



Praktikum 1

Linux-Shell – Aufgaben

Zu bearbeiten bis zum 25. April 2023

Cybersecurity im Sommersemester 2023

Jan Wichelmann, Anja Köhl

Einführung

In diesem Praktikum soll es darum gehen, sich mit der Benutzung des Praktikumssystems und der Linux-Shell vertraut zu machen, damit Sie in den kommenden Wochen sicher damit arbeiten können. Gleichzeitig ist dies für uns und Sie ein Testlauf, um etwaige auftretende Probleme frühzeitig beheben zu können.

Der Übersicht halber finden Sie im Moodle ausführliche Erklärungen zur Linux-Shell in einer separaten „Erklärungen“-Datei, die in gleichnamige Abschnitte wie die Aufgaben gegliedert ist. In dieser Datei finden Sie daher ausschließlich die Aufgabenstellungen, ohne weitere Erläuterungen zu den einzelnen Shell-Befehlen.

Lediglich Aufgabe 2 ist verpflichtend abzugeben, wir empfehlen jedoch die Bearbeitung aller Aufgaben. Sie finden zusätzlich im Moodle ein kleines „Cheat Sheet“ mit den wichtigsten Linux-Befehlen.

Aufgabe 1 Erste Schritte (nicht abzugeben)

Zuallererst müssen Sie sich per SSH mit dem von uns gestellten Linux-Server verbinden:

```
ssh -p 10000 <user>@teaching.its.uni-luebeck.de
```

Die Zugangsdaten sind auf dem Praktikumsserver zu finden.

Geben Sie nun die folgenden Befehle jeweils ein und erklärt Sie sich im Anschluss gegenseitig, was Sie sehen und was sich geändert hat.

1. Geben Sie den Befehl `ls` ein. Was für Dateien finden Sie vor?
2. Geben Sie nun den Befehl `ls -l` ein. Was hat sich verändert?
3. Geben Sie nun den Befehl `ls -la` ein. Was hat sich verändert?
4. Probieren Sie die Optionen `-g -G -h -o` in Zusammenhang mit der Option `-l` aus. Was ändert sich?
5. Vergleichen Sie die Ausgaben von `'ls .'` und `'ls'`.

Aufgabe 2 Neues Erschaffen

1. Erstellen Sie in dem Verzeichnis 'Notizen' einen neuen Ordner namens '2023'.
2. Legen Sie eine neue Datei 'Gruppe.txt' in dem gerade erstellten Ordner an.
3. Schreiben Sie in die 'Gruppe.txt'-Datei Ihren SSH-Benutzernamen (z.B. 'user1').

Aufgabe 3 Ausgabe (freiwillig)

1. Benutzen Sie den Befehl `echo` und den Redirect Operator `>`, um in die Datei `'Notizen/2023/Gruppe.txt'` Ihre studentische E-Mail-Adresse einzutragen. ◀ 20 / 6
2. Sortieren Sie Alice' ToDo-Liste mit den Befehlen `sort`, `cat` und dem Pipe Operator `|`. Speichern Sie die Ausgabe in einer neuen Datei `'SortedToDo.txt'`.
3. Mit dem Befehl `cal` können Sie sich das Kalenderblatt des aktuellen Monats anzeigen lassen. Alice möchte in ihrer sortierten ToDo-Liste am Ende genau so ein Kalenderblatt eingefügt haben. Helfen Sie ihr, indem Sie den Append-Operator `>>` und den `cal`-Befehl benutzen. ◀ 30 / 8
4. Schauen Sie sich die Ausgabe von dem Befehl `ls -R` mit dem Pager `less` an. Was macht die `-R` Option?

Aufgabe 4 Bewegung (freiwillig)

1. Verschieben Sie die Datei `'ToDo.txt'` in den Ordner `'Notizen'`. ◀ 20 / 6
2. Erstellen Sie im Homeverzeichnis einen neuen Ordner `'Logs'` und kopieren Sie die Logdatei `'/var/log/access.log'` dorthin. ◀ 20 / 6
3. Löschen Sie die Datei `'error'` aus dem Ordner `'Fehlgeschlagen'`. ◀ 20 / 6
4. Löschen Sie den Ordner `'Fehlgeschlagen'`. ◀ 20 / 6

Aufgabe 5 Suchen (freiwillig)

1. Benutzen Sie den `grep` Befehl und den Redirect Operator `>`, um die Datei `'Logs/access.log'` nach dem String `'supersecret'` zu durchsuchen und die gefundene(n) Zeile(n) in eine neue Datei namens `'Logs/secret.log'` zu schreiben. ◀ 30 / 8

Aufgabe 6 Shell Scripting (freiwillig)

1. Alice möchte mögliche Hacker etwas ausbremsen, indem sie in einem Ordner `privat` zahlreiche unbrauchbare Dateien und Ordner anlegt. Das möchte sie jedoch natürlich nicht manuell machen, daher bittet sie Sie darum, ein geeignetes Skript `randomfiles.sh` zu erstellen. Das Skript soll:
 - Als Parameter die Anzahl der zu erstellenden Ordner $d \in \{1, \dots, 15\}$ und der pro Ordner zu erstellenden Dateien $f \in \{1, \dots, 15\}$ erhalten.
 - Einen Ordner `privat` im aktuellen Verzeichnis anlegen.
 - Im Ordner `privat` genau d verschiedene Unterordner anlegen, deren Namen aus einer von Bob erstellten Liste `dirname.txt` entnommen werden sollen.
 - In jedem dieser Unterordner genau f verschiedene Dateien mit 100 Bytes zufälligem Inhalt anlegen, wobei die Dateinamen aus einer von Bob erstellten Liste `filenames.txt` entnommen werden sollen. Dateinamen dürfen sich zwischen Ordnern wiederholen.

Die Namenslisten finden Sie im Moodle, und während des Praktikums in dem Home-Verzeichnis auf Ihrem Server.

Das Skript kann beispielsweise mit folgendem Aufruf getestet werden, der drei Ordner mit jeweils zwei Dateien erstellen lässt:

```
./randomfiles.sh 3 2
```

Hinweis: Der *Shell-Skripting*-Abschnitt des Erklärungstextes könnte für diese Aufgabe hilfreich sein. Zudem hat Alice Ihnen bereits etwas Arbeit abgenommen und einige Bausteine zusammengesucht, die vielleicht hilfreich sein könnten:

```
# Read lines of file into an array
readarray -t arrayname < file.txt

# Accessing an array
echo "${arrayname[$j]}"

# Generate file with 100 random bytes
dd if=/dev/urandom of="randomfile.txt" bs=1 count=100
```

2. Alice möchte das erstellte Skript nun mit ihren Freunden teilen und bittet Sie daher darum eine kleine Hilfsoption einzubauen. Bei Aufruf des Skripts mit der Option `-h` sollte dabei ein Hilfstext ausgegeben werden, welcher die allgemeine Anwendung und die Parameter beschreibt, mit folgender Form:

```
usage: randomfiles.sh d f -h
  d: # Erklärung des Parameters
  f: # Erklärung des Parameters
-h: # Erklärung der Option
```

Recherchieren Sie hierzu das Kommando `getopts` und passen Sie Ihr Skript entsprechend an.

◀ 60 / 15

Hinweis: Bitte halten Sie sich genau an das Format aus obigem Beispiel.