

Alphabetische Befehlsübersicht A - L
M - Z

<i>Anzeigen</i>	
<i>Ausgeben</i>	
<i>Ausdruck</i>	
<i>Benutzerkommunikation</i>	
<i>Benutzerumgebung</i>	
<i>Dateibäume sichern</i>	
<i>Dateien und Verzeichnisse</i>	
<i>Dateityp ermitteln</i>	
<i>Datum</i>	
<i>Diskspace</i>	
<i>Drucken System V</i>	
<i>Ersetzen/Umformen</i>	
<i>Expr</i>	
<i>Hilfe</i>	
<i>Informationen über System und User</i>	
<i>Job Control</i>	
<i>Komprimieren/Dekomprimieren</i>	
<i>Kopieren</i>	
<i>Link</i>	
<i>Mixed</i>	
<i>Periodische Prozesse</i>	
<i>Sortieren</i>	
<i>Suchen in Dateien</i>	
<i>Suchen nach Dateien</i>	
<i>Terminaleinstellung</i>	
<i>Zugriffsrechte</i>	
<i>Zugriff auf Remote Rechner</i>	

<i>II</i>
<i>III</i>
<i>1</i>
<i>2</i>
<i>9</i>
<i>11</i>
<i>1</i>
<i>6</i>
<i>3</i>
<i>3</i>
<i>2</i>
<i>6</i>
<i>4</i>
<i>8</i>
<i>9</i>
<i>3</i>
<i>7</i>
<i>10</i>
<i>8</i>
<i>3</i>
<i>3</i>
<i>3</i>
<i>10</i>
<i>7</i>
<i>5</i>
<i>5</i>
<i>7</i>
<i>1</i>
<i>12</i>

Wichtige Dateien:

<i>.cshrc</i>	# Initialisierungsdatei für C-Shell
<i>.envfile</i>	# Initialisierungsfile ksh
<i>.exrc</i>	# Initialisierungsdatei für vi
<i>.login</i>	# Profile für C-Shell
<i>.profile</i>	# Profile für sh und ksh
<i>.kshrc</i>	# Startupfile für ksh wenn ENV=.kshrc
<i>/etc/default/passwd</i>	# Defaultwerte für Password-Ageing
<i>/etc/default/boot</i>	# SCO Variablen für Bootvorgang
<i>/etc/default/filesys</i>	# SCO Filesysteme die beim Boot gemountet werden /
<i>etc/default/tar</i>	# SCO Defaultdevices für tar
<i>/etc/default/man</i>	# SCO Konfigurationsfile für man
<i>/etc/gettydefs</i>	# Leitungsbeschreibung für Verbindungsaufbau
<i>/etc/group</i>	# Gruppenzuordnung der Benutzer
<i>/etc/inittab</i>	# Prozesse für boot und getty nach Änderung für getty Aktivierung durch : init q
<i>/etc/issue</i>	# Login-Bildschirm
<i>/etc/magic</i>	# Magic-Strings zur Dateierkennung
<i>/etc/motd</i>	# Tagesnachrichten (Message of Today)
<i>/etc/passwd</i>	# Benutzerkenndaten
<i>/etc/profile</i>	# Allgemeine Profile für alle Benutzer
<i>/etc/shadow</i>	# Benutzerkennworte
<i>/etc/ttytype</i>	# Terminaltype für für bestimmte Leitungen
<i>/etc/wtmp</i>	# enthält accounting Informationen # muss sporadisch von SU gelöscht werden.
<i>/usr/lib/cron</i>	# Main cron Directory
<i>/usr/lib/cron/cron.allow</i>	# nur angef. User dürfen Einträge in crontab machen
<i>/usr/lib/cron/cron.deny</i>	# angeführte User dürfen keine Einträge vornehmen
<i>/usr/lib/cron/log</i>	# Accounting Informationen
<i>/usr/lib/keyboard/german.map</i>	# Umlaute für Keyboard (INTERACTIVE)
<i>/usr/spool/cron/crontabs</i>	# Spool Bereich
<i>/usr/spool/lp/logs/lpsched</i>	# Protokoll Start/Stop Printservice
<i>/usr/spool/lp/logs/requests</i>	# Protokollierung aller Druckjobs, request und lpsched muß sporadisch vom su gelöscht werden
<i>a.out</i>	# executable File nach Umwandlung C-Programm
<i>core</i>	# Dump nach Programmabbruch (Löschen durch rm core)

A

ar Archiv (Library) erstellen und verwalten
at startet einen Prozess zur angegebenen Zeit 10

B

banner Ausgabe in Großbuchstaben 3
basename 3
batch 10
bg Kommando in Hintergrund stellen 10
break Schleife abbrechen 14

C

cal Kalender anzeigen 13
calendar einfache Terminerinnerung 13
cancel Druckauftrag abbrechen 4
cat Datei auf stdout ausgeben 2
cc Kompilieren und Linken C-Programm
cd Verzeichnis wechseln 1
chgrp neue Gruppenzuordnung für Datei 1
chmod Zugriffsrechte setzen oder ändern 1
chown neuen Eigentümer für Datei 1
chsh Loginshell ändern 1
clear Bildschirm löschen 10
cmp Vergleich von 2 Dateien 3
col Entfernen von Zeilenrückläufen aus Datei
comm Vergleich von 2 sortierten Textdateien
compress Komprimieren von Dateien 8
cp Dateien kopieren 3
cpio Dateien und Directorybäume kopieren 6
crontab Kommandos in vorgegeb. Intervallen ausführen 10
crypt Texte verschlüsseln
csplit Datei aufsplitten
csh Aufruf C-Shell 3
ct Remote Callback eines Terminals
ctags Generierung einer tags-Datei für vi
cu Rechnerkoppelung
cut Felder oder Spalten aus Datei ausgeben 8

D

date Datum und Uhrzeit ausgeben bzw setzen 13
dd kopieren von Datenträgern
df benutzte und freie Platenkapazität anzeigen 6
diff Textdateien vergleichen, Differenz anzeigen
dircmp Directories vergleichen
dirname Vollen Pfad für Dateinamen 3
disable deaktiviert bezeichneten Drucker 4
du Disk usage anzeigen 6

E

echo Argumente auf stdout ausgeben 4
ed Texteditor (zilenorientiert) 3

egrep Erweiterte Suche nach "pattern" 5
enable Aktiviert bezeichneten Drucker 7
env Environment anzeigen 3
expr verwendet die Argumente als Expression 9
exec führt Kommando aus

F

factor Primfaktorzerlegung durchführen
frep Schnelle Suche in Dateien 5
file Dateityp bestimmen 3
find Suche nach Dateien im Baum 5
finger Informationen zu eingeloggten Usern ausgeben
fmt Dateien formatieren
format Disketten formatieren
ftp Kopieren von Dateien auf entfernte Systeme

G

grep Suchen in Dateien 5
groups Gruppenzugehörigkeit eines Benutzers ausgeben

H

hd Hex-Dump ausgeben 2
head Vorderen Teil Datei anzeigen 2

I

id Eigene UID und GID ausgeben 1

J

join Mischen von zwei Dateien
jobs Stopped Jobs anzeigen 10

K

kill Signal an Prozess senden 10

L

last An- und Abmeldezeit von Benutzern ausgeben
line Eine Zeile von stdin lesen
ln Link auf Datei erzeugen 3
logname Eigenen Loginnamen ausgeben
lp Datei drucken (System V) 4
lpc
lpr Datei drucken (BSD)
lpstat Status Druckauftrag (System V) 4
ls Dateinamen anzeigen 1

M

mail Mail schicken 14
man Manualtexte anzeigen 3
mesg Eigenes Terminal sperren bzw freigeben 10
mkdir Direktory anlegen 3
mknod Gerätedatei anlegen
more Seitenweise Ausgabe von Dateien 2
mount Dateisystem mounten
mv verschieben bzw umbenennen Datei

N

newgrp	Gruppenzugehörigkeit wechseln	
news	gibt news aus	11
nice	Prozesspriorität verändern	10
nl	Zeilen in Datei nummerieren	
nohup	Prozess nach Logout weiterlaufen lassen	10
notify	Eingang von neuer Mail sofort melden	11

O

od	Oktal Dump	2
----	------------	---

P

pack	Datei komprimieren	8
passwd	Kennwort eingeben	1
paste	Dateien nebeneinander ausgeben	8
pcat	Mit pack komprimierte Datei ausgeben	8
pg	Seitenweise Ausgabe von Dateien	2
ping	Verbindung zu entferntem System testen	
pr	Druckformatierung von Dateien	2
ps	Prozess-Status ausgeben	1
pwd	Name "Working Directory" ausgeben	1

R

rcp	Dateien auf entferntes System kopieren	
rlogin	Login auf entfernten Rechner	
rm	Dateien löschen	3
rmdir	Verzeichnis löschen	3
ruptime	Im Netz aktive Rechner auflisten	
rwho	Welche Benutzer sind im Netz	

S

script	Protokollierung Befehle und Output	3
sleep	Prozess schläft für angegebene Zeit	10
sort	Dateien sortieren	7
split	Datei teilen	
stty	Terminaleinstellung ausgeben/verändern	7
su	Switch User	
sum	Prüfsumme für Datei berechnen	

T

tabs	Tabulatorpositionen Terminal setzen	
tail	Hintere Zeilen Datei anzeigen	2
talk	Mit anderem Benutzer kommunizieren	14
tar	Kopieren auf/von Band oder Diskette	6
tee	Ausgabeverdoppelung T-Stück	3
test	Test Ausdruck	9
telnet	Remote Login	
tftp	einfaches ftp	
tput	ESC-Sequenzen Terminalcapabilities holen	7
tset	Terminaleinstellung	7
time	Zeitmessung für Programme	
touch	Zeitstempel für Dateien	3

tr	Übersetzung von Zeichen	11
tty	Terminalnamen ausgeben	7
ttymap	Mapping Keycodes	7

U

umask	Permissionmaske setzen/ausgeben	7
uname	Systemname anzeigen	7
uncompress	Dekomprimierung Dateien	8
uniq	Identische Zeilen nur einmal ausgeben	
unpack	Dekomprimierung Dateien	8
untic	Escape Sequenzen für Terminaltyp anzeigen	7
uucp	Kopieren von Dateien über Unix-Netz	
uulog	UUCP-Logdatei ausgeben	
uuname	Über Netz direkt angeschlossene System ausgeben	
uupick	Datei abholen, die mit uuto kopiert wurde	
uustat	Statusinfo von uucp- und uuto-Aufträgen	
uuto	Kopieren von Dateien	
uux	Kommando auf Remote-Rechner ausführen	

V

vi	Editor	
----	--------	--

W

wall	Nachricht an alle Benutzer	14
wc	Worte, Zeilen, Zeichen iener Datei zählen	7
who	Aktive Benutzer auflisten	7
who am i	Aktive Benutzer auflisten	7
write	Nachricht an anderen User senden	14

Z

zcat	mit compress komprimierte Datei anzeigen	8
------	--	---

Benutzerumgebung

chsh # ändert Loginshell (greift auf /etc/shells zu (AIX /etc/security/login.cfg))
id # gibt User-ID und Group-ID aus **uid=100(hahn) gid=(1(other))**

passwd [name] Kennwort für Benutzer ändern ([name] su only)
-f Setzt **finger**-Informationen (DEC,AIX)
-s Change Shell entspricht **chsh (DEC,AIX)**

passwd -s [name] Show Passwortattribute für name
-d Delete Passwort Entry,
-l Lock Entry für name,
-x# Setze Maximum Feld (wenn #=0 erfolgt kein aging),
-n# Setze Minimum Feld (minmale Anzahl Tage zwischen Änderungen)
-f Benutzer muss beim nächsten Login neues Passwort eingeben
-s Show Passwortattribute in Form: name status mm/dd/yy min max
status=> PS=Passworded oder Locked LK=Locked, NP=No Password
-a alle Entries zeigen nur n Verbindung mit -s und ohne [name]

Anzeigen

cd [directory] # Change Working Directory, wenn directory nicht angegeben wird das Working Directory auf \$HOME gesetzt. Wurde ein unvollständiger Pfad (ohne . oder ..) angegeben, so wird das Directory auf \$CDPATH gesucht (analog zu PATH)

ls [-alogsitucRdbmCx] [names] # Inhaltsverzeichnis anzeigen
-a alle Dateien, **-l** Long Format, **-o** wie **-l** ohne owner, **-g** wie **-l** ohne group,
-n wie **-l** group und owner wird jedoch numerisch gezeigt,
-s Anz. 512 Byte Blöcke (incl. indirect Blocks),
-i inode anzeigen,
-t Sort nach Zeit (latest first) statt nach name (Def. Modification time),
-u Sort last access, **-c** last modifc. inode, **-r** Reverse Sortierfolge,
-R rekursiv Subdirectories anzeigen, **-d** Status Directory,
-b Non-Printable Char als \ddd anzeigen, **-m** Stream Output durch ',' getrennt
-C spaltenweise Sortierung (Def. für lc), **-x** Multicolumn Output Sortiert

ps [-options] # Process Status reporten
-a alle Prozesse (Ausnahme Group-Leader und Prozesse die keinem Terminal zugeordnet sind)
-e über jeden Process (every)
-f Full Listing
-l Long Listing
-n name über anderes System (Su only),
-t termlist Prozesse die zum Terminal gehören
-p procllist Prozesse deren ID in Procllist angegeben wurde
-u uidlist Prozesse deren UID oder Loginname angegeben wurde
-g grplist Prozesse deren Groupleader-ID angegeben wurde (ID Loginshell)

pwd # gibt Pfadnamen des 'Working Directory' aus

Zugriffsrechte

chmod ugoa+-=rwx fln # Zugriffsmode setzen oder ändern

u Eigentümer (user)
g Gruppe (group)
o Rest der Welt (other)
a Alle 3 Benutzergruppen

+ Hinzufügen
- Wegnehmen
= Setzen

r Leserechte
w Schreibrechte
x Ausführungsrechte

Beispiel: **chmod a=rwx myfile; chmod u+rwx,go-w myfile**
Die Rechte können auch als Oktalzahl angegeben werden

chmod 544 myfile
chmod 755 myfile

+slt 4000 s set User-ID bit on execution **chmod u+s myfile**
20#0 l set Group-ID Bit on execution, wenn #=7,5,3 oder 1
chmod a=rwx, go-w, g+s myfile; chmod 2755 myfile

owner	group	other
s l t	r w x	r w x
4 2 1	4 2 1	4 2 1

enable mandatory file
locking, wenn #=6,4,2 oder 0
1000 t sticky Bit setzen, wenn
auf Verzeichnis gesetzt, dann kann nur

Eigentümer der Datei löschen.

4000 s Es wird user oder group des Verzeichnisses als Dateieigentümer eingetragen unabhängig vom erzeugenden Prozess.

chgrp [-R] group fln ... # neue Gruppe für Datei (group kann Dezimal oder groupname aus /etc/group sein) (Sys V su-only) -R rekursiv

chown [-R] owner fln ... # ordnet den Dateien neuen Owner zu (greift auf /etc/passwd zu) (Sys V su-only) -R rekursiv über alle Unterverzeichnisse

Ausgeben

cat [*opt*] *file...* # Dateien auf stdout ausgeben (concatenate and print files)

- u Daten ungepuffert ausgeben (Default buffered)
- s keine Fehlermeldung bei nicht existierenden Dateien (silent)
- v nicht druckbare Zeichen oktal anzeigen (Ausnahme Tabs)
- t Tabs als ^I und Form Feed als ^L anzeigen (erfordert -v)
- e am Zeilenende \$ anzeigen (erfordert -v)

head [*option*] [*datei*] gibt Anfangszeilen von Datei auf stdout aus (siehe auch **tail**)

hd [-*format*][-*s* *offset*][-*n* *count*] [*file ...*] # Hex-Dump (siehe auch **od**)

-*format* **acbwla** Spezifikation der Ausgabe als Adresse, Character, Bytes, Words, Long, ASCII.

xdo Ausgabeart x=Hexa, d=Dezimal, o=Oktal

t Ausgabeart Text, vor jeder Zeile steht Adresse

gesetztes High-Bit steht als ~ (Tilde) vor Zeichen
0x00 bis **0x1F** werden als ^@ bis ^_ ausgegeben.

-*s* *offset* Anfangspunkt der Ausgabe. *offset* kann Dezimal, Hex 0x... oder Oktal (mit 0...) angegeben werden. Dahinter kann ein 'Multiplifier' b w l k angegeben werden (byte,word,long,K-Byte)

-*n* *count* Anzahl der auszugebenden Bytes. (Angabe *count* wie *offset*)

od [-*bcdosx*] [*file*] [[+]*offset*.[*b*]] # gibt file oder stdin in einem oder mehreren Formaten als Dump aus (od -xc *datei*). (siehe auch **hd**)

-b Bytes oktal ausgeben

-c Characterform, nichtdruckbare Zeichen erscheinen in C-Notation

null=\0,Backspace=\b,Form Feed=\f,New Line=\n,Return=\r,Tab=\t
andere als \nnn (3stellige Oktalzahl).

-d Words als unsigned decimal

-o Words als oktal

-s Words als signed decimal

-x Words als hex

+**offset.b** Anfang des Dump. i

+ nur erforderlich, wenn file nicht angegeben

. Angabe ist Dezimal (Default Oktal), b Angabe ist in Blocks a 512

pr [*options*] *file* [*file...*] # Ausgabe in aufbereiteter Form auf stdout

-o# *Offset* zum linken Rand, -l# *Seitenlänge* (Def 66), +# *Anfang* mit Seite#

-n# *Zeilennummern* #=*Anz*= *Ziff.*, -f *Formfeed* statt Blanklines, -t no headings

-h *text* *Headertext*, -p *Pause* bei Ausgabe zum Terminal, -d *Double spacing*

pr +3 -o3 -f -n2 myfile # ab seite 3 offset 3 Zeilennummern

pr -2 -o3 -w160 -f -t -l60 instant_usr >inst.txt # Text 2 spaltig

pg [*opt*] *file* [*file...*] # seitenweise Ausgabe einer Datei ? bringt Help

-n *Windowgröße* (Default 23 auf Terminal mit 24 Lines)

-p *string* *string* ist Prompt. z.B pg -p "Seite %d :." myfile

-c *Cursor Home* und *Clear Screen* am Seitenanfang

-e *Keine Pause* bei Dateieende

-f *Zu lange Zeilen* nicht splitten

-s *Messages* und *Prompts* in Standout-Mode (gewöhnlich Revers Video)

+*line* *Start* bei Zeile *linenumber*

+*/pattern/* *Start* bei der ersten Zeile die das 'regular pattern' enthält

pg	zeigt einen Bildschirm an und promptet den Benutzer	
:	ist der Prompt, folgende Kommandos sind möglich	
h	dieser Hilfebildschirm	help
q or Q	beenden	quit
<blank> or \n	nächste Seite	next page
l	nächste Zeile	next line
d or ^D	nächste halben Bildschirm	display half a page more
. or ^L	Bildschirm auffrischen	redisplay current
page		
f	nächste Seite überspringen	skip next page forward
n	nächste Datei	next file
p	vorherige Datei	previous file
\$	letzte Seite	last page
w or z	Bildschirmgröße setzen	window size
s fln	augenblickl. Datei in fln sichern	save current file in fln
/pattern/	suche vorwärts nach pattern	search forward for pattern
?pattern? or	suche rückwärts nach pattern	s earch backward
for pattern		
^pattern^		
!command	führe command durch	execute command
die meisten Kommandos kann eine Zahl vorangestellt werden		
+1\n (nächste Seite); -1\n (vorherige Seite); 1\n (1. Seite)		

more file #
seitenweise Ausgabe

einer Datei ? bringt Help

less file # **LINUX** seitenweise Ausgabe einer Datei

tail [*option*] [*datei*] gibt hinteren Zeilen der Datei aus (siehe auch **head**)

Datum

cal [*mm*] [*jjjj*] # Kalender ausgeben

calendar [-] # Terminüberwachung mit Datei .calendar, [-] Mail erzeugen

date *mmdhmm*[[*yy*] | [*ccyy*]] # Systemdatum setzen (**su only**)

date 912311201994 #neuse Systemdatum 23.01.1994 11:20

date [+*format*] # Datum und Uhrzeit ausgeben +*format* steuert Komponenten

%a Abk. W-tag, %A voller W-tag, %b Mon., %B Monat, %D mm/dd/yy,
%d 01-31, %e 1-31, %m 01-12, %y 00-99, %Y 1991, %j lfd Tag 001-366
%r hh:mm:ss pp (pp=AM|PM), %T hh:mm:ss, %R hh:mm, %H 00-23, %M 00-59,
%w W-tag (0=Sonntag), %U 01-52 (lfd W Mo 1.Tag), %W 01-52 (lfd W So 1.Tag)
%Z Timezone, %p (AM|PM), %n new line, %t tab
%x Country spezific Date, %X Country spezific Time

Dateien und Verzeichnisse

basename *file* [*\.ext*] # gibt nur den reinen Dateinamen zurück, unterdrückt den Path /usr/hahn/instant => instant (siehe auch **dirname**)
cc \$1; mv `basename \$1 [\c]` /\$HOME/bin

dirname *file* # gibt nur den reinen Path zurück, unterdrückt Dateinamen
set \$1=/usr/hahn/instant; dir=`dirname \$1` (siehe auch basename)

rm [-f] [-i] **file** # Dateien löschen,
-f force (auch schreibgeschützte)
-i Interaktiv Genehmigung einholen
rm -r [-f] [-i] **dirname** # Alle Dateien und Subdirectories unter **dirname** rekursiv löschen

rmdir [-p] [-s] **dirname** ... # löscht leeres Directory
-p auch Parentdirectory löschen wenn es durch **rmdir** leer wird
-s Nachricht unterdrücken das Parentdir gelöscht wurde

mkdir [-m mode] [-p] **dirname** ... # Inhaltverzeichnis anlegen
-m mode Schutzcode für neues Directory (oktal wie bei chmod)
-p Fehlende Unterverzeichnisse anlegen

script [-a] *fln* # erzeugt Protokoll der Terminalsitzung beenden mit ^d
-a append an *fln* (AIX,ULTRIX, HP,LINUX)

touch *fln* [...] # Setzt Modificationtime, legt neue Datei *fln* an wenn nicht vorhanden
[-acm] [MMDDhhmm[yy]] *file* ... ändert Datei auf angegebene Zeitmarke

Kopieren

cp [-ifrp] *file1* [*file2*] **target** # **target** kann Dateiname oder Directory sein
Wenn [*file2*...] verwendet wird, **mus** **target** ein Directory sein.
-i Wenn Zieldatei existiert, wird nachgefragt
-f Keine Rückfragen, wenn Schutzcode nicht passt
-r rekursiv kopieren
-p permission-Bits erhalten

Link

ln [-f] *file1* [*file2*...] **target** # Hardlink herstellen.
Wenn [*file2*...] verwendet wird, muss **target** ein Directory sein.
-f Keine Rückfragen, wenn Schutzcode nicht passt
-s Symbolic Link

Mixed

banner *text* # gibt *text* <= 10 Zeichen in Grossbuchtaben auf stdout aus

cmp [-l] [-s] *file1* *file2* # vergleicht die beiden Dateien gibt Zeilennummer und Byte der Differenz zurück (Default nur 1. Unterschied).
-l # Bytenummer Dezimal und Bytes Oktal zurückgeben (erkennt

alle Bytes als unterschiedlich wenn nur erstes Byte anders)

-s # nur Status zurückgeben

cs [-cfinstvVxX] [*args* ...] # Aufruf C-Shell

/bin/echo -n *args* # Ausgabe der Argumente auf stdout, -n no Line Feed
"args" #\b backspace, \c no line feed, \f form feed, \n new line
\r carriage-return, \t tab, \v vertical tab, \ backslash
\0n n=1 bis 3stellige Oktalzahl

ed [-p ">"] *file* # Line Editor aufrufen

env [-] [*name=value*] ... [command [*args*]] # führt das Kommando aus, fehlt command wird das Environment angezeigt
name=value modifiziert für das Kommando das Environment indem die angegebenen Variablen mit eingefügt werden
- Es werden keine Variablen aus dem Environment übernommen

tee [-i] [-a] [*file*] ... # kopiert stdin auf stdout und die angegebenen Dateien.
-a Die Ausgabe wird an die Dateien angehängt (append)
-i Interrupts ignorieren

Dateityp ermitteln

file [option] *fln*... # untersucht *fln* und gibt Typ der Dateien aus
-c prüft nur magicfile es werden keine Dateien untersucht,
-f **file** Dateinamen stehen in **file**
-m **file** alternative magicfile verwenden (statt /etc/magic)

Hilfe

man [opt] [section] *titles* # Manualpages auf stdout anzeigen
1-8 /usr/catman/?_man/man[1-8]*
-T **term** Ausgabe für Terminaltyp **term** aufbereiten
-w nur Pfadnamen relativ zu /usr/catman ausgeben
-d Current Dir durchsuchen statt /usr/catman
-c für Aufbereitung col(1) aufrufen

Drucken System V

lp [options] *file* [*file* ...]
-c # Datei kopieren nicht aus Originaldatei drucken
-n number # Anzahl der Drucke
-m # Mail schicken wenn Druckvorgang beendet ist
-w # Message zum Terminal wenn Druck beendet
-s # Message unterdrücken z.B "request-id is ..."
-q prio # Druckauftrag mit angegebener Prio durchführen (0=high)
-t title # Titel für Banner, wird als \$3 ins Interface übergeben
-f form-name [-d any] # Verwende für den Ausdruck das Form form.name
-d any jeder Drucker, der form-name unterstützt, kann verwendet werden.
-i # Optionen nachträglich ändern, wenn Job bereits in Druckphase wird er angehalten und neu vom Anfang neu aufgesetzt, (wenn nicht -P)
-H *spec-hand* # Special Handling

hold Druck zurückstellen, bis Freigabe durch resume erfolgt
resume Freigabe z.B **lp -i laser-285 -H resume**
-o depend-opt # Printer- oder Class-abhängige Option müssen im Interface
behandelt werden (werden nur durchgereicht)
getprops(C) zeigt verfügbare Optionen
nobanner Banner-Page unterdrücken (kann vom su unterbunden werden)
nofilebreak Druckdateien nicht durch Seitenvorschub trennen
-P page_list nur angegebene Pages drucken (erfordert Filter das das behandelt)

lpstat [options] # Statusinformationen ausgeben
-a [list] # acceptance status, [list] Printer oder Classnames (Def all)
-c [list] # Classnames und Members anzeigen (Def. all)
-d # Default Destination anzeigen
-f [list] [-l] # verifiziere das Forms in list akzeptiert werden
-o [list] [-l] # Status der Outputrequests (Def. all), -l =Long Form
-p [list] [-D] # Status der in list angef. Drucker zeigen, -D =Kurze Form
[-l] # Long Form incl. mounted Forms,Content,Printertype,Interface
-r # Status des LP-Schedulers
-s # Status Summary
-S [list] [-l] # verif. das Character Sets und Print-wheels akzept. werden
-t # Alle Statusinformationen
-u [list] # Status Outputreq. für user in list (def. all)
-v [list] # Namen der Printer, Pfade der zugehörigen Devices (Def. all)

cancel [ids] [printers] # Druck abbrechen (id wird bei lp und lpstat geliefert)

getprops [-v] [-p printer | -m model] zeigt Optionen (SCO only)

Primär Superuser , bei **SCO** auch möglich wenn User lp authorization hat siehe authorize(F)
subsystem(M)
enable printers # Aktiviert den bezeichneten Drucker
disable [opt] printers # Deaktiviert den bezeichneten Drucker, augenblicklich
druckende Requests werden vollständig auf dem gleichen
Drucker oder einem Drucker der Class ausgegeben.
Es werden weiter Druckrequest in die Queue aufgenommen
-c Cancel current Request (nicht in Verb. mit -W)
-r reason Die Begründung für disable wird gespeichert und dem
Benutzer bei lpstat angezeigt
-W Der Drucker wird erst nach Beendigung des laufenden Requests disabled
accept printers # Request für die Queue/Class werden akzeptiert
reject [-r [reason]] printers #Request für die Queue/Class werden nicht mehr angenommen
reason wird von lpstat angezeigt.

/usr/lib/lpshut # Stop Printerservice
/usr/lib/lpsched # Start Printerservice, nicht mit kill beenden, da sonst
Lockfile erhalten bleibt
/usr/lib/lpmove request dest # Druckauftrag zu anderem Drucker umsetzen
/usr/lib/lpmove dest1 dest2 # Alle Aufträge umsetzen dest1 nimmt keine

/usr/lib/lpusers -d prio # systemweite Default-Prio für Druckauftrag
setzen Prio 0 high 39 low (Default 20)
/usr/lib/lpusers -q prio # Maximal anforderbare Prio für alle Benutzer
die nicht explizit aufgeführt sind
/usr/lib/lpusers -q prio -u user-list # Max anfordb. Prio für angef. User
/usr/lib/lpusers -u user-list # angef. User wieder auf Def.-Prio setzen
/usr/lib/lpfilter -f filtername -l # Alle Filter anzeigen
/usr/lib/lpfilter -f filtername -x # benannten filter löschen
/usr/lib/lpfilter -f filtername -i # benannten filter Initialwerte holen
/usr/lib/lpfilter -f filtername -F path # Filter installieren
#wird anstelle von filtername **all** eingesetzt, bezieht
sich die Operation auf alle Filter z.B
/usr/lib/lpfilter -f all -l
lpadmin -p printer #Administration printer (SCO only)
-c classname # erzeugt Klasse
-m model # Model z.B. POSTSCRIPT, laser
-D comment # Kommentar
-v device # Zuordnung eindeutiges Device z.B. /dev/lp0

Suchen in Dateien

grep [opt] *limited-regular-expr* [file...] # durchsucht nach pattern

- b # Blocknummer vor Ausgabezeile schreiben
- c # nur Zeilenanzahl der Zeilen ausgeben die das pattern enthalten
- i # Lower/Upper Case ignorieren
- h # Dateinamen nicht anzeigen
- l # Dateinamen mit 'matching lines' nur einmal zeigen
- n # Zeile mit Zeilennummer anzeigen (erste Zeile 1)
- v # alle Zeilen zeigen die das Pattern nicht enthalten
- s # keine Fehlermeldungen für nicht lesbare Dateien

fgrep [opt] *string* [file ...] # durchsucht nach festen Strings

- x nur Zeilen anzeigen die komplett übereinstimmen
- e spec_string String beginnt mit -
- f file Die Strings werden aus file entnommen
- bcihlv diese Optionen sind wie bei grep (-s fehlt)

egrep [opt] *full-regular-expression* [file...] # durchsucht nach pattern.

- 'pattern' darf Metazeichen enthalten.
- e spec_string String beginnt mit -
- f file Die Strings werden aus file entnommen
- bcihlv diese Optionen sind wie bei grep (-s fehlt)

Suchen nach Dateien

find start-dir [*suchmuster*] [*ausgabeart*]

durchsucht den Baum rekursiv ab dem Startdirectory nach Dateien die dem Suchmuster entsprechen, dabei dürfen die Muster Metazeichen enthalten.

Da sie an das find übergeben werden, müssen sie maskiert werden.

In den Suchmustern kann n angegeben werden als:

n (exakt n), +n (mehr als n), -n (weniger als n)

Suchmuster sind:

- name *file* True wenn der Dateiname mit file übereinstimmt.
find . -name '*.*' -print
- user *username* True wenn die Datei dem user username gehört. username kann numerisch angegeben werden, wird als ID verwendet.
- group *grpname* True wenn die Datei zur Gruppe grpname gehört
- links *n* True wenn Datei n Links hat
- type *c* True wenn die Datei dem Typ c entspricht. 'c' kann sein:
b=Block-special, c=char-spec., d=directory, p=fifo oder f=plain-file (normale Datei)
- size *n[c]* True wenn Datei n Blocks lang ist (a 512 Bytes)
folgt auf n der Buchst. 'c' ist die Angabe in Character
- atime *n* True wenn auf die Datei innerhalb n Tagen zugegriffen wurde. (find ändert die access-time von Directories)
find . -atime +30 -print #last access mehr als ein Monat
- mtime *n* True wenn die Datei innerhalb n Tagen modifiziert wurde.
- ctime *n* True wenn i-node innerhalb n Tagen modifiziert wurde

- newer *file* True wenn die Datei aktueller als file ist
- perm [*-*] *onum* True wenn die Permissionbit exakt mit onum (octal) übereinstimmen.
-perm 755 # exakt 755
Steht vor onum ein - (minus) werden nur die gesetzte Bits geprüft
d.h. die Bits werden als Maske mit XOR mit den Rechten verknüpft.
-perm -002 # mindestens r Rechte für other
-perm -4000 # SUID-Bit gesetzt
- depth Immer True, Es werden erst alle Files in der Directory bearbeitet, bevor die Directory selbst bearbeitet wird.
Ist nützlich wenn Daten mit cpio gesichert werden sollen, im Directory aber keine Schreiberlaubnis gegeben ist
- prune nicht in Unterverzeichnisse verzweigen
- mount Immer True, die Suche wird auf das Filesystem begrenzt in dem das Startdirectory enthalten ist.(DIGITAL,SCO,IRIX)
- xdev wie -mount (AIX,LINUX)
- local True wenn die Datei im lokalen System ist
- fstyp *typ* Filesystem ist vom angegebenen Type z.B. -fstyp nfs
- ! kann zur Verneinung von Suchmustern verwendet werden.
Werden mehrere Suchmuster verwendet, so sind sie implizit per AND verbunden
eine ODER Verbindung kann durch -o erzwungen werden. () Klammern ändern die Auswertungsfolge.

Ausgabeart

- print Der Pfadname wird ausgegeben
- exec *cmd* Das Kommando cmd wird ausgeführt. cmd muß durch ein maskiertes Semikolon abgeschlossen werden. Falls das cmd den Pfadnamen als Argument verwendet, muß mit {} ein Platzhalter definiert werden
find . -name '*.*' -atime +20 -exec rm {} \;
- ok *cmd* Wie -exec, jedoch muß der Benutzer die Ausführung genehmigen (durch Eingabe von y)
- cpio *device* Die Datei wird im cpio Format (512-Byte Records) auf Device geschrieben

Dateibäume sichern

cpio -o[acBvV] [-C *buffsiz*] [[-O *file*] [-M *message*]]

- o # liest von stdin und schreibt auf stdout (Save) erzeugt Container
cd /usr/hahn; find . -print | cpio -ocv > /dev/dsk/f0q18dt
- a Reset access time
- c Write header in ASCII-Form (aus Portabilitätsgründen)
- B Input/Output in Blöcken a 5120 (nur gültig für Char-Spec-Device)
- v verbose, zeige alle Dateinamen
- V Special Verbose, zeige für jeden Dateinamen einen '!'
- C *buffsize* kann nur für char-Spec-Dev angegeben werden
- M *message* Nachricht für Anforderung neues Medium

cpio -i[BcdmrtuvVfsSbk] [-C *buffsize*] [[-I *file*] [-M *message*]] [*pattern*]

- i # liest von stdin und legt Dateien im Current Directory ab (Restore)
(liest aus Container)
cd /tmp; cpio -icvBdum < /dev/dsk/f1q15dt #liest LPI-Install-Backup
 - d Directories erzeugen falls erforderlich
 - u Unconditional, neuere Dateien werden überschrieben
 - m Modification Time soll erhalten bleiben (gilt nicht für Directories)
 - r Rename Dateien interaktiv
 - t Nur Inhaltsverzeichnis ausgeben
 - s Swap Bytes innerhalb Halbwords
 - S Swap Halbworte
 - b Reverse Reihenfolge der Bytes innerhalb Word
 - k Skip corrupted files und header (Last Chance bei schlechtem Medium)
- Default pattern ist *, Multiple Pattern können angegeben werden
dabei ist jedes Pattern in "Double Quotes" einzuschliessen.

cpio -p[adlmuvV] *zieldirectory* #

- p # liest von stdin und kopiert die Dateien und Dir's in das *zieldirectory*. Erzeugt Baum
cd ; find . -print | cpio -pvd /tmp/sicher)
- d Directories erzeugen falls erforderlich

tar [key] [files] # sichert und restores Dateien auf Floppy, Tape oder normale Dateien
key # Funktionsangabe und weitere Optionen

Funktion

- c neues Archiv erzeugen (create)
- r Dateien am Ende des Archivs anhängen
- u Dateien einfügen wenn noch nicht vorhanden oder neuer (update)
- x Dateien aus Archiv extrahieren
- t Dateinamen anzeigen (list)

Option

0,...7 an eine der Funktionen angehängt spezifiziert das Laufwerk
in der Datei /etc/default/tar (Default Laufw. 1) (SCO,INTERACTIVE)

Optionen

- bn n gibt den Blockungsfaktor an (1 <=n <= 20, Default 2)
sollte nur mit raw-Tape-Devices verwendet werden und
einem neu anzulegenden Archiv (d.h nicht mit r und u)
- v Verboseinformationen (Dateiname wird angezeigt)
- f nächstes Argument gibt Archivdatei statt Defaultdevice an
(Defaultdevice ist in /etc/default/tar festgelegt)
wenn statt Dateiname - angegeben wird, wird je nach Funktion
stdin oder stdout angesprochen

-C *dir* DIGITAL führt cd zu angegebenen Verzeichnis aus. Kann mehrfach vor-
kommen. Damit können mehrere Verzeichnisse relativ gesichert werden
weitere Optionen im Systemhandbuch nachschlagen (sind teilweise unterschiedlich)

tar -cvf /tmp/\${USER}_tar . # Alle Dateien ab current dir in Containerfile sichern

tar -cvf /dev/fd0a . # Alle Dateien ab current dir auf Diskette sichern

cd sourcedir; tar -cf - . | (cd destdir; tar -xf -) # Baum übertragen (Subshell erforderlich)

Diskspace

df [-fltv] [*file-system* | *directory* | *mouted-resource*]

zeigt benutzte und freie Plattenkapazität und inodes an.

- f benutzt free-list nicht den Superblock
- l zeigt nur lokale Ressourcen (keine Remote)
- t Total allocated Blocks und I-Nodes
- v Prozentuale Angaben der benutzten und freien Blöcke

du [-sar] [*dir*] # Disk Usage in Blöcken (incl. indirect Blocks)

- s nur grand total,
- a eine Zeile pro Datei,
- r Message für Directories die aus Schutzgründen nicht lesbar waren

Informationen über System und User

umask [ooo] # setzt File-Creation-Mode-Mask. Der Oktalwert wird vom korrespondierenden Digit des Systems 'subtrahiert'
umask 022 führt zu mode 755

uname [-snrvma] # gibt Infos über System
-s Systemname (Default)
-n Nodename
-r Operating-System-Release
-v Operating-system-Version
-m Hardwarebezeichnung der Maschine
-a alle vorherigen Wert

wc [-wcl] [file ...] # Zählen Worte (w), Character (c), Zeilen (l)

who # zeigt Namen, Device und Login-Zeit der Benutzer an

who am i # zeigt eigenen Namen, Device und Login-Zeit an

Terminaleinstellung

clear # löscht Bildschirm, verwendet \$TERM und terminfo

mesg [y|n] # ändert Schutzcode für Terminal in /dev/...
oder zeigt an ob Nachrichten unterdrückt werden

stty [-a] [options] # Leitungsparameter anzeigen [-a] oder setzen
wichtige Options : sane , echo, raw
stty -a; stty -echo; stty eof=^D

tset [option] [type] # Initialisiert, resets Terminal, legt Erase- und Kill-Character für Terminal fest, gibt Informationen über Terminal
- gibt Terminaltype auf stdout aus
-e[c] legt Erase-Charcter fest (Default Backspace oder CTRL/H)
-E[c] wie -e arbeitet nur auf Terminals die Backspace können
-k[c] legt Kill-Character fest (Default CTRL-U)
-Q Unterdrückt die Message "Erase set to ..." und "Kill set to ..."
-h tset soll /etc/ttytype durchsuchen und nicht \$TERM verwenden
-r gibt Terminaltyp auf stderr aus
-s erzeugt je nach Shell ein setnv (C-Shell) oder assignment und export (Bourne-Shell)
-S gibt nur die Strings für die Environmentvariablen aus
-I Unterdrückt die Ausgabe des Initialisierungsstrings d.h erzeugt Process tput reset statt tput init

tty [-l] [-s] # zeigt Device an z.B /dev/vt02
-l gibt die 'synchronous line number' aus, wenn Terminal an aktiver synchroner Leitung hängt
-s gibt nicht den Pfad aus sondern nur Exitcode

2 ungültige Operation
0 stdin ist ein Terminal
1 in allen anderen Fällen (otherwise)

ttymap *mapfile* # Terminalmapping und Scancodeübersetzung (INTERACTIVE) z.B.

ttymap /usr/lib/keyboard/german.map # für Umlaute
-r setzt Scancodes zurück auf US-PC-Keyboard
-d disables current setting
-d Zeichen in string1 löschen
-s Squeeze alle mehrfach vorkommenden Ausgabezeichen die in string2 vorkommen zu einem Zeichen

tput cap holt die Escapesequenz für die angegebene Capability (cap siehe man terminfo)
NORM=\$(tput sgr0); REV=\$(tput rev)
echo "\$(tput cuup 10 60)\c" # Positioniert Cursor Zeile 10 Spalte 60

untic terminaltyp oder **infocmp** terminaltyp (DIGITAL)
zeigt Escapesequenzen für Terminaltyp.
Greift auf Datenbasis terminfo (SYS V) oder termcap (BSD) zu

Sortieren

sort [options] [+pos1[-pos2]]... [inpfiler]...
-d Dictionaryorder, -f fold (Gross/Klein gleich),
-i ignore Zeichen <x20 und >x7e (Steuerzeichen)
-b eading Blanks bei Feldverg. ignorieren,
-r reverse order,
-tx Zeichen x Trennzeichen zwischen Feldern,
-n numerischer Sort
-c Ccheck only, -m Merge, -u Doppelte Entries unterdrücken (unique)
-o name Ausgabe nach name statt stdout
+pos Anzahl zu überspringende Felder in Form m[.n] (Default 0)
sort -t: +2n /etc/passwd
-pos Letzte Stelle des Sortierbegriffs
sort -t: +2 -3 sortdat

Ersetzen/Umformen

cut -clist [*file...*] # angegebene Spalten ausschneiden (1-8,12,15-20)

cut -flist [-*dchar*] [-*s*] [*file...*] # angegebene Felder ausschneiden

-*d* ' ' Blank als Trennzeichen, -*s* Zeilen ohne Trennzeichen unterdrücken

cut -f1,5 -d: /etc/passwd

paste [-*s*] [-*dlist*] [-] *file1 file2 ...* # fügt Dateien zusammen und gibt auf stdout aus,
Trennung erfolgt durch ein \t (Tab-Zeichen)

- von stdin lesen **ls | paste - - - -** (in 4 Spalten ausgeben)

-*dlist* legt das Trennzeichen fest, nur die letzte Datei erzeugt immer
ein \n (new line) **paste -d:\v a b c > xyz**

-*s* fügt erst Zeilen aus *file1* zusammen, dann aus *file2* usw

tr [-*cds*] [*string1* [*string2*]] # kopiert stdin nach stdout, ersetzt dabei

Zeichen die in *string1* gefunden werden durch Zeichen aus *string2*.

tr "015" < tryin > tryout

[*a-z*] definiert einen Bereich im String

tr "[a-z]" "[A-Z]" <klein >GROSS

[*a*n*] steht für eine n-malige Wiederholung von a, fehlt die

Angabe von n, so wird sie als 'huge' angenommen, das

ist für das Auffüllen von *String2* sinnvoll.

tr -cs "[A-z][a-z]" "[012*]" <file1 >file2

-*c* Complement die Zeichen die in *String1* vorkommen

-*s* für gleiche hintereinander vorkommende Zeichen in *string1* nur ein Zeichen
in *string2* ausgeben

-*d* alle Zeichen löschen, die in *string1* vorkommen (*string2* dann nicht sinnvoll)

Komprimieren/Dekomprimieren

pack [-] [-*f*] *datei* [...] Dateien komprimieren nach *datei.z* (**Achtung ersetzt in Place**)

-*f* Datei auch dann komprimieren, wenn kein Platz gespart wird

unpack *datei.z* [...] *.z* Datei dekomprimieren (Achtung ersetzt in Place)

pcat *datei.z* [...] *.z* Datei anzeigen

compress [-*f*] *datei* [...] Dateien komprimieren nach *datei.Z* (**Achtung ersetzt in Place**)

-*f* Datei auch dann komprimieren, wenn kein Platz gespart wird

uncompress *datei.Z* [...] *.Z* Datei dekomprimieren (Achtung ersetzt in Place)

zcat *datei.Z* [...] *.Z* Datei anzeigen

expr

expr *argumente* # expr verwendet die angegebenen Argumente als Expression, bearbeitet sie und gibt das Ergebnis nach stdout aus.

expr1 || **expr2** # Rückgabe expr1 wenn er nicht "0" oder 0 ist, sonst expr2

expr1 & **expr2** # Rückgabe expr1 wenn keiner der beiden expr "0" oder 0 liefert, ansonsten wird 0 zurückgegeben.

expr1 { =, >, >=, <, <=, != } **expr2**

Wenn beide expr Integerausdrücke sind wird das Ergebnis eines Integervergleichs zurückgeliefert, andernfalls das Ergebnis eines lexikalischen Vergleichs.

Beachten Sie die teilweise Maskierung der Operatoren.

expr1 { +, - } **expr2** # Addition bzw Subtraktion von Integerwerten.

wert=`expr \$wert + 1` # ((wert=wert+1)) **kornshell**

expr1 { *, /, % } **expr2** # Multiplik., Div. bzw Divisionsrest der Integerwerte

res=`expr \$wert * 2` # ((res=wert * 2))

expr1 : **expr2** # Der Matching-Operator : vergleicht den expr1 mit expr2.

Dieser (expr2) muss ein Regular Expression sein.

Seine Syntax entspricht den Regular Expression aus ed(1).

Ausnahme, alle Muster werden als "anchored" (beginnend mit ^) betrachtet. Deshalb ist ^ hier kein Special-Char.

Zurückgegeben wird die Anzahl der übereinstimmenden Zeichen oder 0. **len**=`expr "\$var" : ".*"`

\(\) definiert Substring

test **expr** # Alte Form

[**expr**] # Evaluiert den expr und gibt Exitstatus zurück

Operator TRUE wenn file existiert :

-r file und lesbar

-w file und schreibbar

-x file und executable

-f file und normale Datei

-d file und Directory ist

-c file und Character-Special

-b file und Block-Special

-p file und Named-Piped

-u file und User-ID Bit gesetzt

-g file und Group-ID Bit gesetzt

-k file und Sticky-Bit gesetzt

-s file und Dateigrösse > 0

-t [descr] wenn Descriptor mit Terminal verbunden und offen ist

Stringoperator TRUE wenn :

-z s1 Stringlänge s1 = 0

-n s1 Stringlänge s1 nicht 0

s1 = s2 beide Strings gleich sind

s1 != s2 die Strings ungleich sind

s1 kein Nullstring ist

Algebraische Vergleichsoperatoren

n1 -eq n2 gleich

n1 -ne n2 ungleich

n1 -gt n2 größer als

n1 -ge n2 größer oder gleich

n1 -lt n2 kleiner als

n1 -le n2 kleiner oder gleich

Die vorstehenden 'Primaries' können mit folgenden Operatoren verbunden werden,

! Negation (! in Blanks einschliessen)

-a AND-Verbindung

-o OR-Verbindung (-a höheren Vorang)

() Klammern verändern Reihenfolge der Auswertung, evtl quoten \(expr \)

Periodische Prozesse

at *time [date] [+ increment]* startet einen Prozess zur angegebenen Zeit
liest von stdin **at 10:02 < pause_10**
 /usr/lib/cron/at.allow und /usr/lib/cron/at.deny
 beeinflussen Personenkreis, der at ausführen darf

time kann mit 1, 2 oder 4 Ziffern angegeben werden
1, 2 Ziffer geben Stunde an (am, pm möglich)
4 Stunde und Minuten (alternativ auch durch 14:25)
Spezielle Werte sind : **noon, midnight, now, next**
at now + 1 day

date Das Datum kann angegeben werden durch :
at 14:30 Nov 28 Monat und Tag
at 08am monday Wochentag oder at 08am mon
at 1700 today oder at 1700 tomorrow
fehlt date wird
 today angenommen wenn die Zeit > current time ist
 tomorrow angenommen wenn Zeit < current time ist

+increment Zahl gefolgt von **minutes, hours, days, weeks, months**
oder years. (Angabe kann auch im Singular erfolgen)

echo "sh scriptfile" | at 1300 monday next week

at -r job ... löscht Jobs die durch at oder batch scheduled wurden
at -l [job ...] zeigt alle scheduled Jobs für den aufrufenden User

batch # liest Kommandos von stdin, führt aus wenn Zeit vorhanden
 batch
 sort fln > outfln
 <CTRL-D>

crontab [file] # fügt Datei in Directory für 'user-crontabs' ein
Achtung : Jobs in crontab können nicht von Terminal stdin lesen
 und nicht auf Terminal stdout schreiben (umleiten).
 EDITOR=/usr/bin/vi #erfordert vollen Path
 (Einschränkungen siehe Dateien cron.allow cron.deny)

-l zeigt 'crontabs' für den aufrufenden Benutzer
-r löscht Eintrag in crontab
-e editiert crontab des Benutzers. Nicht cp *file* /var/spool/cron/crontabs/*user*
im Verzeichnis ablegen, da sonst cron die Änderung nicht mitbekommt.

Aufbau **file**: min std MonTag(1-31) Mon(1-12) WoTag(0-6) Kommando
00 17 * * 1,2,3,4,5 /usr/hahn/feierabend.x

/var/spool/cron # Hauptverzeichnis für cron
/var/spool/cron/crontabs # Enthält für jeden benutzer ein Verzeichnis für
 aufgesetzte Einträge

/var/spool/cron/cron.allow # enthaltene User dürfen Einträge in crontabs machen
/var/spool/cron/cron.deny # verbietet Einträge in crontabs

SCO

/usr/lib/cron/log Logfile wenn in /etc/default/cron CRONLOG YES
/usr/lib/cron/queuedefs enthält Schedule-Informationen für den clock-Daemon
nice-value, reschedule Intervall falls Queue voll

Job Control

bg [job ...] Setzt den angegebenen Job in den Hintergrund (Korn-, C-Shell)
fg [job ...] Setzt den angegebenen Job in den Vordergrund (Korn-, C-Shell)

jobs [-l] zeigt stopped Jobs an
 -l Anzeigen mit PID

kill [-signo] PID ... # sendet Signal 15 (SIGTERM) bzw das angegebene Signal
 zum Process PID. Nur der SU kann fremde Prozesse killen

nice [-incr] command [args] # command wird mit niedrigerer Prio ausgeführt.
 -incr Increment im Bereich 1-19 mit dem die Prio verändert
 wird (Default 10). Nur SU kann erhöhen z.B. --10

nohup command [args] # führe command aus, ignoriere hangups und quits
nohup profile &
 stdout und stderr werden nach nohup.out geschrieben
 wenn sie nicht umgeleitet werden.

sleep n # Schlafzeit in Sekunden

Benutzerkommunikation

mail [-wt] *persons* # Mail an *persons* senden
-w nicht auf Beendigung der Remoteübertragung warten
-t Zeile 'To: persons' zum Brief hinzufügen

mail [-ehpqr] [-f *file*] [-F *persons*] # Mail lesen, ? bringt Hilfe über Kommandos
-e Mail nicht anzeigen, nur Status 0=mail, 1=no mail vorhanden
-h nur Header ausgeben, Benutzer kann dann auswählen
-p alle Mails anzeigen
-q Mail bei Interrupt beendet (Normal nur Message beenden)
-r Reihenfolge first-in first-out
-f *file* Datei *file* benutzen z.B mbox statt Default mailfile
-F *persons* Wenn in leere Mailbox eingegeben, wird ein FORWARD aller einkommenden Mail an *persons* durchgeführt
mail -F"maier,systemb!huber,systemx,hahn"
Forward wird aufgehoben durch : mail -F""

mesg [y|n] # erlaubt oder verhindert schreiben auf eigenes Terminal

notify [-y] [-n] Neue Mail sofort melden -y, nicht melden -n

news [opt] [items] # gibt news aus /usr/news aus, verändert \$HOME/.news_time
-a gibt alle items aus, \$HOME/.news_time wird nicht geändert
-n gibt nur die Namen von neuen Items aus
-s gibt nur die Anzahl von neuen Items aus

talk *login-name* [*terminalname*] Bildschirm wird in 2 Fenster unterteilt. In einem wird geschrieben, im anderen Fenster empfangen. Abbruch mit CTRL-C.
Wenn User an mehreren Bildschirmen Login hat, dann *terminalname* verwenden.

wall *mesg* schreibt Nachricht an alle User (siehe auch mesg)

write [-r] [-n *host*] *user* [*terminal*] Schreibt Nachricht an *user*, EOF durch ^D.
Terminal muß offen sein (**mesg y**).
-r wartet auf Antwort. Erzeugt *handle* für **write -h handle,reply**
-n *host* gibt Remote-Host an. Kann Alias oder IP-Adresse sein
user Benutzername
terminal Wenn User mehrere virtuelle Bildschirme hat, dann *terminal* verwenden.
Mit **finger** (aktives) Terminal ermitteln auf dem Schreibrechte gegeben
* vor Terminalname "no write permission"
SCO **who -T**

write -h *handle,reply* Antwort auf **write -r**
reply kann sein **ok, cancel** oder **query**

write *user@host* alternative Form für write -h

- wenn -n oder *user@host* verwendet wird, muß der **writesrv** daemon auf beiden Systemen gestartet sein und das Verz. **/usr/spool/writesrv** existieren (DIGITAL nicht gestartet)
mkdir /usr/spool/writesrv ; rcmgr set WRITESRV yes ; /sbin/init.d/write start (su only)

write root@fra80 console

Was ist mit dem Printer in Raum 8 ?

Die Queue nimmt keine Aufträge an

^D

Zugriff auf Remote Rechner

rlogin [-l *user*] *hostname* Führt Login auf Remote-UNIX-System durch. Host muss in /etc/hosts vorhanden sein. ~/.rhost ermöglicht autologin
 l *user* Username auf Remotesystem. Ohne diese Angabe wird Login mit gleichem Namen wie auf lokalem System vorgegeben.

telnet [-l *user*] [*hostname*] Login auf Remote-System (muss nicht UNIX sein)
 l *user* Username auf Remotesystem. Ohne diese Angabe wird Login mit gleichem Namen wie auf lokalem System vorgegeben.
hostname Name des Remotesystem. **telnet** kann mit IP-Adressen arbeiten.

telnet> Prompt von telnet

CTRL] ^] Escape zu lokaler Shell für ein Kommando
 open *hostname* Hostname symbolisch oder als TCP/IP-Adresse.
 close Schließt Verbindung
 quit Beendet telnet
 ? gibt Hilfe zu telnet aus

ftp [-vni] [*hostname*] Startet File-Transfer-Protokoll
 -v verbose
 -n verhindert autologin, wenn ~/.netrc vorhanden ist
 -i schaltet prompt bei multiple file transfer aus (prompt)
 fpt> Prompt von ftp, erwartet Kommandos, arbeitet mit curr. directory
 open *hostname* Hostname symbolisch oder als TCP/IP-Adresse.
 close Schließt Verbindung
 user *username* Loginname auf remotesystem
 get *rem_fl* holt Datei von Remotesystem in current directory
 mget *rem_fl* Mehrere Dateien holen. Filename kann Metazeichen enthalten
 runique hängt .n an File, wenn File bereits auf lokalem System vorhanden
 put *loc_fl* Datei zum Remotesystem übertragen
 mput *loc_fl* Mehrere Dateien übertragen. Meta zulässig
 sunique hängt .n an Remote-Fln, wenn File auf Remotesystem vorhanden
 type *mode* Bestimmt Übertragungsmode (*ascii*, *binary*, *ebcdic* (DEF. *ascii*))
 delete *rem_fl* löscht Remotefile
 rename *rem_from* *rem_to* Umbenennung Remotefile
 verbose toggle verbose
 system zeigt Typ des Remote Operatingsystems
 status zeigt Status der Verbindung
 ? [cmd] Hilfe zu ftp
 pwd, cd, ls, mkdir, arbeiten auf dem Remotesystem in der gewohnten Form
 rmdir, chmod dt
 lcd *dir* cd auf lokalem System (internes ftp Kommando)
 ! startet Subshell auf lokalem System.
 ! cmd [*arg*] führt cmd auf lokalem System in einer Subshell durch
 bye beendet ftp

anonymous ftp bei einem anonymen ftp wird als Benutzername **anonymous** angegeben und als Kennwort **user@site** z.B. hahn@fra80 Auf dem Remotesystem muß der SU ein entsprechendes Account eingerichtet haben.

/etc/hosts Feste IP-Zuordnung ohne Nameserver
 /etc/hosts.equiv Zentrale Datei für autologin (rlogin, rsh oder rcp ohne Kennwortabfrage)
 ~/.rhost ermöglicht autologin (rlogin)
 ~/.netrc ermöglicht autologin (telnet) (mode 600 da Kennworte im Klartext)
 /etc/hosts.allow ACL-Files für den Super-Daemon inetd. In Verbindung mit /etc/inetd.conf
 /etc/hosts.deny kann der Zugriff von fremden Systemen besser gesteuert werden.

```

/etc/hosts
# Feld 1 Klasse Netadr. Hostadr
# 0-126 A 1 3
# 127 immer loopback localhost
#128-191 B 2 2
#192-255 C 3 1
#192.168.0 kenzeichnet Netzwerkteil
# hier sind max 254 Hosts möglich
# die Hostadr. 0 und 255 sind reserviert
#IP-Adr [Domain] Alias
127.0.0.1 localhost
192.168.0.1 detlef.bsb.de detlef
192.168.0.2 hedi.bsb.de hedi
192.168.0.3 carola.bsb.de carola

192.168.0.x sind Adressen die nicht ins
10.x.x.x Internet geroutet werden
Sie können zum Aufbau von privaten
Netzen benutzt werden
  
```

~/.rhosts regelt Zugriff auf Accountebene.

```

~/.rhosts
#hostame account
carola hahn unix0
  
```

```

/etc/hosts.equiv
# Zentrale Datei für autologin
detlef
hedi
  
```

ping [-c *n*] *host* prüft Verbindung zu host

```

~/.netrc      Verwendung in telnet
machine      rs6000
login        hahn
password     xxxxx

machine      134.98.192.11
login        anonymous password user@site

default
login        anonymous password user@site
  
```

User deren Rechner in /etc/hosts.equiv aufgeführt sind, können ohne Kennworteingabe einen Login durchführen, wenn ein Benutzer mit gleichem Namen in der /etc/passwd existiert. Gilt nicht für SU und wenn die Option -l bei rlogin oder rsh verwendet wird + in /etc/hosts.equiv ermöglicht jedem Rechner den Zugriff (Einladung für Hacker)

Ist flexibler da Benutzer auf Rechnern unterschiedliche Namen haben können. Nachteil: Kann kaum durch SU kontrolliert werden, da in Verantwortung des Users.

Reservierte Worte

{ } case do done elif else esac fi for function
if in select then time until while [[]]

Durch ksh gesetzte Variablen

_ ERRNO LINENO OLDPWD OPTARG OPTIND
_ PPID PWD RANDOM REPLY SECONDS

Vom Benutzer gesetzte Variablen

CDPATH COLUMNS EDITOR ENV FCEDIT FPATH
HISTFILE HISTSIZE HOME IFS MAIL MAILCHECK
MAILPATH PATH PS1 PS2 PS3 PS4 SHELL TERM
TMOUT VISUAL

name=value Variable anlegen und Wert zuweisen

typeset [+HLRZilrx /n] [name[=value]] dto

typeset -option	name	Attribut	Wirkung
-u			Uppercase
-l			Lowercase
-i	Integer n		gibt hierbei die Basis an -i8
-L			Linksbündig, n gibt die Feldweite an
-LZ			Linksbündig, n gibt die Feldweite an, führende Nullen werden unterdrückt
-R			Rechtsbündig, n gibt die Feldweite an
-RZ			Rechtsbündig, n gibt die Feldweite an, es wird mit führenden Nullen aufgefüllt
-Z			Äquivalent zu -RZ
-r			Readonly
-x		Export analog	export name
-t			User defined

typeset +option	name	Attribut	aufzuheben
typeset -option	alle Variablen und Values		
			mit dem angegebenen Attribut ausgeben
typeset +option	nur die Namen der Variablen		
			mit dem angegebenen Attribut ausgeben
typeset	1f[txu]	name [name] ...	
f			Attribute für function
t	xtrace		
u			function ist noch nicht definiert
x			bleibt erhalten wenn script nicht ksh separat aufruft

autoload name ... gibt namen der externen Funktionen an. FPATH muß Pfad auf Verz. enthalten
Es ist damit möglich Scripte zentral abzulegen (ähnlich wie include bei php)
autoload ist in der bash leider nicht vorhanden

alias [-tx] [kürzel[=kommandotext]]

-t tracked alias

-x export

unalias kürzel Alias aufheben

ksh -optionletter args beim Aufruf der Shell angegeben

-c string die ksh entnimmt die Kommandos dem string

-i die ksh erzeugt eine interaktive Subshell. In der Subshell

1 wird das Signal TERM ignoriert

1 wird das Signal INT abgefangen, wird das laufende Kommando abgebrochen und ein neuer Prompt ausgegeben

-r die ksh läuft als restricted Shell.

1 cd ist nicht möglich

1 Änderung von PATH, ENV oder SHELL ist unzulässig

1 es können keinen Kommandos ausgeführt werden, die einen / in ihrem Pfad haben (d.h. nur Kommandos im current Directory)

1 Die Umleitung von Ausgaben durch > >| <> oder >> ist unzulässig

-s die ksh liest Kommandos von Standard-Input. Diese Option wird automatisch gesetzt, wenn nicht -c angegeben ist. Ist -c angegeben, wird -s ignoriert

set [+aefhkmnopstuvx-] [+option] [+Aname] [arg]

Optionname letter

allexport a	bgnice	emacs	errexist e	gmacs
ignoreeof	keyword k	markdirs	monitor m	noclobber
noexec n	noglob f	nolog	nounset u	privileged p
	s	t	trackall h	verbose v
vi	xtrace x			

set -o option Optionen setzen

set -optionletter Optionen setzen

set +o option Option löschen

set +optionletter Option löschen

set arg ... set x y z setzt \$1=x \$2=y \$3=z

set -A tabname arg ... set -A tab b c a d (-A löscht die alte Tabelle)

(wird vor der Zuweisung sortiert)

set +A tab x y (tab[2] und tab[3] behalten den alten

Inhalt)

unset [-f] name ...

-f funktionname

export [name[=value]]

readonly [name[=value]]

{ kommandofolge ; } [E/A Umleitung] Die geschweiften Klammern sind eine Blockbildung.

(kommandofolge) [E/A-Umleitung] Die Kommandofolge wird als selbständiger Prozeß bearbeitet

time pipeline eval arg ... Die Parameter arg werden expandiert. Jedes Argument wird

durch ein Blank vom nächsten getrennt. Das Ergebnis wird von der ksh als Kommando

verwendet und ausgeführt. Hierbei werden die Argumente noch einmal expandiert.

Arithmetische Ausdrücke

1	(expression)	# Klammer legt die Reihenfolge fest
1	-expression	# Vorzeichen Minus
1	!expression	# Logische Negation
	~expression	# Bitweise Negation
1	expression * expression	# Multiplikation
	expression / expression	# Division
	expression % expression	# Divisionsrest exp1 durch exp2
1	expression + expression	# Addition
	expression - expression	# Subtraktion
1	expression << expression	# Linksshift exp1 um exp2 Bits nach links
	expression >> expression	# Rechtsshift exp1 um exp2 Bits nach rechts
1	expression <= expression	# Vergleich exp1 kleiner oder gleich exp2
	expression >= expression	# Vergleich exp1 größer oder gleich exp2
	expression < expression	# Vergleich exp1 kleiner als exp2
	expression > expression	# Vergleich exp1 größer als exp2
1	expression == expression	# Vergleich exp1 kleiner oder gleich exp2
	expression != expression	# Vergleich exp1 ungleich exp2
1	expression & expression	# Bitweise Verknüpfung exp1 AND exp2
1	expression ^ expression	# Bitweise Verknüpfung exp1 exlc OR exp2
1	expression expression	# Bitweise Verknüpfung exp1 OR exp2
1	expression && expression	# Logical AND wenn der erste expr false ist, wird der zweite expr nicht mehr evaluiert
1	expression expression	# Logical OR wenn der erste expr true ist, wird der zweite expr nicht mehr evaluiert
1	expression = expression	# Zuweisung
1	expression op = expression	# compound Zuweisung, op ist einer der Operatoren * / % + - << >> & ^

```
let expression let a=5 b=7+a
                let x=(a+b)/2
(( expression )) ((y=(a+9)/2))
`expr arg ` y=`expr \( $a + 9 \) / 2 `
```

```
test ausdruck (alte Form)
[ ausdruck ] (neue Form System V)
[[ ausdruck ]] (neue Form ksh)
```

```
if kommandofolge_1
then kommandofolge_2
[ elif kommandofolge_3 ]
[ else kommandofolge_4 ]
fi
```

```
kommandoliste_1 && kommandoliste_2 k-liste-2 nur ausführen, wenn k-liste-1 TRUE
kommandoliste_1 || kommandoliste_2 k-liste-2 nur ausführen, wenn k-liste-1 FALSE
```

Datei vorhanden und:

-r file	Leserechte gegeben	-k file	sticky Bit gesetzt
-w file	Schreibrechte gegeben	-s file	Länge grösser als 0
-x file	execute Bit gesetzt.	-L file	ist ein <i>symbolic-link</i>
-f file	regular file	-O file	Owner ist die <i>effective user id</i>
-d file	directory	-G file	Owner ist die <i>effective group id</i>
-c file	character special file	-S file	special file vom Type <i>socket</i>
-b file	block special file	-t [filedes]	Datei offen und ein Terminal (Default ist Descriptor 1).
-p file	named pipe (fifo)	filedes	darf nicht weggelassen werden, wenn [[...]] verwendet wird
-u file	set-user-id Bit gesetzt		
-g file	set-group-id Bit gesetzt		

binary File Operatoren

```
file1 -nt file2 file ist neuer als file2
file1 -ot file2 file1 ist älter als file2
file1 -ef file2 file1 ist ein anderer Name für file2 (link)
[[ quota -ef /tmp/try ]] && ( print link ; ls -li quota /tmp/try )
```

unary Operatoren

```
-z s1 hat der String s1 die Länge 0 hat (Leerstring)
-n s1 ist die Länge des Strings grösser 0 .
-o option ist option gesetzt
```

Numerische Operatoren

```
n1 -eq n2 Für den Vergleich von numerischen
n1 -ge n2 Werten sollten immer die entsprechenden
n1 -gt n2 Options verwendet werden und nicht
n1 -le n2 < <= = != >= > da führende Nullen
n1 -lt n2 das Ergebnis verfälschen können
n1 -ne n2
```

Parameter

Automatisch gesetzte Parameter

\$#	Anzahl der positional Parameter
\$-	Flags die an die Shell übergeben wurden oder mit set gesetzt wurden
\$?	Dezimaler Returnwert des letzten Kommandos
\$\$	PID der Shell
\$_	temporäre Variable. Das letzte Argument des vorherigen Kommandos (wird nicht für asynchrone Kommandos gesetzt)
#!	Prozeßnummer des letzten Backgroundkommandos
\$	Alle positional Parameter ab \$1 Z.B. übernommene Parameter an ein anderes script weitergeben. nextscript "\$" # in nextscript kommt nur ein Parameter an!
@\$	Alle positional Parameter ab \$1 jeweils getrennt durch einen Fieldseparator nextscript "\$@" # in nextscript kommen \$# Parameter an!

\${parameter}	\${0}	Scriptbezeichnung
	\${1}	Erster Übergabeparameter
	\${feld}	

 \$#parameter	 \$#*	Anzahl der übergebenen Parameter
	 \$#@	dto
	 \$#feld	Länge der Valueangabe des Parameters Feld
 \$#parameter[*]	 \$#array[*]	Anzahl der Elemente der Tabelle array
	 set -A array wert [...]	#erzeugt Tabelle array

Achtung Der **:** ist in den folgenden Formen optional
(in LINUX-ksh sowie bash führte **:** zu Fehler)

 \${parameter:-word}	Defaultvalue benutzen	 vi \${1:-defdatei}
 \${parameter:=word}	Defaultvalue zuweisen	 print \${datum:-\$(date %d.%m.%y)}
 \${parameter:?word}	Wenn NULL oder unset Fehlermeldung ausgeben	 print \${1:? "Parameter 1 fehlt"}
 \${parameter:+word}	Nur substituieren wenn par. vorhanden (alternate value)	 dir=\${3:+\$HOME/flib} # wenn \$3 vorhanden wird dir gesetzt
		 exec wish \$0 \${1:+\$*} #wenn \$1 vorhanden alle übergeben
 \${parameter:#pattern}	remove small left pattern	 PS1=\${PWD:#\$HOME/}
 \${parameter:##pattern}	remove large left pattern	 echo \${PWD:##*/}
 \$parameter:%pattern}	remove small right pattern	 echo \${HOME:}/\${USER}
 \$parameter:%%pattern}	remove large right pattern	 x=home/hahn/unix/ksh
		 echo \${x:%%/*} -> home

Iterative Kommandos

```

case wort in
  muster_1) kommandofolge_1::
  muster_2) kommandofolge_2::
  ...
  *) defaultkommandos::
esac

```

```

for name [in wort_1 wort_2 ...]
do
  kommandofolge
done

```

```

while kommandoliste_1
do
  kommandoliste_2
done

```

```

until kommandoliste_1
do
  kommandoliste_2
done

```

break verlässt Iteration
break [*n*] Verlässt iterative Anweisung (**for**, **while**..)
n gibt Schachtelungstiefe an (ist nicht überall vorhanden).

continue geht zur Auswertung der Bedingung

```

# Gibt Menu aus, benutzt LINES,
# COLUMNS, PS3 und REPLY
select identifier [in word ..]
do
  kommandoliste
done

```

print [-Rnprsu [*n*]] [*arg*] ...

print verwendet folgende Escapezeichen (Ausnahme -R oder -r wurde angegeben)

\a	Bell-Character ausgeben
\b	Backspace
\c	Zeilenvorschub am Ende unterdrücken
\f	Seitenvorschub ausgeben
\n	Neue Zeile
\r	Return, (zurück an Zeilenanfang)
\t	Tabzeichen ausgeben
\v	Vertical Tab
\\	Backslash
\0x	ASCII-Zeichen ausgeben, x ist eine 1,2 oder 3-stellige Oktalzahl
print \033	gibt ESC-Zeichen aus
-	Alle Zeichen nach - werden als arg aufgefasst, auch wenn sie mit -arg beginnen

-R \ Konventionen nicht verwenden. Nach -R alles als arg verwenden (Ausnahme -n)

-n Der Zeilenvorschub am Ende der Ausgabe wird unterdrückt

-p Die Ausgabe wird zum Co-Prozess umgeleitet (siehe Co-Prozess)

-r Die \ Konventionen sollen nicht verwendet werden. **print -s PATH=\$PATH**
nach ESC auf dem Bildschirm PATH=:/bin:/usr/bin:/usr/local/bin:/usr/hahn/bin:

-u Die Ausgabe auf Descriptor n (Def. 1). n muß vorher mit **exec** geöffnet werden.

/bin/echo -n gibt die Argumente ohne Zeilenvorschub am Ende aus (systemabhängig)

/bin/echo -n "Bitte Wert eingeben"

read wert

read [-prsu [*n*]] [*name*?prompt] [*name*] ...

name erste Teilfeld -> ersten *name_1*, 2.Teilfeld -> *name_2*, Rest -> *name_n*

-p liest vom Co-Prozess. Bei EOF wird der Co-Prozess disconnected.

-r Ein \ in der Zeile signalisiert keine Zeilenverlängerung

-s Eine Kopie der eingelesenen Zeile wird in der Historydatei abgelegt

-un von Descriptor *n* gelesen (Default 1) . 0 <= n <= 9 e.

Ist n > 2 so muß der Descriptor durch **exec** geöffnet werden

exec [command] [arg] ersetzt die aktuelle Shell ohne einen neuen Prozess zu erzeugen.

exec ls -l führt ls durch. Der Anwender ist danach ausgelogged,

Datei öffnen

exec 3< readfile öffnet *readfile* als Einheit 3 zum Lesen

exec 3> writefile öffnet *writefile* als Einheit 3 zum Schreiben

Eine existierende Datei wird auf Länge 0 gebracht,dann wird beim Schreiben angehängt.

exec 3<&- Close von Einheit 3

exec 5<&0 Erzeugt eine Kopie des Descriptors 0 als Descriptor 5

exec 3<> /dev/tty4a öffnen Terminal tty4a read/write

trap [action] [condition]

action definiert die Trap-Kommandos, müssen in ' Single Quote ' eingeschlossen werden

'cmd' cmd wird dieses beim Eintreten des Ereignisses durchgeführt

NULL die ksh ignoriert das Ereignis

- die condition auf ihren ursprünglichen Zustand zurückgesetzt

condition Ereignis bei dem die Trap ausgelöst wird.1(Name oder Nummer eines Signals)

trap 'print Abbruch' INT oder **trap 'print Abbruch' 2**

kill -l gibt eine Übersicht der Signale mit ihren Namen oder man signal

Sinnvolle Signale in der Shell sind: 1 HUP, 2 INT, 3 QUIT, 15 TERM

Nicht abgefangen werden können 9 KILL, 24 STOP

ERR wird ausgelöst, wenn ein Kommando ein Returnvalue ungleich 0 (FALSE) zurückgibt.
wird nicht von Funktionen geerbt.

EXIT oder 0 wird beim Verlassen der Funktion oder der ksh ausgelöst

trap '\$HOME/logout' EXIT # z.B. in .profile

DEBUG wird nach der Ausführung eines jeden Kommandos durchgeführt.

wird nicht von Funktionen geerbt. **trap 'print \$0 [\$LINENO]' DEBUG**

Reihenfolge bei gleichzeitigen Traps DEBUG, ERR, alle anderen Traps, - EXIT

Wenn eine separate ksh aufgerufen wird, so werden Traps die in der Parentshell ignoriert werden nicht an die Subshell ausgeliefert. Auch dann nicht, wenn sie in der Subshell explizit definiert werden.

Systemdateien:

/etc/group # Gruppenzuordnung der Benutzer
/etc/passwd # Benutzerkenndaten
/etc/default/passwd # Defaultwerte für Password-Aging (einige SYS V)
/etc/shadow # Kennworte (AIX /etc/security/passwd)
/etc/profile # Profile für alle Benutzer
/etc/gettydefs # Beschreibung aller Leitungsparameter
/etc/inittab # Prozesse für boot und getty nach Änderung
für getty Aktivierung durch : init q
/etc/issue # Login-Bildschirm
/etc/motd # Tagesnachrichten (Message of Today)
/usr/lib/keyboard/german.map # Umlaute für Keyboard
/usr/options/.license # Anzahl der lizenzierten User
/usr/vpix/etc/.license # Anzahl lizenzierte vpix User
/usr/spool/lp/logs/requests # Protokollierung aller Druckjobs,
/usr/spool/lp/logs/lpsched # Protokoll Start/Stop Printservice
request und lpsched muß sporadisch vom su gelöscht werden
/etc/wtmp # enthält accountig Informationen
muss sporadisch vom SU gelöscht werden.
mit touch wieder anlegen chown und chgrp auf adm

Drucker Systemdateien

/usr/spool/lp/admins/lp/filter.table # aktuelle Filter
/usr/spool/lp/admins/lp/filter.table.i # Initial Filter
/usr/spool/lp/users # Maximale Prios für Benutzer #Umschaltung zwischen

/etc/hosts # Symbolische und IP-Adressen werden hier verbunden
/etc/hosts.lpd # ACL für Remotezugriff auf Drucker
~/.rhosts # ACL für Login ohne Kennwort (rlogin) von Fremdrechner
~/.netrc # Maschinennamen und Kennworte für Zugriff auf Fremdrechner

Virtuellen Terminals der Console ALT+DRUCK dann F1..F8

#F1 = VT1, F8 = CONSOLE

shutdown -y -g0 -i0 # y=preanswer confirm. quest, g0=0Sek.,i0=für Power off

AIX security

/etc/security

/ect/security/login.cfg

/etc/security/mkuser.default

/etc/security/mkuser.sys

/etc/security/user

/etc/security/limits

/etc/security/environ

/etc/security/passwd

Verzeichnis mit sicherheitsrelevanten Daten

Konfigurationsdatei für Login. Enthält auch Shells für **chsh**

wird von smit und mkuser verwendet

Script, wird von mkuser aufgerufen (Änd. durch su mögl.)

erw. 4 Param. Homeverz., Username, Gruppe, Shell

Benutzerspezifische Attribute und Standardeinstellungen

Limits der Anwender bezgl. Systemressourcen

Spezifische Umgebungsvariablen

Crypted Passwords

Drucken INTERACTIVE

Drucker Systemdateien

/usr/spool/lp/admins/lp/filter.table # aktuelle Filter
/usr/spool/lp/admins/lp/filter.table.i # Initial Filter
/usr/spool/lp/users # Maximale Prios für Benutzer #Umschaltung zwischen

Drucker Kommandos

/usr/lib/lpshut # Stop Printerservice
/usr/lib/lpsched # Start Printerservice
/usr/lib/lpmove request dest # Druckauftrag zu anderem Drucker umsetzen
/usr/lib/lpmove dest1 dest2 # Alle Aufträge umsetzen dest1 nimmt keine weiteren Aufträge mehr an

/usr/lib/lpusers -d prio # systemweite Default-Prio für Druckauftrag setzen Prio 0 high 39 low (Default 20)

/usr/lib/lpusers -q prio # Maximal anforderbare Prio für alle Benutzer die nicht explizit aufgeführt sind

/usr/lib/lpusers -q prio -u user-list # Max anfordb. Prio für angef. User

/usr/lib/lpusers -u user-list # angef. User wieder auf Def.-Prio setzen

lp [options] file [file ...]

-c # Datei kopieren nicht aus Originaldatei drucken
-n number # Anzahl der Drucke
-m # Mail schicken wenn Druckvorgang beendet ist
-w # Message zum Terminal wenn Druck beendet
-s # Message unterdrücken z.B "request-id is ..."
-q prio # Druckauftrag mit angegebener Prio durchführen (0=high)
-t title # Titel für Banner, wird als \$3 ins Interface übergeben
-f form-name [-d any] # Verwende für den Ausdruck das Form form.name
-d any jeder Drucker, der form-name unterstützt, kann verwendet werden.
-H spec-hand # Special Handling
hold Druck zurückstellen, bis Freigabe durch resume
resume Freigabe z.B lp -i laser-285 -H resume
immediate sofortiger Druck (su only), ein bereits druckender Job muss vorher mit hold zurück gestellt werden.
-o depend-opt # Printer- oder Class-abhängige Option müssen im Interface behandelt werden (werden nur durchgereicht)
(sdn sind scaled-decimal-number d.h durch einen Postfix
i Inch, c Centimeters, zB 27.94c (Länge DIN A4) 11i (11 Inch)
alle sdn können nicht in Verbindung mit -f gegeben werden)
nobanner Banner-Page unterdrücken (kann vom su unterbunden werden)
nofilebreak Druckdateien nicht durch Seitenvorschub trennen
length=sdn Seitenlänge in Zeilen, cm oder inch

width=sdn Seitenbreite in Spalten, cm oder inch

lpi=sdn Lines per inch

cpi=sdn Character per inch (hier kann auch pica =10 cpi oder elite =12 cpi oder compressed max cpi angegeben werden)

stty=stty-option-list

-P page-list # Angabe der zu druckenden Pages (erfordert entspr. Filter)

-T content-type [-r] # Ausgabe auf Drucker der entsprechenden Content-Type unterstützt. Ist kein entsprechender Drucker vorhanden, wird ein Filter gesucht. -r unterdrückt Filtersuche

-y mode-list # Mode-list wird umgesetzt und von einem Filter bearbeitet

lpstat options

-a [list] # acceptance status, [list] Printer oder Classnames (Def all)

-c [list] # Classnames und Members anzeigen (Def. all)

-d # Default Destination anzeigen

-f [list] [-l] # verifiziere das Forms in list akzeptiert werden

-o [list] [-l] # Status der Outputrequests (Def. all), -l =Long Form

-p [list] [-D] # Status der in list angef. Drucker zeigen, -D =Kurze Form

[-l] # Long Form incl. mounted Forms,Content,Printertype,Interface

-r # Status des LP-Schedulers

-s # Status Summary

-S [list] [-l] # verif.das Character Sets und Print-wheels akzeptiert werden

-t # Alle Statusinformationen

-u [list] # Status Outputreq. für user in list (def. all)

-v [list] # Namen der Printer, Pfade der zugehörigen Devices (Def. all)

/usr/lib/lpfilter -f filtername -l # Alle Filter anzeigen

/usr/lib/lpfilter -f filtername -x # benannten filter löschen

/usr/lib/lpfilter -f filtername -i # benannten filter Initialwerte holen

/usr/lib/lpfilter -f filtername -F path # Filter installieren

#wird anstelle von filtername **all** eingesetzt, bezieht sich die Operation auf alle Filter z.B

/usr/lib/lpfilter -f all -l

Drucken unter BSD

Folgende Programme wickeln unter BSD den Printverkehr ab.
Alle haben das SUID und SGID-Bit gesetzt.

/etc/lpd	daemon	wird beim Booten gestartet
/usr/bin/lpr	print	stellt Auftrag in Warteschlange
/usr/bin/lpq	query	zeigt Inhalt der Spooldirectory
/usr/bin/lprm	remove	löscht Auftrag aus Warteschlange
/usr/bin/lpc	control (su)	enable/disable Queue
/usr/bin/lpstat	status	Abfrage Druckerstatus, ursprünglich System V aber auch bei DIGITAL UNIX vorhanden
Referenzdateien		
/etc/printcap	capabilities	Beschreibung Drucker wird vom su verwaltet. für Remotedrucker
/etc/hosts.lpd		

lpr [option] *fln*

Options

- h** Burst-Page unterdrücken
- C *str*** Die Zeichenkette *str* wird auf der Burst-Page an Stelle des Hostnamens ausgegeben
- r** Datei nach Ausdruck löschen (**remove**)
- m** Benutzer bekommt eine **Mail** wenn der Ausdruck beendet wurde
- #*n*** Anzahl der Kopien
- l** Control-Character sollen ausgegeben werden (Seitenvorschübe werden unterdrückt)
- f** das erste Zeichen jeder Zeile wird als Vorschubsteuerzeichen verwendet.
- Px** Angabe der Printerbezeichnung *x*. Fehlt die Option wird der Defaultprinter verwendet oder der Printer, der in der Environmentvariablen **PRINTER** angegeben ist oder **LPDEST**

lpq [-l] [**Pprinter**] [*jobnr*] [*user...*]

- l** Anzeige erfolgt im Long-Format
- Pprinter** Angabe des Printers über seinen Namen.

lprm [-Pprinter] [-] [*jobnr*] [*user*]

- Pprinter** Angabe des Printers über seinen Namen oder die zugeordnete Nummer.
Wurde keine Angabe gemacht, so wird auf den Defaultprinter zugegriffen oder auf den Printer der in der Environmentvariable **PRINTER** definiert ist
- Alle Druckaufträge des Benutzers löschen

lpc [command [argument ...]]

Werden keine commands angegeben, so arbeitet das Programm interaktiv.

- ? [command ...] beide Formen geben eine kurze Übersicht über die angegebenen
- help [command ...] Kommandos
- status { all | *printer* }
- abort { all | *printer* } Terminiert einen aktiven Spooldaemon. Es können keine neuen

clean { all | *printer* }
disable {all | *printer*}

enable {all | *printer*}

down { all | *printer* }
message ...

up { all | *printer* }
start { all | *printer* }
stop { all | *printer* }

restart { all | printer }
topq printer [jobnum ...] [user ...]

exit oder quit

lpstat -a [*printer*]
-o [*printer*] zeigt Printerrequests
-p [*printer*] Status der Warteschlange -p"lp0, lp1, lp2"
-v [*printer*] Liste der Pfadnamen
-u [users] Printstatus für Jobs der angegebenen User
-r Status Printdaemon
-s Status Summary
-t total (alle Statusinformationen)

lpd Line Printer Daemon (8) . Verwendet Systemcalls listen(2) und accept(2) um Printaufträge zu bekommen. Setzt per fork() Child-Prozesse auf, die den Printauftrag behandeln. Verwendet /etc/printcap /etc/hosts.equiv /etc/hosts.lpd

Daemons für den angegebenen Printer aufgesetzt werden.
Alle temporären Daten- und Kontrolldateien werden gelöscht
Die angegebenen Queues werden gestoppt. Es können keine neuen Jobs mit lpr in die Queues gestellt werden.
Die angegebenen Queues werden aktiviert. Es können neuen Jobs mit lpr in die Queues gestellt werden.
Die angegebenen Printer werden gestoppt. Es können keine neuen Jobs mit lpr in die Queues gestellt werden.
lpq zeigt die Message, damit der Benutzer weiß, warum der Printer down ist.
Die angegebenen Printer werden gestartet
Erlaubt Ausgabe auf Printer und startet einen neuen Spooldaemon.
Stoppt den Spooldaemon nach Abschluß der augenblicklichen Ausgabe. Verhindert die Übergabe neuer Jobs an den Printer
Versucht einen neuen Spooldaemon zu starten.
Mit diesem Kommando kann die Reihenfolge der Ausgabe beeinflußt werden.
Beide Kommandos verlassen den interaktiven lpc.

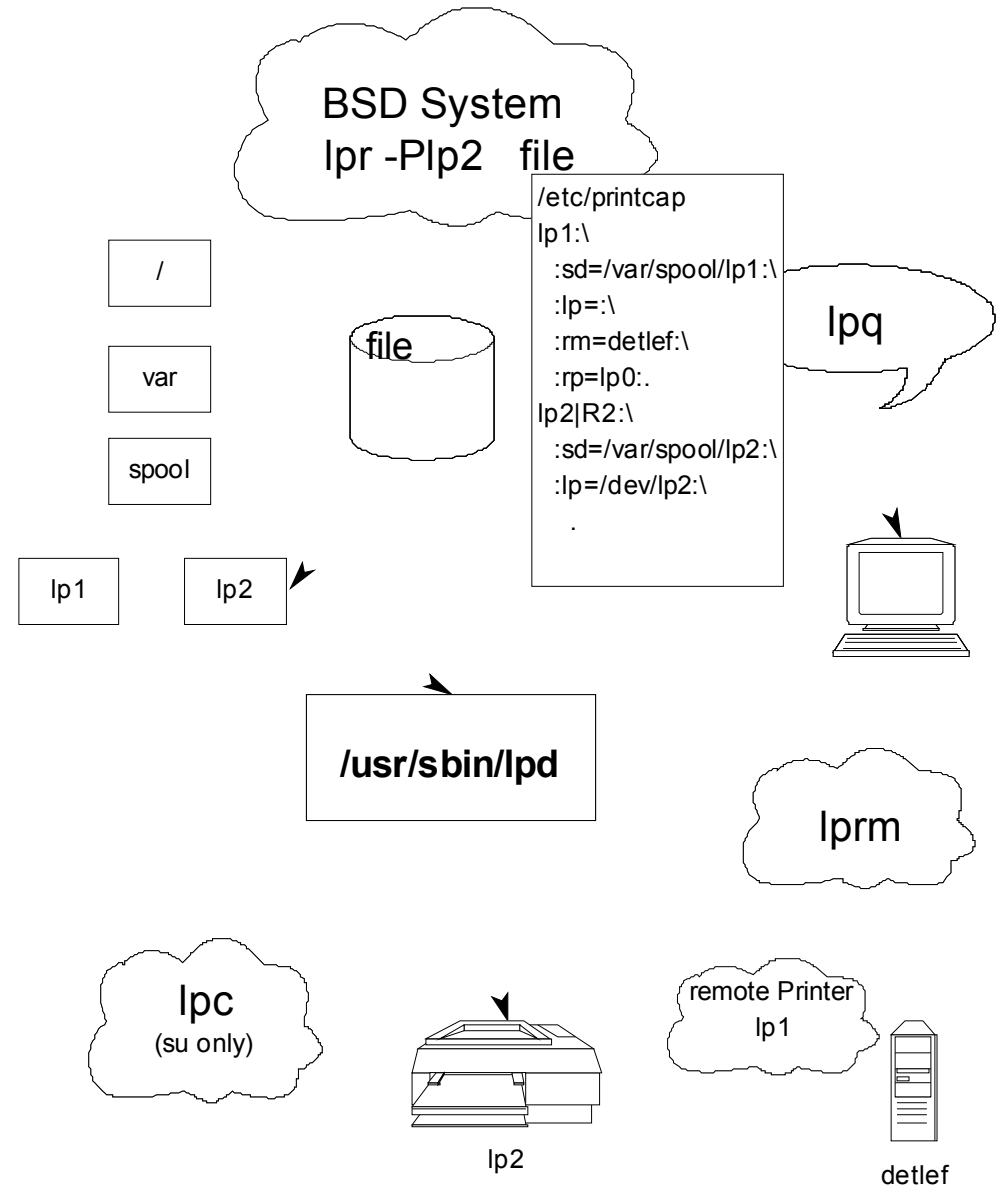
Capability

Die Capabilities können als String, Bool- oder numerischer Wert auftreten.
Sie können Defaultwerte haben (in eigener Systemdoku nachlesen).

Die wichtigsten Capabilities sind:

af	Name der Accountingdatei
if	Name des Textfilters der Accounting durchführt
br	Übertragungsgeschwindigkeit (Baud Rate)
ct	Verbindungsart (connect type) z.B. lat
lp	sd gibt das Spoolverzeichnis an. Das Verzeichnis muß existieren Deviceangabe der Ausgabedatei
lf	Error Logfile
if	Textfilter der Accounting durchführt
of	Output Filter
xf	Pass-through-filter
rm	Maschinenname für Remoteprinter
rp	Argument für Remoteprinter (Def. lp)
rs	Restrict Remote User. Nur User die auch ein lokales Account haben dürfen drucken.
pc	Preis pro Fuß oder Seite in 0,01 Cents
pl	Seitenlänge in Zeilen (Def. 66)
pw	Zeilenlänge in Zeichen (Def 132)
mx	Maximale Dateigröße (in BUFFSIZ Blöcken), 0 entspricht unlimited.
rw	Printerdevice read/write öffnen
sh	Überdrücke Burstheaderpage
fc	Wenn lp Device tty clear Flagbits
fs	Wenn lp Device tty setze Flagbits
xc	Clear Local-Mode-Bits
xs	Set Local-Mode-Bits
Op	bei LAT Port Ts bei LAT Server

Weitere Capabilities man 4 printcap (evtl auch man 5 printcap)



copy [**opt**] *source ... dest* # **(INTERACTIVE)** Kopiert den Inhalt von Directories in
anderes Directory (falls erforderlich werden Subdirectories angelegt).
Wenn *source* und *dest* einfache Dateien sind arbeitet es wie cp

- a fragt nach ob copy durchgeführt werden soll (ASK) setzt -ad Option
- l verwendet wo möglich Links (nie für Directories und Special Files)
- n *dest* muß neu sein (gilt nicht für Directories, für Special Files wird immer -n
angenommen)
- o Owner und Group bleibt erhalten (Default Owner und Group wird auf den Wert
gesetzt den der aufrufende Process hat)
- m Modifikation und Access-Time bleibt in der Kopie erhalten
- r Subdirectories werden rekursiv abgearbeitet (Default Subdirectories werden
ignoriert)
- ad Fragt Benutzer ob Subdirectory abgearbeitet werden soll
- v verbose `copy -rv /usr/hahn/ksh /usr2/ksh`

Wenn -n oder user@host verwendet wird, muß der **writesrv** daemon auf beiden Systemen gestartet sein und das Verz. **/usr/spool/writesrv** existieren (DIGITAL nicht gestartet)

mkdir /usr/spool/writesrv

rcmgr set WRITESRV yes # yes|no start at boot time (ändert etc/rc.config)
/sbin/init.d/write start # start|stop start manuel

write [-r] [-n host] user [terminal] Schreibt Nachricht an user, EOF durch ^D.

Terminal muß offen sein (**mesg y**).

-r wartet auf Antwort. Erzeugt *handle* für **write -h handle,reply**

-n host gibt Remote-Host an. Kann Alias oder IP-Adresse sein

user Benutzername

terminal Wenn User mehrere virtuelle Bildschirme hat, dann terminal verwenden.

Mit **finger** aktives Terminal ermitteln

write -h handle,reply Antwort auf **write -r**
reply kann sein **ok, cancel** oder **query**

write user@host alternative Form für write -h

Beispiele :

```
1.)
write hahn tty1
Hallo Detlef, was ist mit einer Pause ?
^D
```

```
2.) Holding Conversation (fast wie talk)
write user1
Ich möchte was besprechen
over
.
write user2
Ok passt 13:00 ?
over
```

```
write user 1
OK
bye
^D
```

```
3)
write - console
Was ist mit der Druckerqueue R8
^D
```

```
4.)
write root@fra80 console
Was ist mit dem Printer in Raum 8 ?
Die Queue nimmt keine Aufträge an
^D
```