

1. Arbeiten an der Konsole.

Sie gelangen in die Konsole:

- Login bei einer Anmeldung am Textbildschirm (virtuelle Konsole).

```
pegasus.gate.local login: kubi
Password: (Das Passwort bleibt beim Tippen unsichtbar, auch keine Symbole für die Zeichen)
Last login: Sun Oct  9 20:38:49 on tty2
[kubi@pegasus ~]$
```

Das Ausloggen an einem Textterminal erfolgt mit dem Kommando `logout` oder `exit`

```
[kubi@pegasus ~]$ logout
```

oder nach Anmeldung bei grafischem Login durch Starten der Anwendung für das Terminal- bzw. Konsolenfenster:

- KDE Programmnenü KDE - SYSTEM.
- Gnome: AKTIVITÄTEN - SYSTEMWERKZEUGE.

● **Fedora 18-20**, vermutlich bedingt durch diverse Neuerungen im Installationsprogramm Anaconda und/oder dem Systemstart der Konsolenkonfiguration über `systemd`, können nicht **deutsche Umlaute und Apostroph in den virtuellen Textkonsolen** korrekt angezeigt bzw. eingegeben werden. Dies kann behoben werden, wenn in der Datei `/etc/rc.d/rc.local` folgende Einträge aufgenommen werden:

Bei **Fedora 18, 19**:

```
#!/bin/sh
# Umstellung auf deutsche Zeichen in virt. Texconsolen
setfont -C /dev/tty1 latarcyrheb-sun16
loadkeys --console=/dev/tty1 /lib/kbd/keymaps/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty2 latarcyrheb-sun16
loadkeys --console=/dev/tty2 /lib/kbd/keymaps/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty3 latarcyrheb-sun16
loadkeys --console=/dev/tty3 /lib/kbd/keymaps/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty4 latarcyrheb-sun16
loadkeys --console=/dev/tty4 /lib/kbd/keymaps/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty5 latarcyrheb-sun16
loadkeys --console=/dev/tty5 /lib/kbd/keymaps/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty6 latarcyrheb-sun16
loadkeys --console=/dev/tty6 /lib/kbd/keymaps/i386/qwertz/de-latin1-nodeadkeys.map.gz
```

Bei **Fedora 20**:

```
#!/bin/sh
# Umstellung auf deutsche Zeichen in virt. Texconsolen
setfont -C /dev/tty1 latarcyrheb-sun16
loadkeys --console=/dev/tty1 /lib/kbd/keymaps/legacy/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty2 latarcyrheb-sun16
loadkeys --console=/dev/tty2 /lib/kbd/keymaps/legacy/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty3 latarcyrheb-sun16
loadkeys --console=/dev/tty3 /lib/kbd/keymaps/legacy/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty4 latarcyrheb-sun16
loadkeys --console=/dev/tty4 /lib/kbd/keymaps/legacy/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty5 latarcyrheb-sun16
loadkeys --console=/dev/tty5 /lib/kbd/keymaps/legacy/i386/qwertz/de-latin1-nodeadkeys.map.gz
setfont -C /dev/tty6 latarcyrheb-sun16
loadkeys --console=/dev/tty6 /lib/kbd/keymaps/legacy/i386/qwertz/de-latin1-nodeadkeys.map.gz
```

● **Wichtig:** Zu der Datei `/etc/rc.d/rc.local`, die manuell angelegt werden muss, lesen Sie bitte das Unterkapitel „Prozesse beim Systemstart ausführen.“ im Hauptkapitel „Prozessverwaltung“.

1.1 Kommandoeingabe.

Die Kommandoeingabe erfolgt schematisch nach dem Prinzip:

Kommando [Optionen] [Parameter]

- **Optionen**, die Reihenfolge der Optionen ist bis auf wenige Ausnahmen beliebig, beginnen immer
 - mit einem vorangestellten - bei einzelnen Optionsbuchstaben. *Beispiel:*

```
$ who -b
Systemstart 2011-12-12 11:05
```

Es dürfen mehrere Optionsbuchstaben hinter dem - Zeichen aufgeführt werden (ohne Trennungszeichen). Beispiel:

```
$ who -m -H
NAME LEITUNG ZEIT KOMMENTAR
kubi pts/0 2011-12-12 12:19 (:0.0)
$ who -mH
NAME LEITUNG ZEIT KOMMENTAR
kubi pts/0 2011-12-12 12:19 (:0.0)
```

- mit vorangestellten -- vor einem Optionswort. *Beispiel:*

```
$ who --boot
Systemstart 2011-12-12 11:05
```

- **Ende der Optionen kennzeichnet man mit --**, dies kann in speziellen Fällen nötig sein, wenn man zum Beispiel eine Datei vorliegen hat, die mit einem oder zwei Minuszeichen beginnt. Ohne -- Kennzeichnung der Optionen würde in allen Fällen die Hilfe zu den Kommandos ausgegeben werden:

```
$ touch -- --help
$ ls -l -- --help
-rw-r--r--. 1 kubi kubi 0 30. Aug 15:22 --help
$ rm -i -- --help
rm: reguläre leere Datei '--help' entfernen? j
$
```

- **Parameter** beginnen nie mit - oder --

```
$ who am i
kubi pts/0 2011-12-12 12:19 (:0.0)
```

1.2 Hilfen anzeigen.

Eine **Kurzbeschreibung** erhält man durch **Eingabe eines Kommandos mit der Option --help**.

```
$ who --help
Aufruf: who [OPTION]... [ DATEI | ARG1 ARG2 ]
Informationen über gerade angemeldete Benutzer ausgeben.

-a, --all      dasselbe wie -b -d --login -p -r -t -T -u
-b, --boot     Zeit des letzten Rechnerstarts („system boot“)
-d, --dead     tote Prozesse ausgeben
-H, --heading  Kopfzeile mit Spaltenbezeichnungen ausgeben

... usw. ...
```

Die **Bedienungsanleitung**, die sogenannten Manual Pages oder Manpages, erhält man durch **Aufruf des Kommandos man gefolgt vom Namen des Programms**.

```
$ man who
WHO(1)                                     FSF                                     WHO(1)

NAME
      who - Zeigen, wer eingeloggt ist

ÜBERSICHT

      ... usw. ...
```

COPYRIGHT

Copyright © 2000 Free Software Foundation, Inc.
 Dies ist freie Software; die Kopierbedingungen stehen in den Quellen.
 Es gibt keine Garantie; auch nicht für VERKAUFBARKEIT oder FÜR SPEZIELLE ZWECKE.

GNU sh-utils 2.0.11

Mai 2001

WHO(1)

Die wichtigsten **Tasten zum Navigieren** für man (es sind die gleichen wie beim Kommando less)

Bild-Oben/Unten	Seitenweise vor und zurück blättern.
d	Halbe Seite nach unten (engl. down)
u	Halbe Seite nach oben (engl. up)
g	An den Anfang der Manpage springen.
G	An das Ende der Manpage springen.
h	Hilfe (engl. help) zur Navigation aufrufen.
q	Ansicht der Manpage beenden (engl. quit).
/<Muster>	Abwärts nach Vorkommen des Musters in Manpage suchen.
?<Muster>	Aufwärts nach Vorkommen des Musters in Manpage suchen.
n	Nach unten weiter suchen (eng. next).
N	Nach oben weiter suchen.

Die **wichtigsten Optionen und Parameter** des Kommandos man:

man [<KapitelNr> -a] [-L <Locale>] [-it] [-k] <Suchwort>	
KapitelNr	Die Kapitelnummer der Online Hilfe:
1	Ausführbare Programme oder Shellbefehle
2	Systemaufrufe (Kernelfunktionen)
3	Bibliotheksauftrufe (Funktionen in System-Bibliotheken)
4	Spezielle Dateien (gewöhnlich in /dev)
5	Dateiformate und Konventionen, z. B. /etc/passwd
6	Spiele
7	Makropakete und Konventionen, z. B. man(7), groff(7)
8	Systemadministrationsbefehle (in der Regel nur für root)
-a	Alle Kapitel (Standard Kapitel für erstes Auftreten des Suchwortes).
-L Locale	Angabe der Sprache als Locale (Lokale für Deutsch = de, für Englisch = en).
-i	Ignoriere Groß- / Kleinschrift des Suchwortes.
-t	Formatiert Ausgabe nach Postscript.
-k	Durchsucht Manual <u>und</u> Kurzbeschreibungen nach Suchwort. Identisch mit dem Kommando apropos
Suchwort	Suchwort kann ein Kommando, ein Dateiname sein.

```
$ man -k who
who (1)           - anzeigen, wer angemeldet ist
whoami (1)        - Die effektive Nutzer-ID ausgeben
at.allow (5)       - determine who can submit jobs via at or batch
at.deny (5)        - determine who can submit jobs via at or batch
ftpwho (1)         - show current process information for each FTP session
ipsec_newhostkey (8) - generate a new raw RSA authentication key for a host
ipsec_showhostkey (8) - show host's authentication key
jwhois (1)         - client for the whois service
pom_xpath_disable (7) - Disable all modules whose POM files satisfy given XPath condition
rwho_selinux (8)    - Security Enhanced Linux Policy for the rwho processes
w (1)              - Show who is logged on and what they are doing.
who (1p)           - display who is on the system
whois (1)           - client for the whois service
whois.jwhois (1)    - client for the whois service

$ apropos who
who (1)           - anzeigen, wer angemeldet ist
whoami (1)        - Die effektive Nutzer-ID ausgeben
at.allow (5)       - determine who can submit jobs via at or batch
at.deny (5)        - determine who can submit jobs via at or batch
ftpwho (1)         - show current process information for each FTP session
ipsec_newhostkey (8) - generate a new raw RSA authentication key for a host
ipsec_showhostkey (8) - show host's authentication key
jwhois (1)         - client for the whois service
pom_xpath_disable (7) - Disable all modules whose POM files satisfy given XPath condition
```

```
rwho_selinux (8)      - Security Enhanced Linux Policy for the rwho processes
w (1)                 - Show who is logged on and what they are doing.
who (1p)               - display who is on the system
whois (1)              - client for the whois service
whois.jwhois (1)       - client for the whois service
```

Zur Erzeugung der **Manpages als PDF Dateien** oder zum **Ausdruck auf einem Drucker** verwende man die folgenden Kommandos, zum Beispiel für die Manpage von who:

- Manpage von who als PDF Datei manpage_who.pdf erzeugen:

```
$ man -t who | ps2pdf14 - manpage_who.pdf
```

- Manpage drucken (es wird vorausgesetzt dass ein Standarddrucker eingerichtet wurde):

```
$ man -t who | lpr
```

Das **Kommando whatis** durchsucht im Gegensatz zu den Kommando apropos bzw. man -k nur die Kurzbeschreibungen, jedoch nicht die Manpages selbst.

```
$ whatis who
who (1)                 - anzeigen, wer angemeldet ist
who (1p)                - display who is on the system
```

1.3 Die Info Dokumente.

Für etliche Kommandos stehen neben den Manpages Info Dokumente zur Verfügung. Diese lassen sich mit dem **Kommando info** einsehen.

```
$ info who
File: coreutils.info,  Node: who invocation,  Prev: users invocation,  Up: User information
20.6 `who': Print who is currently logged in
=====
`who' prints information about users who are currently logged on.
Synopsis:
  `who' [OPTION] [FILE] [am i]
  ... usw. ...
```

Mit der Taste q beendet man das Programm. Weitere Details der Bedienung können Sie durch Aufruf der Infoseiten von info nachlesen.

1.4 Weitere Hilfen.

Weiter Dokumente zu Programmen finden sich im **Verzeichnis /usr/share/doc/** in den entsprechenden Unterverzeichnissen. Im **Internet** findet man die Webadresse zu einem Programm Paket in der Paketinformation (s. dazu das Kapitel über die Paketverwaltung, weil ein Programmname nicht unbedingt gleich dem Paketnamen sein muß).

```
$ rpm -qi tree
Name       : tree
Version    : 1.5.3
Release    : 3.fc15
Architecture: i686
Install Date: Sa 01 Okt 2011 12:31:36 CEST
Group      : Applications/File
Size       : 64935
License    : GPLv2+
Signature  : RSA/SHA256, Fr 11 Feb 2011 17:09:27 CET, Key ID b4ebf579069c8460
Source RPM : tree-1.5.3-3.fc15.src.rpm
Build Date : Mi 09 Feb 2011 21:42:32 CET
Build Host : x86-01.phx2.fedoraproject.org
Relocations: (not relocatable)
Packager   : Fedora Project
Vendor    : Fedora Project
URL       : http://mama.indstate.edu/users/ice/tree/
```

```
Summary      : File system tree viewer
Description  :
The tree utility recursively displays the contents of directories in a
tree-like format. Tree is basically a UNIX port of the DOS tree utility.
```

1.5 Terminalbedienung.

In einer Textkonsole (Konsole, Terminal) können die folgenden **Tastenkürzel und Sondertasten** verwendet werden:

Taste	Aktion
Alt+F2...F6	Zwischen virtuellen Terminals umschalten (bei grafischer Oberfläche zusätzlich STRG).
Alt+F1	Virtueller Grafikdesktop (bei grafischer Oberfläche zusätzlich STRG).
	● Bei anderen Distributionen liegt der virtuelle Grafikdesktop auf ALT+F7.
Alt+<	Zum Anfang der Eingabezeilen Historie (älteste Eingabezeile).
Alt+>	Zum Ende der Eingabezeilen Historie (jüngste Eingabezeile).
ALTGR+E	Euro Zeichen €.
Backspace, Entf	Zeichen löschen.
BildUnten	Wechsel zur neuesten Kommandozeile.
Ende	Cursor an das Ende der Eingabezeile.
Enter bzw. Return	Kommando ausführen.
Pfeil oben/unten	Eingabezeilen Historie.
Pos1	Cursor an den Anfang der Eingabezeile.
Shift+BildOben	Bildschirm scrollen.
Shift+BildUnten	Bildschirm scrollen.
Strg+C	Programmausführung eines Konsolenprogramms abbrechen.
Strg+D	EOF Dateiende Marke (engl. End Of File).
Strg+L	Bildschirm löschen (ohne aktuelle Kommandozeile). *)
Strg+Q	Ausgabe fortsetzen.
Strg+R	Rückwärtige Suche alter Kommandoeingaben. **)
Strg+S	Ausgabe stoppen.
Strg+U	Eingabezeile löschen.
Strg+W	Wort links vom Cursor löschen.
Tabulator	Datei-, Pfad- und Programmnamen ergänzen.
!Nummer	Führt die Kommandozeile mit der Nummer aus der Datei <code>~/.bash_history</code> aus. **)

*) Der Bildschirm kann sowohl mit dem **Kommando clear**, als auch mit dem **Kommando reset** gelöscht werden. Bei letzterem Befehl wird die Konsole neu initialisiert.

) Die letzten 1000 eingegebenen Kommandozeilen werden in die **Datei `~/.bash_history` geschrieben, neu hinzugekommene werden erst beim Abmelden hinzugefügt. Mit dem **Kommando history** können alle Kommandoeingaben in der Datei nummeriert ausgegeben werden (mit **Option -c** wird Inhalt der Datei `.bash_history` gelöscht, mit der **Option -a** werden alle neuen Kommandozeilen in die Datei geschrieben).

Für Linux gilt wie unter den grafischen Oberflächen auch an einer Textkonsole für die **Zwischenablage**: Durch alleiniges Markieren von Text mit der **linken Maustaste**, wird dieser sofort in die Zwischenablage kopiert (es ist also nicht STRG-C zu drücken). Die **mittlere Maustaste** fügt diesen Text wieder ein (es ist also nicht STRG-V zu drücken).

● Bei einem Textbildschirm muss das **Paket gpm** installiert sein, damit die Zwischenablage mit den Maustasten bedient werden kann.

1.6 Der Editor vi.

Der **Editor vi** bzw. **vim** ist auf jedem Unix/Linux Rechnern standardmäßig verfügbar. Der Aufruf des Editors kann mit Angabe einer Datei, die dann gleich geladen wird, oder ohne Dateiangabe erfolgen:

vi [<Datei>]

Editor zum Bearbeiten der angegebenen Datei (ohne Dateiangabe neue Datei).

Ohne Dateiangabe wird eine neue Datei geöffnet. Wird die Datei angegeben und sie existiert noch nicht, wird eine neue leere Datei mit diesem Dateinamen zur Bearbeitung angelegt. Folgende Grundoperationen des Editors sollte man auf jeden Fall beherrschen:

- Editor ohne Speicherung beenden.
- Speichern.
- Speichern und beenden.
- Cursorbewegung im Text.
- Wechsel zwischen Eingabe- und Kommandomodus.
- Löschen von Zeichen.
- Einfügen von Zeichen.
- Überschreiben von Zeichen.
- Blockoperationen über visuellen Modus.

Der vi besitzt **3 verschiedene Modi**:

- Nach dem Start des vi befindet man sich im **Kommandomodus**.
- Um Text zu erstellen, wechselt man **aus dem Kommandomodus** mit der **Taste i** oder der **Einfg-Taste in den Schreibmodus**.
- Mit der **Taste ESC** wechselt man **aus dem Schreibmodus zurück in den Kommandomodus**.
- **Aus dem Kommandomodus** wechselt man durch Eingabe der **Taste :** in den **Last-Line-Modus**.
- **Aus dem Last-Line-Modus** wechselt man mit der **Taste ESC** zurück in den **Kommandomodus**.

Last-Line-Modus.

Eingabe	Aktion	Eingabe	Aktion
:%s/<alt>/<neu>	Suche nächste Stelle mit alten Text und ersetze ihn durch den neuen Text.	:q	vi beenden, falls Datei gespeichert.
:%s/<alt>/<neu>/g	Suche im gesamten Text allen alten Text und ersetze diese durch den neuen Text.		
:w	Datei speichern (Dateiname bleibt).	:q!	vi ohne Speichern beenden.
:w <Name>	Datei unter neuem Namen speichern.	:wq	Speichern und beenden.

Kommandomodus.

Die wichtigsten Cursorbewegungen:

Taste	Bewegung	Taste	Bewegung
l	Zeichen rechts.	STRG + F	Seite vor
h	Zeichen links.	STRG + B	Seite zurück
k	Zeile hoch.	(Satz zurück.
j	Zeile runter.)	Satz vor.
0	Zeile Anfang.	{	Absatz zurück.
\$	Zeile Ende	}	Absatz vor.
b	Wort rechts.	G	Dateiende.
w	Wort links.	gg	Dateianfang.
/<Text>	Suche Textstelle abwärts.	?<Text>	Suche Textstelle aufwärts.
n	Finde nächste Textstelle.	N	Finde vorherige Textstelle.

Die wichtigsten Kommandos zum Ändern von Text:

Anzahl	Befehl	Beschreibung
	<<	Tabulatoreinzug am Zeilenanfang entfernen.
	>>	Tabulatoreinzug am Zeilenanfang.
+	a	Text kann hinter dem Cursor eingegeben werden.
+	A	Text kann am Zeilenende eingegeben werden.
	c	ändert den Text im Bereich Bereich
	C	Text ab Cursor wird bis Zeilenende gelöscht und kann neu eingegeben werden.
	d	löscht den mit Bereich angegebenen Text.
+	dd	löscht ganze Zeile.
	D	Text wird ab Cursorposition bis Zeilenende gelöscht.
+	i	Text kann vor Cursorposition eingefügt werden.
+	I	Text kann am Zeilenanfang eingefügt werden.
+	J	Verbindet aktuelle Zeile mit nachfolgender Zeile.
+	o	Fügt unter der aktuellen Zeile eine leere Zeile ein, dann Eingabemodus.
+	O	Fügt über der aktuellen Zeile eine leere Zeile ein, dann Eingabemodus.
	p	Einfügen nach Cursor.
	P	Einfügen vor Cursor.
+	r	Ersetzt Zeichen an Cursorposition durch neues Zeichen.
	R	Vorhandener Text kann überschrieben werden.
+	s	Ersetzt Zeichen durch Eingabe neuer Zeichen.
+	S	Komplette Zeile wird durch neue Eingabe ersetzt.
	u	Letztes Kommando rückgängig machen.
	U	Alle Änderungen in einer Zeile rückgängig machen.
	v	Visuelles markieren von Text, danach: <ul style="list-style-type: none"> • d löscht den markierten Text. • x schneidet den markierten Text aus. • y kopiert den markierten Text. • p, P (s.o., Text einfügen). • > rückt ein. • < rückt aus.
+	x	Zeichen an Cursorposition löschen.
+	X	Zeichen links von Cursorposition löschen.
+	yy	Zeile Kopieren.

Anzahl +: ist Optional, es kann eine Zahl zur Wiederholung des Kommandos verwendet werden (Beispiel: 3dd löscht 3 Zeilen).



Wenn Sie das **Paket vim-enhanced** installieren, können Sie ein **interaktives Tutorial** mit dem Kommando **vimtutor german** starten. Ferner können Sie statt des vi den Editor vim starten, bei dem **Syntax Highlighting** integriert ist.

1.7 Alternative Editoren.

• Editoren für Textkonsolen:

- ed
- efte, netfe (ähnlicher Editor wie früherer Turbo Pascal Editor)
- emacs
- joe (Joe's own Editor, Ähnlichkeiten mit früherer Textverarbeitung WordStar)
- moe (My own Editor, abgewandelter joe Editor)
- nano
- pico
- sed

• Editoren für grafische Oberflächen:

- gedit (Gnome Standardeditor)
- kwwrite (KDE Standardeditor)
- leafpad (ähnlich dem Standardeditor unter Windows®)
- marave („ungewöhnliches“ Design)
- mousepad (XFCE Standardeditor, ähnlich leafpad)
- pyroom
- scratchpad
- vim-X11 (grafische Variante des vi)
- xemacs (grafische Variante des emacs)

1.8 Arbeiten als Superuser root an der Konsole.



Arbeiten Sie aus Sicherheitsgründen als **normaler Benutzer**, wenn Sie keine Systemadministrationen vornehmen. Legen Sie immer eine **Sicherungskopie** von Konfigurationsdateien an, bevor Sie Änderungen an ihnen vornehmen - auch, wenn Sie mit einem (grafischen) Konfigurationstool arbeiten! Im Notfall lässt sich so der alte Zustand unproblematisch wieder herstellen.

Wichtig! Sicherungskopien von Systemdateien, zum Beispiel in /etc befindliche, **müssen immer mit der -p und -c Option beim cp Kommando erstellt werden!** Siehe dazu auch die Kapitel zur Dateiverwaltung und zu SELinux.

Arbeiten Sie als Benutzer an einer Konsole nicht als root, um nicht versehentlich Änderungen im System vorzunehmen. Sie können in einem Terminalfenster als Benutzer jederzeit mit dem **Kommando su** den Benutzer wechseln, als root benötigen Sie keine Kenntnis des Passwortes, als User benötigen Sie die Kenntnis des Passwortes. Geben Sie den Usernamen nicht an, ist es der User root.

su [- | -l] [-c <Kommando>] [<Username>]

- Neue Login-Shell.
- l Wie – (neue Login Shell).
- c Kommando Auszuführendes Kommando.
- Username Benutzer, unter dessen Kennung die Shell laufen soll (Standard: root).

Mit dem **Kommando exit** nehmen Sie die ursprüngliche Identität wieder an. Beispiel für den Wechsel der Identität und zurück.

```
$ who am i
kubi
$ pwd
/home/kubi
$ su -
Password: Root -Passwort
# pwd
/root
# who am i
root
# exit
```

```
$ who am i
kubi
$ pwd
/home/kubi
```

Beispiel für ein einzelnes Kommando, das als root ausgeführt werden soll:

```
$ ls -l /etc/gshadow
-r----- 1 root root 699 30. Jan 18:39 /etc/gshadow

$ cat /etc/gshadow | grep root
cat: /etc/gshadow: Keine Berechtigung

$ su - -c "cat /etc/gshadow | grep root"
Passwort: Root-Passwort
root:::root
root:::root,bin,daemon
root:::root,bin,daemon
root:::root,bin,adm
root:::root,adm,daemon
root:::root
... usw. ...
```

- Sie können das Kommando su auch ohne die - Option verwenden. Davon ist jedoch abzuraten, da sonst nicht das Heimatverzeichnis des Benutzers, dessen Identität man annimmt, gesetzt wird. Geben Sie immer das Minuszeichen mit an, damit Sie in das entsprechende Heimatverzeichnis wechseln.