

Titel: Praxiskurs Unix-Shell
Autor: Martin Dietze
Preis: 19,90
ISBN: 978-3-89721-565-8

1 Allgemeine Bemerkungen

Das Buch ist richtet sich an Menschen, die noch keinerlei Erfahrung mit der Unix-Shell haben. Fortgeschrittene Benutzer werden hingegen wenig Freude an diesem Buch aus der O'Reilly-Reihe „basics“ finden.

2 Gliederung des Buches

Das Buch ist thematisch in zehn Kapitel gegliedert. Das elfte Kapitel besteht aus Lösungen zu den Übungen, wie sie in jedem Kapitel zu finden sind. Im zwölften Kapitel erzählt der Autor abschließend zwei Geschichten aus seiner Studentenzeit, die mit der Shell zu tun haben. Im dreizehnten Kapitel sind die behandelten Befehle, Variablen und Operatoren aufgelistet.

2.1 Kapitel 1: Grundlagen

Im Kapitel über Grundlagen wird zunächst auf die geschichtliche Entstehung der Unix-Shell und ihre Aufspaltung in verschiedene Shell-Familien wie *Bourne-Shell*, *Korn-Shell* und *Bourne Again Shell* eingegangen.

2.2 Kapitel 2: Erste Schritte

Im zweiten Kapitel wird beschrieben, wie man die Shell startet und wie man sie beendet. Ferner lernt man verschiedene Funktionen kennen, wie z.B. die History-Funktion.

Ein wichtiges Bedienungselement der Shell sind z.B. *Tastenkürzel*, die in Anlehnung an bekannte Editoren wie Emacs- und vi als Tastenbelegungen in der Art eines Kommandozeileneditors realisiert sind. In diesem Kapitel werden sie kurz beschrieben.

Weiter wird erklärt, wie man mit den Werkzeugen der Shell auf Dateien und Verzeichnisse im Dateiensystem zugreifen kann. Wir lernen die Struktur der Notation von Dateien, Verzeichnissen und Pfaden.

Zum Schluss des Kapitels stellt der Autor das Hilfesystem *Online-manuals* (*Manpages*) kurz vor und erklärt, wie es zu benutzen ist.

2.3 Kapitel 3: Die Arbeit mit Dateien

Im dritten Kapitel geht es um Dateien unterschiedlichster Art. Wir erfahren, dass sogar für Geräte wie Drucker im Unix-System eine Datei erstellt wird.

Dateien werden nach Typen klassifiziert und von der Shell erkannt und unterschiedlich dargestellt.

Der Zugriff auf Dateien kann über Links (Zweitnamen) erfolgen, die wiederum in normale und *symbolische Links* unterteilt werden. Der Unterschied zwischen diesen beiden Link-Arten wird ausführlich erklärt.

Ein weiteres grosses Thema des Kapitels stellt das Thema der Zugriffsrechte und deren Verwaltung dar. Hiermit eng verknüpft ist das Thema der Benutzer- und Gruppenverwaltung, das ebenfalls anschließend ausführlich behandelt wird.

Weiter im Kapitel erfahren wir, wie man Dateien kopieren, löschen, verschieben und umbenennen kann.

Gesondert geht der Autor auf das Verwenden von sog. *Wildcards* und Quotings in der Shell ein.

2.4 Kapitel 4: Befehle kombinieren und erweitern

Im vierten Kapitel geht darum, wie Befehle miteinander kombiniert verwendet werden können. Der Autor behandelt in diesem Kapitel Operatoren, die für die Ein- und Ausgabe verantwortlich sind.

Dazu gehören die sog. *Umlenkungsoperatoren*, die ausführlich behandelt werden.

Mit Hilfe von sog. *Pipes* werden kombinierte Befehle ermöglicht, in denen durch diese Pipes Prozesse, bzw. deren Ein- und Ausgabe, miteinander verbunden sind.

Ferner geht der Autor darauf ein, dass die Shell auch Variablen kennt, was wir sonst nur aus dem Gebiet der Programmierung kennen.

Wir lernen den Begriff des *Alias* kennen, mit dem man ein Pseudokommando erzeugen kann, um z.B. einen kombinierten Befehle zu vereinfachen.

Weiter erklärt der Autor wie man Prozesse unterbricht oder stoppt.

2.5 Kapitel 5: Die Arbeitsumgebung anpassen

Im fünften Kapitel lernen wir unsere Arbeitsumgebung anzupassen. Dazu benötigt man einen Zugang zur Konfigurationsdatei, die meistens in *etc* liegt.

Die Einstellungen für die Shell werden über *Umgebungsvariablen* gesteuert. Die folgenden Variablen werden kurz behandelt: *PATH*, *LD_LIBRARY_PATH*, *MANPATH*, *PAGER*.

Weiter wird erklärt, wie man eine Eingabeaufforderung einrichtet und weitere Einstellungen wie z.B. Standardrechte für neue Dateien vergibt oder einen Willkommensspruch für jede Shell-Sitzung einrichtet.

2.6 Kapitel 6: Daten verarbeiten

In diesem Kapitel geht es um Verarbeitung von Daten mit Hilfe von Filtern, Regulären Ausdrücken oder Operatoren.

Der Autor stellt weitere Hilfsprogramme vor, die in der Shell aufgerufen werden, um z.B. Daten, die komprimiert aus dem Netz gezogen wurden zu entpacken.

Es werden die Kommandos *zip*, *unzip* und *tar* vorgestellt.

2.7 Kapitel 7: Arbeiten im Netzwerk

Im siebten Kapitel werden Shell-Werkzeuge vorgestellt, die für das Arbeiten im Netzwerk von Bedeutung sind.

Zu Anfang wird kurz erklärt, wie das Netzwerkprotokoll funktioniert und um welche Protokolle es sich hierbei im Einzelnen handelt.

Wir erfahren, wie man eine interaktive verschlüsselte Sitzung via *ssh* aufbaut.

2.8 Kapitel 8: Shell-Skripte-Grundlagen

In diesem Kapitel lernen wir die Shell als Programmiersprache kennen. Der Autor führt als erstes Beispiel für ein Shell-Skript ein „*Hallo Welt*“-Programm ein.

Wie bei einer Skript-Sprache dürfen auch bei der Shell die Kommandozeilenargumente nicht fehlen, die der Autor kurz vorstellt.

Als nächstes werden Verzweigungen besprochen, die mit Hilfe von *if* und *else* ausgedrückt werden.

Es folgt die Erörterung von mathematischen Funktionen und Schleifen (*for-Schleife*, *while-Schleife* und *until-Schleife*).

2.9 Kapitel 9: Shell-Skripte für Fortgeschrittene

In diesem Kapitel werden weitere Elemente für Shell-Skripte vorgestellt, z.B. wie man sog. *Inline-Dokumente* erstellen kann.

Es folgen Abschnitte zum trapping von Signalen, der dynamischen Erzeugung von Programmtext (mittels *eval*), die Auswertung von Kommandozwischenschaltern (*getopts*), zur Definition von Funktionen und zur Einbindung von Shell-Skripten in Cron-Jobs und in die System-V-Init-Prozedur.

Als praktische Beispiele wird ein Beispiel für Log-Rotation programmiert sowie eine Verzeichnishierarchie geklont.

Als weiterer wichtiger Punkt wird in diesem Kapitel die Fehlerbehandlung vorgestellt.

Am Ende des Kapitels wird diskutiert, welche Alternativen es zu Shell-Skripten gibt.

2.10 Kapitel 10: Der Unix-Systemeditor vi

In diesem Kapitel wird der Systemeditor *vi* vorgestellt.

3 Kritik

3.1 Tippfehler

Seite 2, Absatz 2, Zeile 4: *Unter anderen ist dieser ...* muss geändert werden in *anderem*.

Seite 3, Absatz 2, Zeile 4: *Sie soll unter anderen sicherstellen, ...* muss geändert werden in *anderem*.

Seite 8, Absatz 1, Zeile 1: *unter der Prompt* muss geändert werden in *dem*.

Seite 38, *Unterteilung der Manuals*, Absatz 3, Zeile 2: *unter es Umständen* muss gelöscht werden *es*.

Seite 65, *cp: (change directory)* muss geändert werden in *copy*.

Seite 84, Absatz 4: Es fehlt ein Hilfsverb im Satz: *Deshalb man ihm noch mitteilen, welche Art von spezieller Datei er anlegen soll...*

Seite 104f: Nach allen Sätzen der Struktur mit der Nummer X wird ...muss nach X entweder ein Punkt oder ein Komma stehen, damit dem Anfänger klar wird, dass nicht etwa durch die Nummer die nachfolgende Aktion ausgelöst wird, sondern dass es sich bei der Nummer um die Nummer des Signals handelt, das unter den nachfolgenden Bedingungen erzeugt wird.

Seite 105: *SIGTSTP* muss geändert werden in *SIGSTP*.

Seite 169, Absatz 2, Zeile 4: *einfache Art darauf ...* muss geändert werden in *darauf*.

Seite 185, 188, 189: Im Beispiel mit der Zeile `<div align = „center„>` fehlt „

Seite 207: Beim Shellskript fehlt im help-String eine schließende eckige Klammer nach den Schalteroptionen.

Seite 233, doppelter Eintrag muss gelöscht werden: *'Key Springt zu der Stelle mit der Marke Key*.

4 Fazit:

Das Buch von Martin Dietze „Praxiskurs Unix-Shell“, aus der Reihe *basics* ist ein guter Einstieg in die Unix-Shell-Welt, für alle diejenigen, die hierbei Neuland betreten. Die dargebotene Theorie wird mit Übungen verfestigt, die in jedem Kapitel nach wichtigen Themen vorkommen. Auch für sporadische Shell-Nutzer ist dieses Buch sowohl als systematische Einleitung als auch für die Auffrischung von Kenntnissen gut geeignet. Das Buch zeichnet sich vor allem durch seine Beschränkung auf wichtige Themengebiete aus und geht kaum in die Tiefe, was bei der komplexen Materie für einen Anfänger ein großer Vorteil ist. Shell-Gurus oder solche, die es werden wollen, sind hingegen mit einem solchen Buch eher schlecht beraten.