

Übungen zur Shell und zu Shellscripts

Entwickeln Sie Lösungen zu folgenden Problemen. Treffen Sie alle notwendigen Annahmen.

Entwickeln Sie ein Script, das alle als Argument angegebenen Dateien zusammenhängt und schreiben Sie sie in die gewünschte Ausgabedatei. Machen Sie Sicherheitsabfragen, um Datenverluste zu vermeiden. Aufruf: `hang Ausgabedatei Datei(en)`

Simulieren Sie folgende C-Schleife in einem Shellscript:

```
for (i=1; i<=10;i++)  
    printf ("%i\n":i);
```

Backup: Es soll ein Shellscript geschrieben werden, das die Aufgabe übernimmt die Daten der Benutzer eines Unixsystems zu sichern. Diese Daten sollen jeweils gepackt und für jeden Benutzer in eine individuelle Datei geschrieben werden.

Ein Administrator will auf seiner sich langsam füllenden Homepartition etwas Platz schaffen und will die core dumps der Benutzer löschen. Dazu werden die Dateien mit dem Namen `core` gesucht und gelöscht. Das Löschen der Dateien wird protokolliert.

Ein Administrator braucht ein Script, das eine Datei einliest und bis auf die letzten `n` Zeilen alles auf `stdout` ausgibt. Es sollen also die letzten `n` Zeilen nicht ausgegeben werden. Aufruf: `abschneiden Datei NrZeilen`

Ein Benutzer macht mit `wget` Downloads. Er lässt den PC am Abend allein mit den Downloads laufen. Nachdem die Downloads beendet sind soll die Internetverbindung geschlossen und der PC heruntergefahren werden.

Ein Benutzer nimmt sein Notebook von zu Hause mit zur Arbeit. Es stört ihn, dass er in der Arbeit einen anderen smtp- und DNS-Server benutzen muss als zu Hause und diese Einstellungen manuell durchführen muss. Die IP-Adresse und der Gateway sind in beiden Netzen identisch. Die Konfiguration des Notebooks soll automatisch ohne Zutun des Benutzers erfolgen. Tipp: Im Firmennetz gibt es Server, die es zu Hause nicht gibt.

In einer alten Linuxlösung an der Gewerbeoberschule gab es 2 Linuxserver (`goblin` und `goblin2`) und keinen eigenen Gateway. Beide Server waren als X-Terminalserver eingerichtet und die Benutzer haben auf beiden PCs gearbeitet. Der Internetzugang ist über den Server `goblin` über ein ISDN-Modem gelaufen. Das ISDN-Modem wird durch ein Script geöffnet. Das Script muss überprüfen, ob es auf dem richtigen Server gestartet wurde, ob das Modem schon offen ist und ob das Script nicht schon läuft. Das Script kann zum Öffnen des Modems `ifup ppp0` verwenden.

Ein älteres RealTimeLinux-Notebook steuert eine Fertigungsanlage. Fällt der Strom aus, muss das Notebook eine analoge Internetverbindung öffnen, eine E-Mail an das Telefon eines Mitarbeiters schicken (Programm `mail`) und dann nach 10 Minuten ohne Strom selbst herunterfahren. Kann die Stromversorgung innerhalb der 10 Minuten aber wieder hergestellt werden und bleibt für eine Minute erhalten, wird die Anlage wieder hochgefahren (`service produktion start`). Das Vorhandensein des Stroms kann auf dem Notebook mit `apm` abgefragt werden. Ein Ergebnis mit Stromversorgung kann so aussehen: „AC on-line, battery charging: 100% (5:45)“, ohne Strom hingegen so: „AC off-line, battery status high: 100% (5:45)“.