



Betriebssysteme und Sicherheit, WS 2021/22

## 1. Aufgabenblatt – Unix I

Bearbeitungszeitraum: 18.10.2021 – 22.10.2021

In dieser Übung soll allen Studierende, die bisher noch keine Erfahrungen mit Unix-artigen Systemen – wie dem im Rechenzentrum bereitgestellten „Ubuntu“ – gesammelt haben, den Umgang mit einem solchen näher bringen. Die Beantwortung der konkreten Fragen steht dabei nicht im Vordergrund, sie sind vielmehr als Anregung zu verstehen sich mit den typischen Arbeitsweisen in einem Unix-System vertraut zu machen. Insofern ist es nicht zielführend, die Fragen allein mittels Recherche im Internet zu beantworten.

Solltet ihr selber kein Unix-artiges System verwenden und auch keins direkt zur Hand haben, dann gibt es auf unserer Website eine Anleitung, um dennoch an solch ein System zu kommen: <https://tu-dresden.de/inf/os/studium/vorlesungen/bs/uebungsmaterialien>.

**Aufgabe 1.1** Diese Aufgabe macht Sie mit der Benutzung von Unix-Hilfeseiten („Manpage“, kurz für „manual page“) mit Hilfe des Kommandos `man` vertraut. Veranschaulichen Sie sich bei c) bis e) Ihre Antworten durch eigene Beispiele. Zum Nachvollziehen der Kommandozeilen-Aufrufe können Sie u. a. auch <https://explainshell.com> nutzen.

- (a) Wie kann man allgemein die Manpage zu einem Shell-Kommando aufrufen? Wie lautet speziell der Aufruf, um sich über Aufbau und Inhalt von Manpages zu informieren? Durch welche Eingabe wird die Anzeige einer Manpage beendet (und damit die Rückkehr zur Shell möglich)?
- (b) Die Unix-Manpages sind in Abschnitte eingeteilt. Welche Abschnitte gibt es, und wie greift man darauf zu?
- (c) Welche `man`-Option zeigt Kurzbeschreibungen von Manpages auf der Kommandozeile an?
- (d) Welche Option von `man` durchsucht die Kurzbeschreibungen aller im System verfügbaren Manpages nach einem bestimmten Begriff?
- (e) Benutzen Sie die Manpages der entsprechenden Kommandos, um die nachfolgenden Fragen zu beantworten:
  - Mit dem Kommando `ls` kann man sich den Inhalt eines Verzeichnisses anzeigen lassen.
    - `ls` zeigt standardmäßig versteckte Dateien (deren Namen mit einem Punkt beginnen) nicht an. Wie kann man dieses Verhalten ändern? Wie lassen sich zusätzlich alle Dateien im ausführlichen Format (d. h. mit Rechten, Größe usw.) anzeigen?
    - Wie kann man sich die Dateien der Größe nach geordnet anzeigen lassen?
    - Auf welche Weise kann man die Dateigröße in einem leichter lesbaren Format (mit Einheiten für Byte, KByte und MByte) anzeigen?
  - Welche Header-Dateien müssen eingebunden werden, um den `open()`-Systemaufruf verwenden zu können? Was sind mögliche Rückgabewerte von `open()`, und welche Fehler können beim Aufruf von `open()` auftreten?
  - Mit dem Befehl `cd` kann man das aktuelle Arbeitsverzeichnis wechseln. Warum muss `cd` durch die Shell selbst bereitgestellt werden und kann nicht als separates Programm implementiert sein? Welche zusätzlichen Optionen unterstützt `cd`?

**Aufgabe 1.2** Zum besseren Verständnis der nachfolgenden Aktivitäten können Sie sich den im Laufe der Aufgabe entstehenden Verzeichnisbaum aufzeichnen und sich vergegenwärtigen, in welchem Verzeichnis Sie sich jeweils befinden.

- (a) Legen Sie in Ihrem Home-Verzeichnis ein Verzeichnis namens `mydir` an. Wechseln Sie in dieses Verzeichnis und erzeugen Sie mittels `touch myfile` eine (leere) Datei `myfile`. Zeigen Sie die ausführlichen Informationen über diese Datei an.

*Hinweis:* Das Kommando `mkdir` erzeugt ein neues Verzeichnis, `cd` ermöglicht den Wechsel des Arbeitsverzeichnisses, und die Kommandos `ls` und `stat` zeigen Datei-/Verzeichnis-Informationen an.
- (b) Kehren Sie in Ihr Home-Verzeichnis zurück. Legen Sie ein weiteres Verzeichnis mit dem Namen `yourdir` an und informieren Sie sich über den ausführlichen, *vollständigen* Inhalt Ihres Home-Verzeichnisses.

- (c) Erzeugen Sie in dem Verzeichnis `yourdir` mit Hilfe des Kommandos `ln` einen Hardlink `yourfile`, der auf `myfile` zeigt. Zeigen Sie den ausführlichen Inhalt der Verzeichnisse `mydir` und `yourdir` an. Worin unterscheiden sich diese Inhalte? Worin unterscheidet sich der jetzige Zustand von `myfile` gegenüber dem ursprünglichen Zustand nach a)? Erklären Sie die Ergebnisse.
- (d) Mit dem Kommando `cat` und den beiden Formen der Ausgabeumlenkung `>` und `>>` ist es möglich, einzelne Wörter direkt in eine Datei zu schreiben. Die Wörter werden jeweils durch einen Zeilenumbruch getrennt, die gesamte Eingabe wird mit `Strg+D` beendet. Untersuchen Sie die Wirkung der beiden Eingabeumlenkungen, indem Sie im Verzeichnis `mydir` eine Datei `test` erzeugen und in diese Datei in der beschriebenen Weise mehrmals schreiben. Informieren Sie sich jedes Mal über den Inhalt der Datei (z. B. mit dem Kommando `cat`). Löschen Sie am Ende die Datei `test` (Kommando: `rm`).
- (e) Fügen Sie folgende Wörter abwechselnd in die Dateien `myfile` und `yourfile` ein: `thread`, `process`, `task`, `memory`, `TLB`. Was beobachten Sie beim Betrachten der Inhalte beider Dateien?
- (f) Vergewissern Sie sich, in welchem Verzeichnis Sie sich befinden; wechseln Sie ggf. in das Verzeichnis `mydir`. Suchen Sie in der Datei `myfile` nach allen Zeilen, die den Buchstaben `t` enthalten; sortieren Sie das Ergebnis alphabetisch absteigend und geben Sie die erste Zeile aus.

*Hinweis:* Das Kommando `grep` durchsucht Inhalte von Dateien, `sort` sortiert Zeilen, und das Kommando `head` beschränkt eine Liste auf eine Anzahl von Zeilen. Nutzen Sie Pipes zum Verknüpfen von Befehlen.

- (g) Welche beiden Möglichkeiten haben Sie, in einem Schritt (also durch *einmalige* Anwendung von `cd`) in das Verzeichnis `yourdir` zu wechseln? Wählen Sie eine davon aus und überzeugen Sie sich vom Erfolg. Ändern Sie Ihr eigenes Zugriffsrecht für die Datei `yourfile` auf „nur-lesend-zugreifbar“. Versuchen Sie nun, in die Datei `myfile` zu schreiben. Erklären Sie den Effekt.

*Hinweis:* Zur Änderung von Zugriffsrechten kann das Kommando `chmod` verwendet werden.

- (h) Legen Sie ein Verzeichnis `somedir` an und wechseln Sie in dieses Verzeichnis. Kopieren Sie die Datei `myfile` in dieses Verzeichnis. Überprüfen Sie den Inhalt der Datei `yourfile`.
- (i) Zeigen Sie den ausführlichen Inhalt des aktuellen Verzeichnisses an. Wieso hat der Link-Zähler der Datei `myfile` den Wert 1, obwohl es diese Datei in Ihrem Verzeichnisbaum zweimal gibt?
- (j) Löschen Sie die Datei `myfile` zunächst in `somedir` und anschließend in `mydir`. Untersuchen Sie jeweils die Eigenschaften von `yourfile` und erklären Sie die Veränderungen.

*Zusatzaufgabe:* Führen Sie die oben beschriebenen Schritte durch, aber verwenden Sie einen symbolischen Link (der gleichfalls durch `ln` erzeugt werden kann) statt eines Hardlinks.

**Aufgabe 1.3** Vollziehen Sie die in der Vorlesung vorgeführten Beispiele zur Programmentwicklung und zu Grundlagen von Unix nach. Versuchen Sie nach Möglichkeit alle damit verbundenen Vorgehensweisen, Informationen und Systemaktivitäten zu erklären.

*Hinweis:* Das vorgestellte C-Programm `hello1.c` steht im Internet bereit und kann mit folgendem Befehl in Ihr aktuelles Verzeichnis geladen werden:

```
wget https://os.inf.tu-dresden.de/Studium/Bs/hello1.c
```

**Aufgabe 1.4** Informieren Sie sich anhand der Manpages über die folgenden Systemaufrufe:

```
fork(), execve(), exit(), wait(), waitpid()
```

Machen Sie sich im Hinblick auf die 3. Unix-Übung mit Format, Wirkungsweise und wichtigen Eigenschaften dieser Systemaufrufe vertraut.