



SV3: Linux Server

Die Shell Teil 3

Agenda

Die Bash

- Kanäle der Bash
- Ein- und Ausgabeumleitung
- Pipes

Umleitung der Standardein- und -ausgabe

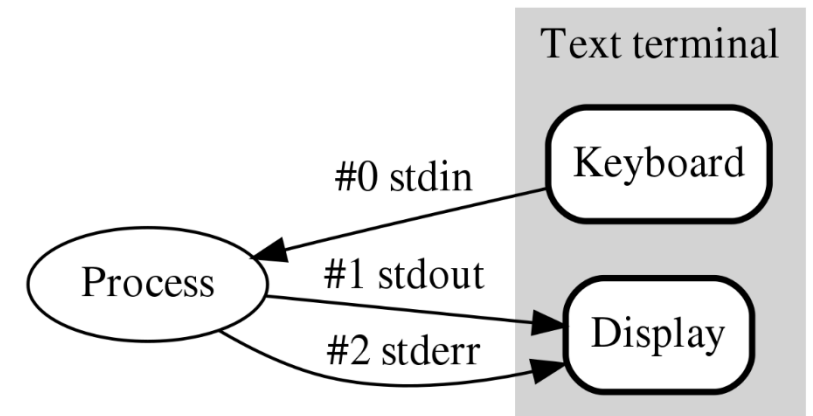
Ein typisches Programm liest Daten (z.B. die Gehälter der Mitarbeiter einer Firma), bearbeitet sie (z.B. addiert die einzelnen Gehälter auf) und gibt das Endergebnis (hier die Summe aller Gehälter) aus.

- Bei den meisten Linux/Unix-Programmen liest das Programm von »der Standardeingabe« und schreibt das Ergebnis auf »die Standardausgabe«.

Die Shell Teil 3: Die Bash

Die Standardein- und Standardausgabe sind das Terminal des jeweiligen Benutzers.

- Fehlermeldungen, die die Ausführung des Kommandos betreffen, werden ebenfalls am Bildschirm angezeigt. Zu diesen Fehlermeldungen gehören z.B. Hinweise, dass eine Datei nicht vorhanden ist.
- Wenn Sie versehentlich »sl« statt ls aufrufen, werden Sie die Fehlermeldung erhalten »sl not found« (sl nicht gefunden). Die Ein- und Ausgabe ist nummeriert (0, 1, 2).



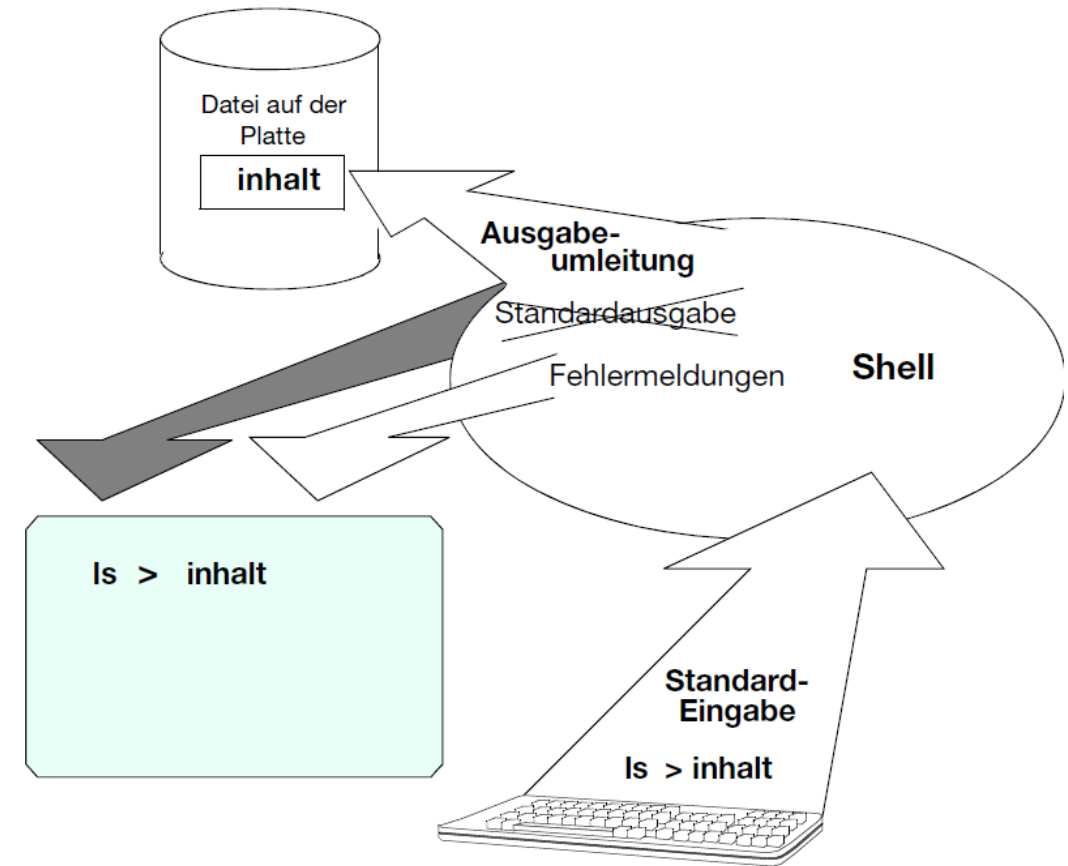
Wie und wann leiten Sie die Standardausgabe um?

- Unter der Shell wird diese Umleitung in eine Datei dadurch erreicht, dass nach dem Kommando das Umleitungszeichen »>« eingegeben wird.

Kommando > Datei oder Gerät

Die Shell Teil 3: Die Bash

- Die Ausgabe (das Inhaltsverzeichnis des aktuellen Directories) nicht auf den Bildschirm, sondern in eine Datei auf der Platte geschrieben wird.
- Die Datei wird, da kein zusätzlicher Pfadname angegeben ist, in dem aktuellen Directory angelegt.



Die Shell Teil 3: Die Bash

Erinnern Sie sich noch an das Kommando cat?

Die Grundfunktion des Kommandos ist es, einen Dateiinhalt anzuzeigen.

```
sb@ub:~/test$  
sb@ub:~/test$ #Umleitung der Ausgabe von ls:  
sb@ub:~/test$ ls -la > inhalt  
sb@ub:~/test$  
sb@ub:~/test$ #Anzeige des Dateiinhalts von "Inhalt":  
sb@ub:~/test$ cat inhalt  
insgesamt 16  
drwxr-xr-x  2 sb sb 4096 Apr 16 13:31 .  
drwxr-xr-x 17 sb sb 4096 Apr 16 14:55 ..  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 befehle  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 D*  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 Datum  
-rw-r--r--  1 sb sb   0 Apr 16 14:56 inhalt  
-rw-r--r--  1 sb sb  50 Apr 16 13:28 inhalt2  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 neu  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 projektA  
-rw-r--r--  1 sb sb  26 Apr 16 13:31 ueb1  
sb@ub:~/test$
```

Die Shell Teil 3: Die Bash

Wie können Sie Dateien mit Hilfe der Ausgabeumleitung erstellen?

Sie rufen das Kommando cat ohne Dateinamen auf, dann wird als Standardeingabe das Terminal angenommen. Gleichzeitig geben Sie an, dass die Ausgabe auf eine Datei umgeleitet werden soll:

```
cat > datei
```

Strg + D beendet die Ausgabe

Die Shell Teil 3: Die Bash

Genauso kann jedes andere Kommando umgeleitet werden, so z.B. die Ausgabe von `date` in die Datei `Datum`: **`date > Datum`**

Die Umleitung ist sehr einfach, sie kann allerdings unangenehme Auswirkungen haben, wenn Sie einen Dateinamen angeben, der bereits vorhanden ist.

Wenn Sie das Kommando **`date > neu`** aufrufen, würde der in der vorherigen Datei eingegebene Text, der bereits in der Datei `neu` steht, überschrieben werden, ohne dass das System Sie warnt.

Wie können Sie eine Datei ergänzen?

Um das Ergebnis eines Kommandos an eine bestehende Datei anzuhängen, werden zwei Größer Zeichen als Umleitung verwendet:

Kommando >> bestehende Datei

>> hängt die Ausgabe an eine Datei an!

Die Shell Teil 3: Die Bash

Wie können Sie eine Datei ergänzen?

- An die mit `ls > inhalt` erstellte Datei kann z.B. mit dem Kommando `date` das Datum angehängt werden

```
sb@ub:~/test$  
sb@ub:~/test$ date >> inhalt  
sb@ub:~/test$  
sb@ub:~/test$ cat inhalt  
insgesamt 16  
drwxr-xr-x  2 sb sb 4096 Apr 16 13:31 .  
drwxr-xr-x 17 sb sb 4096 Apr 16 14:55 ..  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 befehle  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 D*  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 Datum  
-rw-r--r--  1 sb sb   0 Apr 16 14:56 inhalt  
-rw-r--r--  1 sb sb  50 Apr 16 13:28 inhalt2  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 neu  
-rw-r--r--  1 sb sb   0 Apr 16 13:22 projektA  
-rw-r--r--  1 sb sb  26 Apr 16 13:31 ueb1  
Do 16. Apr 15:06:37 CEST 2020  
sb@ub:~/test$
```

Die Shell Teil 3: Die Bash

Und nun wird Ihnen auch die Bedeutung von cat (concatenate – zusammenhängen) klar werden:

```
cat Dateiname1 Dateiname2 Dateiname3 >> Dateiname4
```

cat – Das Kommando, um Dateien aneinanderzuhängen

Die Dateien *Dateiname1*, *Dateiname2*, *Dateiname3* werden an das Ende der Datei *Dateiname4* angehängt.

Die Shell Teil 3: Die Bash



Linux/Unix ist sehr umweltfreundlich 😊

Wie werfen Sie die Ausgabe weg?

Wird das Umleitungszeichen (>) für die Ausgabe zusammen mit /dev/null verwendet, so wird das Kommando zwar ausgeführt, das Ergebnis jedoch nicht angezeigt, sondern verworfen.

Kommando > /dev/null

> /dev/null um die Ausgabe eines Kommandos zu verwerfen

Die Shell Teil 3: Die Bash

Wie können Sie die Fehlerausgabe umleiten?

- Bei der Umleitung von Ein- und Ausgabe wird die Unterscheidung durch das Größer-»>« und Kleinerzeichen »<« ausgedrückt.
- Soll die Ausgabe der Fehlermeldung umgeleitet werden, wird vor das Größer Zeichen die Zahl 2 gesetzt:

Kommando 2> Dateiname oder Gerät

2> um Fehlermeldungen umzuleiten

Die Shell Teil 3: Die Bash

Wie können Sie bei Ausgaben umleiten?

- Sollen beide Ausgaben umgeleitet werden, dann wird vorzugsweise die erste Variante gewählt. Die 2te funktioniert nicht bei allen Linux Distributionen:

Kommando > /dev/null 2>&1

Kommando 2>&1> /dev/null

Die Shell Teil 3: Die Bash

Wie können Sie die Fehlermeldungen umleiten?

```
sb@ub:~/test$ cat inhalt dtum 2> fehler # Eine Datei dtum gibt es nicht
insgesamt 16
drwxr-xr-x  2 sb sb 4096 Apr 16 13:31 .
drwxr-xr-x 17 sb sb 4096 Apr 16 14:55 ..
-rw-r--r--  1 sb sb   0 Apr 16 13:22 befehle
-rw-r--r--  1 sb sb   0 Apr 16 13:22 D*
-rw-r--r--  1 sb sb   0 Apr 16 13:22 Datum
-rw-r--r--  1 sb sb   0 Apr 16 14:56 inhalt
-rw-r--r--  1 sb sb  50 Apr 16 13:28 inhalt2
-rw-r--r--  1 sb sb   0 Apr 16 13:22 neu
-rw-r--r--  1 sb sb   0 Apr 16 13:22 projektA
-rw-r--r--  1 sb sb  26 Apr 16 13:31 ueb1
Do 16. Apr 15:06:37 CEST 2020
sb@ub:~/test$
sb@ub:~/test$ # Die Fehlermeldung wird in die Datei "fehler" umgeleitet
sb@ub:~/test$
sb@ub:~/test$ cat fehler
cat: dtum: Datei oder Verzeichnis nicht gefunden
sb@ub:~/test$
```


Wie leiten Sie die Standardeingabe um?

Geben Sie bei einem Kommando das Kleinerzeichen und dann den Namen einer Datei an, erwartet Linux/Unix keine Eingabe am Terminal, sondern holt sich die Information aus der nach dem Kleinerzeichen angegebenen Datei.

Kommando < Datei

< um die Standardeingabe umzuleiten

Die Shell Teil 3: Die Bash

Wie können Sie die Umleitung der Standardeingabe nutzen?

Wenn Sie z.B. einen Brief an den Benutzer dirk versenden wollen, und Sie haben den Text dazu bereits in einer Datei mit dem Namen dirk.txt abgelegt, dann können Sie folgendes Kommando verwenden:

```
sb@ub:~$ mail dirk < dirk.txt
```

Der Brief wird in diesem Fall nicht am Terminal eingegeben, sondern es wird stattdessen auf die Datei dirk.txt zurückgegriffen.

Die Shell Teil 3: Die Bash

Verkettung von Kommandos

Um mehrere Kommandos hintereinander aufzurufen, können Sie diese in einer Zeile durch ein Semikolon »;« verketten.

Kommando1; Kommando2; Kommando3

; um Kommandos zu verketten

```
sb@ub:~$ date; who; ls
Do 16. Apr 15:21:15 CEST 2020
sb      :1      2020-04-16 13:16 (:1)
Bilder  Dokumente examples.desktop Musik      Personal  test      Vorlagen
dirk.txt Downloads  inhalt      Öffentlich Schreibtisch Videos
sb@ub:~$
```

Verkettung von Kommandos / Bedingte Kommandoausführung

Weitere Möglichkeiten Kommandos zu verketteten:

- **&&** - Der Befehl wird nur ausgeführt, wenn der vorherige Befehl fehlerfrei ausgeführt wurde. (Exitcode == 0)
- **||** - Der Befehl wird ausgeführt, wenn der vorherige Befehl nicht fehlerfrei ausgeführt wurde. (Exitcode <> 0)

Die Shell Teil 3: Die Bash

Der Pipe-Mechanismus unter der Shell

Was ist eine Pipe? Auch wenn die Köpfe manchmal unter Linux/Unix rauchen, die Pipe hat nichts mit einer Tabakpfeife zu tun. Die Pipe ist hier eher als pipeline (Rohrleitungssystem) zu verstehen.

Kommando1 | Kommando2

| um Kommandos über ein Pipe miteinander zu verbinden

Die Ausgabe von Kommando1 ist die Eingabe von Kommando2

Die Shell Teil 3: Die Bash

Soll das Inhaltsverzeichnis von dem Directory **/bin** seitenweise am Bildschirm angezeigt werden, brauchen Sie ein Programm.

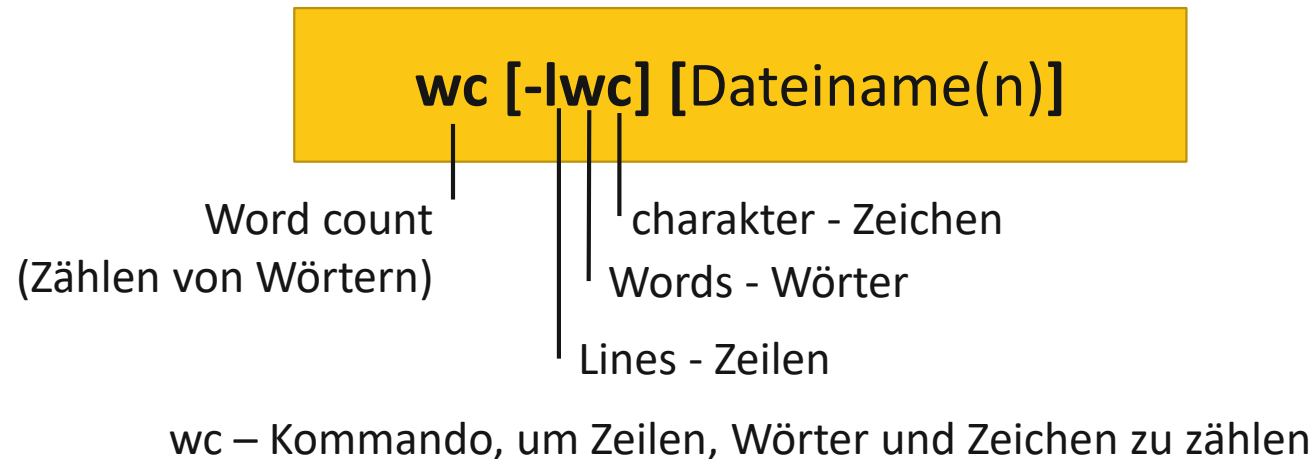
- Die Ausgabe von **ls /bin/** wird an den Befehl **more** weitergeleitet und daher seitenweise ausgegeben.

```
sb@ub:~$ ls /bin | more
```

Die Shell Teil 3: Die Bash

Um die Pipe an weiteren Beispielen zu zeigen, verwenden Sie das Kommando wc.

Mit diesem Kommando können Sie zählen, wie viele Zeilen, Wörter und Zeichen in einer Datei enthalten sind.



Die Shell Teil 3: Die Bash

Wenn Sie z.B. wissen wollen, wie viele Dateien (bzw. Kommandos) unter dem Directory /bin enthalten sind, so können Sie mit dem Pipe das Ergebnis erhalten:

```
sb@ub:~$ ls /bin | wc -w  
162  
sb@ub:~$
```


Welche Eigenschaften hat eine Pipe?

- Sie verbindet zwei Kommandos über einen temporären Puffer.
- Alles, was in diesen Puffer geschrieben wird, wird sofort an das zweite Kommando weitergeleitet.
- Das Betriebssystem und die Shell übernehmen dabei die Pufferung und die Synchronisation der Weiterleitung der Daten.
- Damit kann die Ausgabe des vorherigen Befehls immer weiter bearbeitet werden (schnellere Verarbeitung als bei der Ein- und Ausgabeumleitung).
- In einer Kommandofolge dürfen beliebig viele Pipes vorkommen.

Wie können Sie die Pipe nutzen?

Ein paar praktische Beispiele:

- Es kann sein, dass der Inhalt eines Directories mit ls nicht richtig alphabetisch sortiert angezeigt wird.
- Enthält Ihr Directory Dateinamen und/oder Unter-Directories, die mit Groß- und Kleinbuchstaben beginnen, so könnten zuerst die mit Großbuchstaben beginnenden Namen angezeigt werden, dann erst alle Namen, die mit Kleinbuchstaben anfangen.

Die Shell Teil 3: Die Bash

Verbinden Sie das ls-Kommando durch eine Pipe mit einem Sortierprogramm.

Dann können Sie sich z.B. die Ausgabe des Inhaltsverzeichnisses so sortieren lassen, dass Groß- und Kleinbuchstaben gleichwertig alphabetisch sortiert werden

```
sort [-frn] [Dateiname(n)]
```

f = fold - Klein- und Großbuchstaben sind gleichberechtigt

r = reverse – Es wird in umgekehrter Reihenfolge sortiert

n = Am Zeilenanfang beginnende Zahlen, die durch Leerzeichen von anderen Zeichen getrennt sind, werden in arithmetischer Reihenfolge sortiert.

Die Shell Teil 3: Die Bash

Verbinden Sie das ls-Kommando durch eine Pipe mit einem Sortierprogramm.

Dann können Sie sich z.B. die Ausgabe des Inhaltsverzeichnisses so sortieren lassen, dass Groß- und Kleinbuchstaben gleichwertig alphabetisch sortiert werden

```
sb@ub:~/test$ ls | sort -f
befehle
D*
Datum
fehler
inhalt
inhalt2
neu
projektA
ueb1
sb@ub:~/test$
```

Die Shell Teil 3: Die Bash

Mit dem Kommando **grep** (als Eselsbrücke: greifen, heraussuchen) können Sie bestimmte Worte oder Zeichenfolgen aus Dateien oder Ergebnissen von Kommandos herausfiltern.

```
grep „Muster“ [Dateiname(n)]
```

grep – Kommando, um nach Mustern in Dateiinhalten zu suchen

Die Shell Teil 3: Die Bash

Die Ausgabe von journalctl wird nach den Schlüsselwörtern „Intel und Hyper-V“ durchsucht.

```
sb@ub:~$ journalctl | grep Intel
Apr 15 09:45:49 ub kernel: Intel GenuineIntel
Apr 15 09:45:49 ub kernel: smpboot: CPU0: Intel(R) Core(TM) i5-8500 CPU @ 3.00GHz (family: 0x6, model: 0x9e, stepping: 0xa)
Apr 16 13:16:07 ub kernel: Intel GenuineIntel
Apr 16 13:16:07 ub kernel: smpboot: CPU0: Intel(R) Core(TM) i5-8500 CPU @ 3.00GHz (family: 0x6, model: 0x9e, stepping: 0xa)
sb@ub:~$ journalctl | grep Hyper-V
Apr 15 09:45:49 ub kernel: DMI: Microsoft Corporation Virtual Machine/Virtual Machine, BIOS Hyper-V UEFI Release v4.0 01/30/2019
Apr 15 09:45:49 ub kernel: Hypervisor detected: Microsoft Hyper-V
Apr 15 09:45:49 ub kernel: Hyper-V: features 0x2e7f, hints 0x60c2c
Apr 15 09:45:49 ub kernel: Hyper-V Host Build:18362-10.0-0-0.720
Apr 15 09:45:49 ub kernel: Hyper-V: LAPIC Timer Frequency: 0xc3500
Apr 15 09:45:49 ub kernel: tsc: Marking TSC unstable due to running on Hyper-V
Apr 15 09:45:49 ub kernel: Hyper-V: Using hypercall for remote TLB flush
Apr 15 09:45:49 ub kernel: Hyper-V: PV spinlocks enabled
Apr 15 09:45:49 ub kernel: Hyper-V: Using IPI hypercalls
Apr 15 09:45:49 ub kernel: Hyper-V: Using enlightened APIC (xapic mode)
Apr 16 13:16:07 ub kernel: DMI: Microsoft Corporation Virtual Machine/Virtual Machine, BIOS Hyper-V UEFI Release v4.0 01/30/2019
```



**VIELEN DANK
FÜR IHRE
AUFMERKSAMKEIT!**