

## Systeme I: Betriebssysteme

### Übungsblatt 2 - Praxis

#### Aufgabe 1 (6 Punkte)

- a) Unter Linux stehen Ihnen eine Reihe von Texteditoren zur Auswahl, zum Beispiel `emacs`, `nano`, `vi` und `vim`. Die meisten Editoren sind in den Poolräumen in Gebäude 082 unter Linux vorinstalliert. Wählen Sie dort beim Neustart Ihres Rechners die Option `ubuntu`, um zu einem Linux Betriebssystem zu gelangen. Alternativ können Sie auf die Editoren per SSH über `login.uni-freiburg.de` (mit RZ-Account) zugreifen. Fehlende Editoren auf Ihrer persönlichen Linux-Distribution lassen sich i.d.R. mit `sudo apt-get install <Editorname>` nachträglich installieren.

Machen Sie sich mit zwei der oben genannten Editoren (`vi` und `vim` zählen als ein Editor) soweit vertraut, dass Sie mindestens folgende Funktionen verwenden können:

- 1) Öffnen einer Datei
- 2) Ändern des Dateiinhaltes
- 3) Suche nach einem Begriff
- 4) Speichern der Änderungen
- 5) Beenden des Editors mit und ohne Übernehmen der Änderungen

Schreiben Sie für die Abgabe einen Merktzettel, beispielsweise in Form einer kleinen Tabelle. Dieser Merktzettel soll erläutern, wie die oben genannten Funktionen mit den beiden von Ihnen gewählten Editoren verwendet werden können.

*Hinweise zur Benutzung der Editoren:* Für alle Editoren stehen `man`-pages bereit und jeder Editor hat eine eingebaute Hilfe-Funktion.

`emacs`: Menü „Help“ oder `Strg-h t` (Tutorial) oder `Strg-h d` (Suche in der Hilfe)  
`nano`: `Strg-g`  
`vi(m)`: `:help`

Die Bedienung von `vi(m)` ist etwas unkonventionell: `vi(m)` unterscheidet zwischen verschiedenen Modi. Die Texteingabe ist nur im Insert-Modus möglich, der durch die Eingabe von `i` erreicht und mit `ESC` wieder verlassen werden kann. Die Eingabe der meisten Kommandos (wie das Löschen eines Zeichens oder die Suche nach einem Muster) erfolgt im Command-Modus, der mit `:` aktiviert wird.

## Aufgabe 2 (1+2+2+1+1+2 Punkte)

Dateirechte werden unter Linux mit dem Befehl `chmod` gesetzt. Informieren Sie sich mit `man chmod` über die Verwendung des Befehls.

Geben Sie für die Aufgabenteile b)–d) und f) in Ihrer Abgabe die ausgeführten Befehle und deren Ausgabe an.

- a) Nehmen Sie an, Sie führen den Befehl `ls -l` aus und erhalten dabei folgendes Ergebnis:

```
$ ls -l
drwxr-x--x 2 osswald staff      4096 2013-11-04 14:01 meine_dateien
```

Beschreiben Sie in eigenen Worten, was die Zugriffsrechte des Verzeichnisses `meine_dateien` bedeuten.

- b) Benutzen Sie einen der Editoren aus Aufgabe 1, um eine Datei namens „`begruessung.sh`“ mit folgendem Inhalt im Verzeichnis `/tmp` anzulegen:

```
#!/bin/sh
echo -n "Bitte geben Sie Ihren Namen ein: "
read NAME
echo "Hallo $NAME."
```

- Wenn Sie nun versuchen, die Datei durch die Eingabe von „`./begruessung.sh`“ auszuführen, sollten Sie eine Fehlermeldung ähnlich „`Permission denied.`“ zu sehen bekommen. Ändern Sie die Zugriffsrechte der Datei „`begruessung.sh`“ so ab, dass Sie selbst (und niemand sonst) das Recht besitzen, die Datei auszuführen.
  - Ändern Sie nun die Zugriffsrechte der Datei „`begruessung.sh`“ so, dass sowohl Sie als auch Ihr Subgruppen-Partner<sup>1</sup> (aber kein Dritter, der nicht zur Subgruppe gehört) mit seinem Account die Datei ausführen kann.
- c) Erstellen Sie im Ordner `/tmp` eine Kopie des Systemprogramms `whoami` mit folgenden Befehlen:

```
cd /tmp
cp /usr/bin/whoami werbinich
```

Sie können das kopierte Programm dann mit „`./werbinich`“ ausführen.

- Vergleichen Sie mithilfe von „`ls -l`“ die Eigenschaften des Originalprogramms „`/usr/bin/whoami`“ mit Ihrer Kopie „`werbinich`“. Welche Eigenschaften haben sich geändert?

---

<sup>1</sup>Falls Sie keinen Subgruppenpartner haben, fragen Sie einen beliebigen anderen Studenten. Damit Sie die geänderten Rechte testen können, muss sich Ihr Subgruppen-Partner am selben Rechner einloggen.

- Ändern Sie bei Bedarf die Attribute der Datei „`werbinich`“ so, dass auch Ihr Subgruppen-Partner mit seinem Account die Datei ausführen kann. Welcher Username wird Ihrem Subgruppenpartner angezeigt, wenn er „`./werbinich`“ ausführt?
- Wie können Sie erreichen, dass Ihr Subgruppenpartner bei der Ausführung nicht seinen Login, sondern den Login des Besitzers der Datei als Ausgabe erhält?  
*Hinweis: `whoami` gibt die Benutzerkennung aus, unter der das Programm läuft.*

d) Führen Sie folgende Befehle aus, um eine Datei `loesche_mich.txt` im Verzeichnis `mein_verzeichnis` zu erstellen:

```
mkdir mein_verzeichnis
touch mein_verzeichnis/loesche_mich.txt
chmod a=r mein_verzeichnis/loesche_mich.txt
chmod a=rx mein_verzeichnis
```

Sowohl das Verzeichnis als auch die Datei sind nun schreibgeschützt. Welche Rechte müssen Sie mindestens zur Datei oder zum Verzeichnis hinzufügen, um die Datei löschen zu können? Seien Sie bei der Rechtevergabe möglichst restriktiv.

e) Erstellen Sie mit den folgenden Befehlen die Textdatei `readmeifyoucan.txt` und modifizieren Sie die Zugriffsrechte:

```
echo "You can." > readmeifyoucan.txt
cat readmeifyoucan.txt
chmod u-rwx readmeifyoucan.txt
cat readmeifyoucan.txt
```

Wie lautet die Ausgabe beim zweiten Zugriff auf die Datei mit `cat`? Würde sich diese Ausgabe unterscheiden, wenn die Datei einem anderen Benutzer gehörte? Erläutern Sie, wie es zu diesem Verhalten kommt.

f) Wechseln Sie mit `cd` in Ihr Home-Verzeichnis (`/home/benutzername`). Legen Sie dort mit dem Befehl `mkdir systeme-public` ein neues Unterverzeichnis an.

- Setzen Sie die Rechte von `systeme-public` so, dass jeder Benutzer lesend auf den Inhalt zugreifen und in den Ordner navigieren kann.
- Was müssen Sie nun tun, damit immer noch jeder auf `systeme-public` zugreifen kann, aber aus Gründen der Sicherheit der Inhalt des übergeordneten Homes für alle Benutzer außer dem Besitzer nicht mehr aufgelistet werden kann?

Bei Benutzer `osswald` würde dies so aussehen: Der Inhalt von `/home/osswald` ist für andere Benutzer nicht sichtbar, ein `ls /home/osswald/systeme-public` zeigt aber trotzdem alle vorhandenen Dateien des Unterordners an.

**Abgabe: Als PDF-Datei über Ilias bis 9. November 2015, 23:59:00 Uhr.**