.

Nichtproportionalschrift kursiv

Kennzeichnet Text, der durch nutzereigene Werte oder Werte, die sich aus dem Kontext ergeben, ersetzt werden soll. Wird außerdem für kurze Hinweise rechts von Codelistings verwendet.

Nichtproportionalschrift hervorgehoben

Dient in komplexen Programmlistings dazu, die Aufmerksamkeit auf bestimmte Textteile zu lenken.

³ Die `bash`-Version auf macOS ist uralt, und ihr fehlen wichtige Funktionen. Um die `bash` zu aktualisieren, lesen Sie den Artikel »Upgrading Bash on macOS« (<https://oreil.ly/35jux>) von Daniel Weibel.



Tipp

Dieses Element kennzeichnet einen Tipp oder einen Vorschlag.



Hinweis

Dieses Element kennzeichnet einen allgemeinen Hinweis.



Warnung

Dieses Element kennzeichnet eine Warnung.

Die Verwendung von Codebeispielen

Es steht zusätzliches Material (Codebeispiele, Übungen usw.) zum Herunterladen unter <https://efficientlinux.com/examples> zur Verfügung.

Falls Sie eine technische Frage oder ein Problem bei der Verwendung der Codebeispiele haben, schicken Sie bitte eine E-Mail an bookquestions@oreilly.com.

Dieses Buch soll Ihnen helfen, Ihre Arbeit zu erledigen. Falls ein Beispielcode in diesem Buch angeboten wird, dürfen Sie ihn im Allgemeinen in Ihren Programmen und Dokumentationen nutzen. Sie müssen nicht extra Kontakt mit uns aufnehmen und um Erlaubnis bitten, es sei denn, Sie reproduzieren einen beträchtlichen Teil des Codes. So erfordert zum Beispiel das Schreiben eines Programms, das einige Codeschnipsel aus diesem Buch verwendet, keine Erlaubnis. Wollen Sie dagegen Beispiele aus O'Reilly-Büchern verkaufen oder vertreiben, brauchen Sie eine Erlaubnis. Das Beantworten einer Frage, indem Sie dieses Buch und Beispielcode daraus zitieren, erfordert keine Erlaubnis. Nehmen Sie dagegen einen beträchtlichen Teil des Beispielcodes aus diesem Buch in die Dokumentation Ihres Produkts auf, ist eine Erlaubnis notwendig.

Wir wissen eine Nennung zu schätzen, verlangen sie aber nicht. Eine Nennung umfasst üblicherweise Titel, Autor, Verlag und ISBN, zum Beispiel: »*Efficient Linux at the Command Line* von Daniel J. Barrett (O'Reilly). Copyright 2022 Daniel Barrett, 978-1-098-11340-7«.


Falls Sie der Meinung sind, dass Ihre Nutzung der Codebeispiele außerhalb des Bereichs des fairen Einsatzes liegt, den wir oben beschrieben haben, schreiben Sie uns an permissions@oreilly.com.

Danksagungen

Es war eine Freude, dieses Buch zu schreiben. Mein Dank geht an die fantastischen Leute bei O'Reilly, vor allem die Lektoren Virginia Wilson und John Devins, die Herstellungsredakteure Caitlin Ghegan und Gregory Hyman, die Content-Managerin Kristen Brown, die Korrektorin Kim Wimpsett, die Indexredakteurin Sue Klefstad und das überaus hilfsbereite Tools-Team. Danken möchte ich außerdem den Fachkorrektoren dieses Buchs, Paul Bayer, John Bonesio, Dan Ritter und Carla Schroder, für ihre vielen hilfreichen Kommentare und Anmerkungen. Danke außerdem an die Boston Linux Users Group für die Titelvorschläge. Ein besonderer Dank geht an Maggie Johnson bei Google für ihre freundliche Erlaubnis, dieses Buch schreiben zu dürfen.

Meinen tief empfundenen Dank richte ich darüber hinaus an Chip Andrews, Matthew Diaz und Robert Strandh, mit denen ich vor 35 Jahren an der The Johns Hopkins University studiert habe. Sie haben mein neu erwachtes und wachsendes Interesse an Unix erkannt und dem Informatikfachbereich zu meinem großen Erstaunen empfohlen, mich als seinen nächsten Systemadministrator anzuheuern. Dieser kleine Akt des Vertrauens hat den weiteren Verlauf meines Lebens verändert. (Robert gebührt außerdem Dank für den Tipp zum Blindschreiben in Kapitel 3.) Danke auch an die Schöpfer und Hüter von Linux, GNU Emacs, Git, AsciiDoc und vielen anderen Open-Source-Werkzeugen – ohne diese klugen und großzügigen Menschen sähe meine Karriere heute vermutlich ganz anders aus.

Und natürlich danke ich an dieser Stelle meiner wunderbaren Familie, Lisa und Sophia, für ihre Liebe und Geduld.

Diese Leseprobe haben Sie beim
 [edv-buchversand.de](https://www.edv-buchversand.de) heruntergeladen.
Das Buch können Sie online in unserem
Shop bestellen.

[Hier zum Shop](#)