

SIEMENS
NIXDORF

SINIX

SINIX V5.41 (MX300/MX500)

Ergänzungen

Sie haben

uns zu diesem Handbuch etwas mitzuteilen?
Schicken Sie uns bitte Ihre Anregungen unter
Angabe der Bestellnummer dieses Handbuches.

Siemens Nixdorf Informationssysteme AG
Manualredaktion STM QM 2
Otto-Hahn-Ring 6
W-8000 München 83

Fax: (0 89) 6 36-4 04 43

email im EUnet:
man@sieqm2.uucp

SINIX V5.41 (MX300/MX500)

Ergänzungen

Einleitung

Systemadministration

Netzwerke

COLLAGE

C-Programmierung

Manual Pages

Reguläre Ausdrücke

Verzeichnisse

Wollen Sie mehr Wissen . . .

. . . zu diesem Produkt
. . . oder zu einem anderen Thema der Informationstechnik?

Unsere Training Center stehen mit ihrem Kursangebot für Sie bereit.
Besuchen Sie uns in Berlin, Essen, Frankfurt/Main oder Hamburg,
in Hannover, Mainz, München, Stuttgart, Wien oder Zürich.

Informationen zu unserem Trainingsangebot erhalten Sie über:

München (0 89) 6 36-20 09

Oder schreiben Sie an:

Siemens Nixdorf Training Center
Postfach 83 09 51, W-8000 München 83

X/Open XPG3-konform
Warenzeichen beantragt

SINIX® Copyright © Siemens Nixdorf Informationssysteme AG, 1990.
SINIX ist das UNIX® der Siemens Nixdorf Informationssysteme AG.
UNIX ist ein eingetragenes Warenzeichen von UNIX System Laboratories, Inc.

Copyright © Siemens Nixdorf Informationssysteme AG, 1992. Alle Rechte vorbehalten.
Weitergabe sowie Vervielfältigung oder Übersetzung dieser Unterlage, Verwertung und
Mitteilung ihres Inhaltes nicht gestattet, soweit nicht ausdrücklich zugestanden.
Zuwiderhandlungen verpflichten zu Schadenersatz.
Liefermöglichkeiten und technische Änderungen vorbehalten.

Inhalt

Einleitung	1
Aufbau des Ergänzungsbandes	1
Systemadministration	7
Konsole konfigurieren (MX500)	8
SX-Board konfigurieren	9
Drucker am TACLAN verwalten	11
2,88 MB SCSI-Disketten verwalten	13
Sicherung der Benutzerkennungen	15
Nachträge	16
Nachträgliche Installation einer Festplatte	16
Einrichten eigener Gerätedateien	16
Anzahl der Pseudo-TTYs	17
Protokolldatei des fsck	17
Gerätedateien für Datenträger	17
Plattenslices	18
Änderung des Run-Levels	19
Informationen über installierte Produkte	19
Magnetband-Kassetten-Laufwerk 60/150 MByte	19
Magnetband-Kassetten-Laufwerk 2,3 Gbyte	20
1/2 Zoll Magnetbandgerät FS1000/FS2000	21
Konfigurieren von BA80-Terminals	21
Netzwerke	23
Ladbarer Ethernet-Prozessor 97882-143 (MX500)	24
Einhängen der Rechner in das Netz	25
inet - Konfigurationsdatei	26
Asynchrones Schreiben über NFS	28
Netzüberwachung mit SNMP-Agenten	29
Funktion des SNMP-Agenten	29
SNMP-Agenten und Netzmanagementstationen	29
Domänen und Community-Namen	30
Operationen des SNMP-Agenten unter SINIX V5.41	31
Installieren des SNMP-Agenten	31
Installieren der SNMP-Software	32
Erweitern der Verwaltungsdatei services	32

Start/Stop des SNMP-Agenten	33
Rekonfiguration des SNMP-Agenten	34
Festlegen der Community-Namen	34
Änderung der Einstellungen mit dem Kommando snmpadm	35
COLLAGE	37
Das Menü 'Druckaufträge'	38
Neuen Editor eintragen	39
Hinweise zur Bedienung von Alpha-COLLAGE	40
In einem Shell-Fenster arbeiten	40
Sondertaste 'Ausgabe unterdrücken - ganz'	40
C-Programmierung	41
ABI386-Konformität	42
Optionen zur Optimierung und Codegenerierung	43
CES-Abweichungen vom ANSI-Standard	45
Präprozessor-Anweisung #pragma weak	45
Installation von Fortran77-Komponenten	46
Post&Wait-Treiber (MX500)	47
Post&Wait - Grundoperation	47
Implementierung von Post&Wait-Operationen mit einem Pseudogerät	48
Zusätzliche Funktion des Post&Wait-Pseudotreibers	50
Asynchronous Raw I/O (MX500)	52
Introduction	52
The ioctl Commands	53
Performance Results	55
Manual Pages	61
acctcom - Suchen und Drucken von Prozeßabrechnungsstatistiken	63
ar - Bibliotheken verwalten	64
awk - Programmierbare Bearbeitung von Textdateien	65
bc - Arithmetische Sprache	66
cal - Kalender ausgeben	69
copy - Dateien gruppenweise kopieren	70
cpio - Dateien und Dateiverzeichnisse ein- und auslagern	71
crypt - Text verschlüsseln und entschlüsseln	72
csh - C-Shell	73
df - Anzahl der freien und belegten Plattenblöcke und I-Nodes ausgeben	74
diff - Dateien zeilenweise vergleichen	75
diff3 - Drei Dateien zeilenweise vergleichen	75
dirname - Pfad-Präfix vom Dateinamen trennen	76
flchk - Labelbereich einer Diskette überprüfen (floppy check)	77
gettext - Zeichenketten in einer Datenbasis für Meldungstexte suchen	78
ksh - Korn-Shell	79

login - Sich neu am System anmelden	80
lp - Dateien ausdrucken	81
lpr - Dateien ausdrucken und Druckaufträge steuern	81
lpstat - Informationen über Druckaufträge ausgeben	81
ls - Informationen über Dateiverzeichnisse und Dateien ausgeben	82
mail - Nachrichten senden oder lesen	83
mailx - Nachrichten interaktiv bearbeiten	83
mt - Magnetband oder Magnetbandkassette bearbeiten	84
mv - Dateien versetzen oder umbenennen	85
newgrp - Gruppenzugehörigkeit ändern	86
passwd - Login-Kennwort und Kennwortattribute eintragen oder ändern	87
pr - Dateien formatieren und auf die Standardausgabe ausgeben	88
ps - Prozeßdaten abfragen (process status)	89
rm - Dateien löschen	90
sar - Über Systemtätigkeit berichten	91
secure - Benutzererkennung überprüfen	92
sed - Editor im Prozedurbetrieb	95
sh - Bourne-Shell	96
shl - Schichtenverwaltung für Shells	96
sort - Dateien sortieren und/oder mischen	97
tail - Den letzten Teil einer Datei ausgeben	98
talk - Dialog mit anderem Benutzer führen	99
tar - Archivieren von Dateien auf Magnetbandkassette, Band oder Diskette und Archive bearbeiten	100
time - Laufzeit eines Kommandos messen	101
timex - Laufzeit eines Kommandos messen, Prozeßdaten und Systemaktivitäten anzeigen	101
uniq - Mehrfache Zeilen suchen	102
vi - Bildschirmorientierter Editor	103
wc - Wörter, Zeichen und Zeilen zählen (word count)	105
wcheck - Veränderte Dateien suchen (write check)	106
who - Aktive Benutzerkennungen anzeigen	107
tcpdump(1)	108
addbad(1M) (MX500)	125
bind(1M) (MX500)	126
boot(1M) (MX300)	128
colltbl(1M)	135
edvtoc(1M)	136
gated(1M)	138
smtper(1M)	157
swap(1M)	158

Inhalt

traceroute(1M)	159
exa(7)	165
shd(7)	168
Reguläre Ausdrücke	173
Literatur	185
Stichwörter	191

Einleitung

In diesem Ergänzungsband sind nicht dokumentierte Neuerungen zu verschiedenen Themenbereichen des Betriebssystem SINIX V5.41 für MX300 und MX500 zusammengefaßt. Die Themenbereiche sind:

- Systemadministration,
- Netzwerke,
- COLLAGE,
- C-Programmierung,
- Kommandos und Gerätetreiber-Dateien (Manual Pages).
- Reguläre Ausdrücke

Aufbau des Ergänzungsbandes

Den oben genannten Themenbereichen sind abschnittsweise die Änderungen und Ergänzungen zu SINIX V5.41 zugeordnet. Bei den Änderungen sind die meisten Abschnittüberschriften so gewählt, daß sie den jeweiligen Abschnittüberschriften in den betroffenen Handbüchern entsprechen. Bei den Ergänzungen beinhalten die Abschnittüberschriften jeweils Hinweise auf die Kapitel, die ergänzt werden.

Änderungen oder Ergänzungen, die entweder nur MX300 oder nur MX500 betreffen, sind in den Abschnittüberschriften und ggf. in den Kopfzeilen wie folgt gekennzeichnet:

MX300: Abschnittüberschrift (MX300)

MX500: Abschnittüberschrift (MX500)

In den meisten Abschnitten wird Ihnen zunächst das Handbuch und das Kapitel genannt, das von der Änderung oder Ergänzung betroffen ist. Bei Änderungen folgt dann noch die Abschnittüberschrift.

Den Tabellen auf den nächsten Seiten können Sie entnehmen, in welchen Abschnitten dieses Ergänzungsbandes Änderungen (Ä) und in welchen Abschnitten Ergänzungen (E) zum genannten Handbuch beschrieben ist. Die vollständigen Titel der Handbücher und deren Bestellnummern finden Sie am Ende dieses Ergänzungsbandes.

Systemadministration	Status	Handbuch
Konsole konfigurieren	Ä	} Bediensystem für Systemverwalter [2]
SX-Board konfigurieren	Ä	
Drucker am TACLAN verwalten	Ä	
2,88 MB SCSI-Disketten verwalten	Ä	} Kommandos, Band 3 [14]
Sicherung der Benutzerkennungen	Ä	} Bediensystem für Systemverwalter [2]
Nachträge		
- Nachträgliche Installation einer Festplatte	E	} Leitfaden für Systemverwalter [1]
- Einrichten eigener Gerätedateien	Ä	
- Anzahl der Pseudo-TTYs	Ä	
- Protokolldatei des fsck	Ä	} Leitfaden für Systemverwalter [1] } Referenzhandbuch für Systemverw. [11]
- Gerätedateien für Datenträger	Ä	} Kommandos, Band 3 [14]
- Plattenslices	Ä	} Leitfaden für Systemverwalter [1]
- Änderung des Run-Levels	Ä	
- Informationen über installierte Produkte	Ä	} Leitfaden für Systemverwalter [1] } Referenzhandbuch für Systemverw. [11]
- Magnetband-Kassetten-Laufwerk 60/150 MByte	Ä	} Kommandos, Band 3 [14] } Leitfaden für Benutzer [1]
- Magnetband-Kassetten-Laufwerk 2,3 GByte	Ä	
- 1/2 Zoll Magnetbandgerät FS1000/FS2000	Ä	
- Konfigurieren von BA80-Terminals	Ä	} DPTG-2 V1.0 Terminal-Multiplexer-Protokoll, Installationsanleitung [16]

Tabelle 1: Änderungen (Ä), Ergänzungen (E) im Kapitel Systemadministration

Netzwerke	Status	Handbuch
Einhängen der Rechner in das Netz	Ä	Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung [4]
inet - Konfigurationsdatei	Ä	
Asynchrones Schreiben über NFS	Ä	
Netzüberwachung mit SNMP-Agenten	Ä	

Tabelle 2: Änderungen (Ä), Ergänzungen (E) im Kapitel Netzwerke

COLLAGE	Status	Handbuch
Das Menü 'Druckaufträge'	Ä	COLLAGE-Bediensystem [5]
Neuen Editor eintragen	E	
Hinweise zur Bedienung von Alpha-COLLAGE	E	COLLAGE V4.0 [6]

Tabelle 3: Änderungen (Ä), Ergänzungen (E) im Kapitel COLLAGE

C-Programmierung	Status	Handbuch
ABI386-Konformität	E	CES, Leitfaden f.d. Programmierung [7] CES, cc-Kommando für MX300 + WX200 [8]
Optionen zur Optimierung und Codegenerierung	E	
CES-Abweichungen vom ANSI-Standard	E	CES, Leitfaden f.d. Programmierung [7]
Installation von Fortran77-Komponenten	E	CES, Referenzhand.f. Programmierer [9]
Post&Wait Treiber	E	
Asynchronous Raw I/O	E	Asynchronous I/O Reference Manual [10]

Tabelle 4: Änderungen (Ä), Ergänzungen (E) im Kapitel C-Programmierung

Manual Pages	Status	Handbuch	
acctcom	Ä/E	} Kommandos, Band 1 [12]	
ar	E		
awk	Ä/E		
bc	Ä/E		
cal	E		
copy	Ä		
cpio	E		
crypt	E		
csh	Ä		
df	Ä		
diff	Ä		
diff3	E		
dirname	E		
flchk	E		
gettxt	Ä		
ksh	Ä/E		
login	E		} Kommandos, Band 2 [13]
lp	E		
lpr	E		
lpstat	E		
ls	Ä/E		
mail	Ä/E		
mailx	Ä		
mt	Ä		
mv	Ä		
newgrp	E		
passwd	E		
pr	Ä/E		
ps	E		
rm	E		
sar	E		
secure	E		
sed	Ä		
sh	Ä		
shl	Ä		
sort	E		
tail	E		
talk	Ä		
tar	E		
time	E		
timex	E		
uniq	Ä		
vi	Ä/E		
wcheck	E		
wc	E		
who	Ä/E		

Fortsetzung ...

Manual Pages	Status	Handbuch
tcpdump(1)	E	} Netzwerke – Leitfaden für Benutzer und Verwalter [3]
addbad(1M)	Ä	
bind(1M)	E	} Referenzhandbuch für Systemverw. [11]
boot(1M)	Ä	
colltbl(1M)	Ä	
edvtoc(1M)	Ä	
gated(1M)	E	
smtpqer(1M)	Ä	} Referenzhandbuch für Systemverw. [11]
swap(1M)	Ä	
traceroute(1M)	E	} Netzwerke – Leitfaden für Benutzer und Verwalter [3]
exa(7)	Ä	
shd(7)	Ä	} Referenzhandbuch für Systemverw. [11]

Tabelle 5: Änderungen (Ä), Ergänzungen (E) im Kapitel Manual Pages

Zielgruppe des Handbuchs

Da die einzelnen Abschnitte unterschiedlichen Handbüchern zugeordnet sind, entnehmen Sie bitte die jeweilige Zielgruppendefinition dem Verzeichnis der Handbücher am Ende dieses Ergänzungsbandes.

Systemadministration

In diesem Kapitel sind die Änderungen und Ergänzungen beschrieben, die zur Systemadministration gehören. Diese Änderungen und Ergänzungen betreffen folgende Verwaltungsaufgaben und Informationen zur Verwaltung:

- Konsole konfigurieren,
- SX-Board konfigurieren,
- Drucker an TACLAN verwalten,
- 2,88 MB SCSI-Disketten verwalten,
- Sicherung der Benutzerkennung,
- Nachträge aus den Freigabemittellungen zu SINIX V5.40.

Konsole konfigurieren (MX500)

Im Handbuch "Bediensystem für Systemverwalter" [2] ändert sich im Kapitel *Das Menü 'Konfigurieren'* im Abschnitt *Konsole konfigurieren* folgendes:

Der MX500 hat ein eigenes Board für die Konsole. Das Startfenster *Boards* enthält den Eintrag *Konsole*.

Der Befehl *Konfigurieren* öffnet das Dialogfenster *Konfiguration der Konsole*. Es bietet Ihnen die Möglichkeit, den Bildschirmtyp und die Tastatur der Konsole des MX500 umzukonfigurieren.

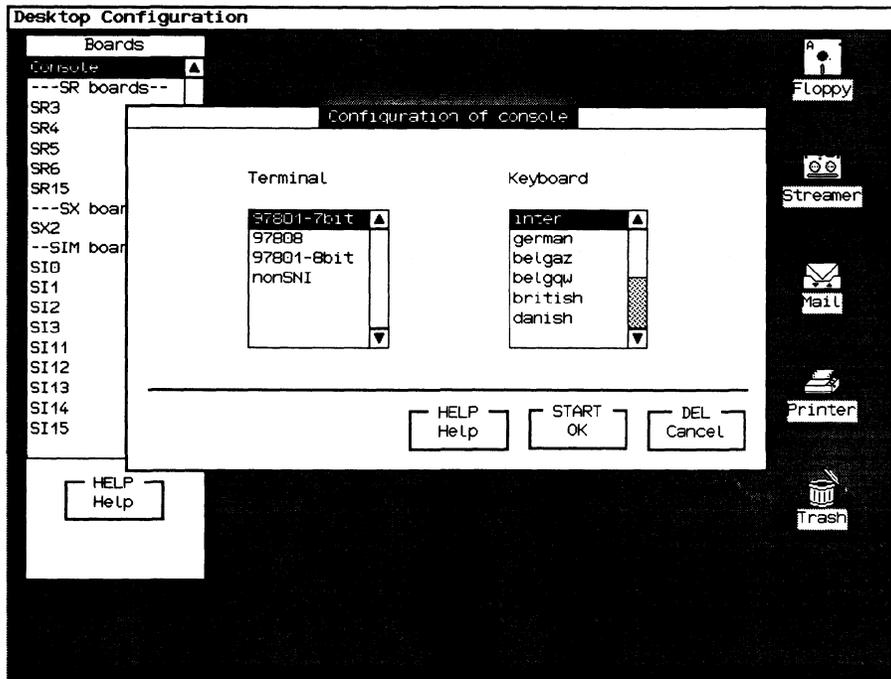


Bild 1: Das Menü 'Konfiguration der Konsole'

SX-Board konfigurieren

Die Beschreibung *SX-Board konfigurieren* im Handbuch "Bediensystem für Systemverwalter" [2] ändert sich. Die ursprünglich falsche Bezeichnung der Cluster-Bildschirme wird sowohl im folgenden Dialogfenster als auch in der anschließenden Beschreibung berichtigt.

Der Befehl *Konfigurieren* öffnet bei diesem Board das Dialogfenster *Konfigurierung von Board SXnn*. Es bietet Ihnen die Möglichkeit, am SX-Board *SXnn* Geräte zu konfigurieren oder zu dekonfigurieren. Das Board haben Sie zuvor im Startfenster *Boards* ausgewählt.

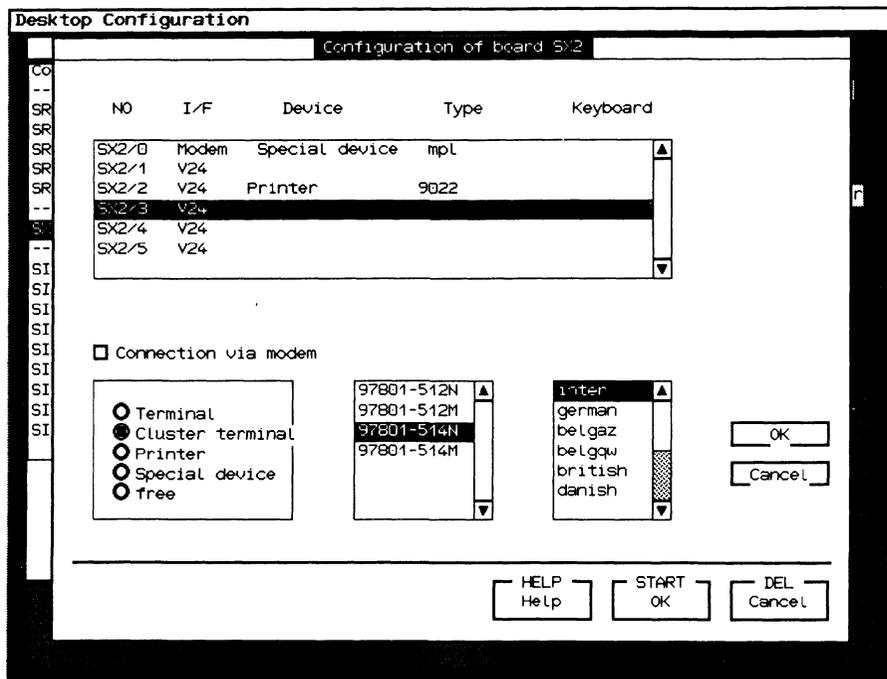


Bild 2: Das Dialogfenster 'Konfigurierung von Board SXnn'

Cluster-Bildschirm

Alle Typen von Cluster-Bildschirmen werden von der Konfigurierung unterstützt. Bei direktem Anschluß an die V24-Schnittstelle sind dies:

97801-512x
97801-514x

x kann bei beiden Bildschirmen 'N' oder 'M' sein. 'N' bedeutet, daß der jeweilige Cluster-Bildschirm im Normalbetrieb konfiguriert ist, 'M' bedeutet eine Konfigurierung im Multiplex-Betrieb, d.h. mit angeschlossenem Zusatzgerät. Wenn Sie zu einem Cluster-Bildschirm ein Zusatzgerät konfigurieren wollen, dann müssen Sie den Befehl *Konfigurieren* für den betreffenden Cluster-Bildschirm auswählen. Bei Anschluß über Modem sind dies:

- 514-1200-N Bildschirm 97801-514 im Normalbetrieb;
Übertragungsgeschwindigkeit 1200 Baud.
- 514-1200-M Bildschirm 97801-514 im Multiplex-Betrieb;
Übertragungsgeschwindigkeit 1200 Baud.
- 514-2400-N Bildschirm 97801-514 im Normalbetrieb;
Übertragungsgeschwindigkeit 2400 Baud.
- 514-2400-M Bildschirm 97801-514 im Multiplex-Betrieb;
Übertragungsgeschwindigkeit 2400 Baud.
- 514-9600-N Bildschirm 97801-514 im Normalbetrieb;
Übertragungsgeschwindigkeit 9600 Baud.
- 514-9600-M Bildschirm 97801-514 im Multiplex-Betrieb;
Übertragungsgeschwindigkeit 9600 Baud.

Drucker am TACLAN verwalten

Zum Handbuch "Bediensystem für Systemverwalter" [2] gibt es Ergänzungen zum Kapitel *Drucker am TACLAN verwalten*. Dieses Kapitel wird um die Abschnitte *Erzeugen* (Spoolverwaltung, Menü *Drucker*) und *Löschen* (Spoolverwaltung, Menü *Drucker*) erweitert.

Sie können als Systemverwalter im Bediensystem für Systemverwalter Drucker am TACLAN konfigurieren oder bereits konfigurierte TACLAN-Drucker wieder löschen.

Wählen Sie im Menü *Systemverwaltung* den Befehl *Spoolverwaltung* aus. Damit erhalten Sie die Oberfläche des Spoolverwaltungsprogramms.

Wählen Sie in dieser Oberfläche aus dem Menü *Einstellungen* den Befehl *Drucker*. Durch die Auswahl des Befehls *Drucker* wird ein Dialogfenster *Drucker* geöffnet und neben dem Menü *Einstellungen* das Menü *Drucker* eingeblendet. Das Menü *Drucker* enthält folgende Einträge:

- Sperren
- Freigeben
- Priorisieren
- Testen
- Vordruck setzen
- Voreinstellen
- Erzeugen *
- Löschen *

Die mit * gekennzeichneten Einträge sind im Handbuch "Bediensystem für Systemverwalter" [2] nicht beschrieben.

Nachdem Sie im Dialogfenster *Drucker* aus der Liste der eingerichteten Drucker einen Drucker markiert haben, können Sie ihn durch Auswahl eines Befehls im Menü *Drucker* bearbeiten.

Die Befehle *Erzeugen* und *Löschen* dienen zur Verwaltung von TACLAN-Druckern. Die beiden Befehle werden im folgenden beschrieben.

Erzeugen (Spoolverwaltung, Menü *Drucker*)

Der Befehl *Erzeugen* erlaubt es, einen Drucker am TACLAN zu konfigurieren. Dazu wird ein eigenes Dialogfenster für die benötigten Konfigurations-Daten geöffnet.

Links oben im Dialogfenster erscheint der lokale Druckername in der Form `DNxxx`, wobei `xxx` drei beliebige Ziffern sind.

Darunter finden Sie eine Liste, in der die möglichen Druckertypen aufgeführt sind. Aus dieser Liste wählen Sie den Typ des zu erzeugenden Druckers aus. Dadurch wird dieser in das Textfeld darunter eingetragen, das Sie nicht direkt verändern können.

Den Netznamen des Druckers wählen Sie aus der Liste in der rechten Hälfte des Dialogfensters aus, oder tragen diesen in das Textfeld darunter ein.

Dann tragen Sie die TCP-Portnummer für diesen Drucker in das Feld *Port-Nr.:* ein. Diese Portnummer wurde im Programm TACLAN.ADM verwendet, um den TACLAN-Drucker zu definieren. Die Nummer darf nicht kleiner als 1024 sein.

Im unteren Teil des Fensters gibt es zwei Einstellmarken: *Restart* und *Permanente Verbindung*. Mit *Permanente Verbindung* geben Sie an, ob die Verbindung zum Drucker ständig besteht oder nicht. Mit *Restart* können Sie das Spoolsystem nach dem Ende der Konfiguration neu starten.

Mit dem Quittierfeld *Ok* schließen Sie das Fenster. Die Konfiguration des TACLAN-Druckers wird dann so vorgenommen, wie Sie dies im Dialogfenster eingestellt haben. Je nachdem, ob Sie die Einstellmarke *Restart* markiert haben oder nicht, wird auch das Spoolsystem erneut gestartet oder nicht. Wird es nicht neu gestartet, so kann mit dem Drucker nicht gearbeitet werden.

Mit dem Quittierfeld *Abbruch* schließen Sie das Fenster, ohne daß der TACLAN-Drucker konfiguriert wird. Die gemachten Änderungen sind verloren.

Löschen (Spoolverwaltung, Menü *Drucker*)

Dieser Befehl löscht den TACLAN-Drucker, der im Dialogfenster markiert ist. Sie können nur solche Drucker mit diesem Befehl löschen, bei denen in der Anzeige im Dialogfenster *Drucker* unter der Spalte TACLAN ein *J* steht.

Wenn Sie *Löschen* auswählen, erscheint ein Fenster, das Sie auffordert, diese Aktion zu bestätigen. In diesem Fenster können Sie zusätzlich eine Einstellmarke *Restart* markieren. Wenn Sie *Restart* markieren, dann wird nach dem Löschen des TACLAN-Druckers das Spoolsystem neu gestartet. Markieren Sie *Restart* nicht, dann bleibt der Drucker bis zum nächsten Neustart noch ansprechbar.

2,88 MB SCSI-Disketten verwalten

Unter SINIX V5.41 wird am SCSI-Bus ein Diskettenlaufwerk für 3,5 Zoll große Disketten unterstützt. Diese Disketten haben eine Speicherkapazität von 2,88 MB in formatiertem Zustand (4 MB unformatiert).

Die weiteren zur Verfügung stehenden Diskettenformate und die entsprechenden Gerätedateien sind im Handbuch "Kommandos Band 3, Tabellen und Verzeichnisse" [14] im Abschnitt *Gerätedateien für Diskettenlaufwerke (Floppy Disk)* beschrieben. Dieser Abschnitt wird mit der folgenden Beschreibung des neuen SCSI-Diskettenformats und der entsprechenden Gerätedateien ergänzt.

Die zum Format der SCSI-Disketten zugehörigen Device-Knoten lauten:

```
/dev/[r]dsk/f/l3e[t]
```

- r kennzeichnet Gerätedateien mit direktem Zugriff (raw devices).
- l bezeichnet ein Diskettenlaufwerk (0 oder 1).
- 3 kennzeichnet, daß es sich um eine 3,5 Zoll Diskette handelt.
- e kennzeichnet, daß es sich um extra-hohe Schreibdichte handelt.
- t kennzeichnet, daß die gesamte Kapazität der Diskette zur Verfügung steht; fehlendes t bedeutet, daß die Kapazität der Diskette um die Anzahl Spuren von Zylinder 0 reduziert ist.

Die weiteren Kenndaten des neuen SCSI-Diskettenformats finden Sie in der folgenden Tabelle.

Name der Gerätedatei	Dichte	Anzahl Spuren	Sektoren je Spur	Bytes je Sektor	Kapazität in KB
[r]dsk/f03e	extra-hoch	158	36	512	2844
[r]dsk/f03et	extra-hoch	160	36	512	2880

Tabelle 6: Kenndaten des 2,88 MB SCSI-Diskettenformats

Um das neue SCSI-Diskettenformat in Ihrem System verfügbar zu machen, gehen Sie vor wie im Handbuch "Leitfaden für den Systemverwalter" [1], Kapitel *Verwalten von Geräteattributen*, Abschnitt *Hinzufügen eines SCSI-Geräts* beschrieben.

Wenn das Diskettenlaufwerk auf Ihrem System verfügbar ist, können Sie die neuen Disketten wie gewohnt mit dem Kommando `format` formatieren. Das Kommando `format` ist im Handbuch "Kommandos Band 1, A - K" [12] beschrieben.

Die zum Formatieren einer 2,88 MB SCSI-Diskette notwendigen Informationen entnehmen sie der Tabelle oben.

Beispiel

Um mit `format` eine 3,5-Zoll-SCSI-Diskette mit einer Speicherkapazität von 2,88 MB für die Benutzung vorzubereiten, legen Sie die Diskette in das entsprechende Laufwerk (hier 0) und geben ein:

```
format /dev/rdisk/f03et
```

Sicherung der Benutzerkennungen

Im Handbuch "Bediensystem für Systemverwalter" [2] gibt es eine geänderte Beschreibung zum Kapitel *Dateien zur Systemdatensicherung*, Abschnitt *Sicherung der Benutzerkennungen*.

Die Beschreibung, daß das Programm `loginsave` aus den Dateien `/etc/passwd` und `/etc/shadow` die lokalen Benutzerkennungen sichert, muß korrigiert werden. Die Datei `/etc/shadow` wird bei der Sicherung der Benutzerkennungen nicht berücksichtigt, das heißt, die Kennwörter der Benutzer müssen nach der Installation neu vergeben werden.

Damit trifft auch der Hinweis nicht zu, daß die Kennwörter der Benutzer gesichert werden.

Nachträge

Die nachfolgenden Abschnitte sind den Freigabemittellungen zu SINIX V5.40 entnommen. Diese Abschnitte gelten auch für SINIX V5.41 und sind in den bisher erschienenen SINIX-Handbüchern nicht dokumentiert. Folgende Handbücher sind betroffen:

- "Leitfaden für Systemverwalter" [1],
- "Referenzhandbuch für Systemverwalter" [11],
- "Kommandos, Band 3" [14],
- "DPTG-2 V1.0 Terminal-Multiplexer-Protokoll, Installationsanleitung" [16].

Die Zuordnung der folgenden Abschnitte zu den entsprechenden Handbüchern entnehmen Sie bitte der Tabelle 1: Änderungen (*), Ergänzungen (+) im Kapitel *Systemadministration*

Nachträgliche Installation einer Festplatte

Grundsätzlich sind nur Gerätedateien für die bei der Installation von SINIX V5.40 eingerichteten Festplatten vorhanden. Wird nachträglich eine weitere Festplatte eingebaut, so muß das Programm `diskadd` verwendet werden, um die für diese Platte notwendigen Geräteeinträge zu erzeugen. Außerdem übernimmt `diskadd` auch die Partitionierung der Platte (`fdisk`) und die Aufteilung in Slices (`disksetup`). Das Kommando `diskadd` ist im "Referenzhandbuch für Systemverwalter" [11] beschrieben.

Hinweis

Insgesamt können auf der logischen Partition bis zu 16 Slices eingerichtet sein. Slice 0 und Slice 7 werden automatisch generiert und systemintern genutzt. Der Benutzer sollte mit Gerätedateien betreffend Slice 7 grundsätzlich nicht arbeiten. Gerätedateien betreffend Slice 0 sollen nur dort verwendet werden, wo sie explizit angegeben sind.

Einrichten eigener Gerätedateien

Beim ersten Reboot nach dem Generieren eines neuen Kernels (mit `/etc/conf/bin/idbuild`) wird automatisch das Kommando `/etc/conf/bin/idmknod` gestartet. Dieses Kommando entfernt alle Gerätedateien für nicht benötigte Geräte ("non-required devices" haben kein "r" in der 3. Spalte des entsprechenden `/etc/conf/cf.d/mdevice`-Eintrags) im `/dev`-Verzeichnis plus Unterverzeichnisse.

Sollten Sie sich also für ein solchermaßen betroffenes Gerät mit `mknod` eine Gerätedatei (z.B. für VPSS, Terminal) unter `/dev` angelegt haben, so muß unbedingt auch ein Eintrag in die "node"-Files, im Verzeichnis `/etc/conf/node.d`, erfolgen (siehe auch "Referenzhandbuch für Systemverwalter" [11]: `mdevice(4)`, `sdevice(4)` und `idmknod(1M)`).

Anzahl der Pseudo-TTYs

Die Anzahl der zur Verfügung stehenden Pseudo-TTYs kann mit dem Shell-Skript `/usr/sbin/chgpty` geändert werden. Die Anzahl der Pseudo-TTYs kann auch im COLLAGE-Bediensystem über den Punkt *Konfiguration von Pseudo-ttys* eingestellt werden.

Protokolldatei des fsck

Alle Ausgaben und Meldungen des `fsck` werden in der Datei `/var/adm/log/FSCK` protokolliert.

Gerätedateien für Datenträger

Disketten

Eine Auflistung der Gerätedateien zum Ansprechen von Disketten befindet sich im Handbuch "Kommandos, Band 3" [14].

Formatieren:

Wird eine schon formatierte Diskette mit einem anderen Format neu formatiert, sollte das Formatierprogramm mehrmals (zwei- bis dreimal) aufgerufen werden, um unerwartete Fehler beim anschließenden Lesen und Beschreiben zu vermeiden (z.B. Diskette kann zwar beschrieben, aber nur teilweise gelesen werden).

Verwendung von Block-Device-Gerätedateien:

Blockorientierte Geräte (`/dev/dsk/*`) sind mit asynchronem Read-Ahead implementiert, daher wird beim Lesen von einem Block-Device das Diskettenende nicht erkannt. Für den Fall, daß dieses Verhalten unerwünscht ist, sollten die entsprechenden zeichenorientierten Geräte (`/dev/rdisk/*`) verwendet werden.

Magnetband-Kassette (MBK)

Eine Auflistung der Gerätedateien zum Ansprechen der Magnetband-Kassette befindet sich im Handbuch "Kommandos, Band 3" [14].

Kommando `/usr/ucb/mt`, Schalter `erase`:

Der Schalter `erase` des `/usr/ucb/mt`-Kommandos löscht nicht ab der aktuellen Bandposition, sondern es wird, unabhängig davon, wo der Schreib-Lese-Kopf gerade steht, immer das komplette Band gelöscht. Der Grund liegt darin, daß vom Streamercontroller nur das Löschen des kompletten Bandes unterstützt wird.

Das Schreiben einer Streamerkassette:

Das Schreiben einer Streamerkassette ist nur am Bandanfang oder nach dem Kommando `/usr/ucb/mt -f /dev/tapen eom` (d.h. am Ende des letzten Bandarchivs) erlaubt. Beim einfachen Positionieren durch Verwendung von `/dev/tapen` ist also kein Fortschreiben möglich.

Eine Auflistung der Gerätedateien und wichtige Hinweise zum Ansprechen des 2,3 GByte Magnetband-Kassetten-Laufwerks (ExaByte) befindet sich in diesem Kapitel im Abschnitt *Magnetband-Kassetten-Laufwerk 2,3 GByte*.

Hinweis

Sollen Video-8-Magnetband-Kassetten (ExaByte) mit dem Kommando `labelit` bearbeitet werden, so sind die Geräteeinträge unter `/dev/rmt` zu verwenden:

<code>/dev/exa0</code>	entspricht	<code>/dev/rmt/exalt0</code>
<code>/dev/exa0r</code>	entspricht	<code>/dev/rmt/exalt0v</code>
<code>/dev/exa8</code>	entspricht	<code>/dev/rmt/exalt0n</code>
<code>/dev/exa8r</code>	entspricht	<code>/dev/rmt/exalt0vn</code>

Plattenslices

Ein **Plattenslice** (i.a. eine Partition) kann unter SINIX V5.40 gleichzeitig als Dateisystem und Swapbereich eingerichtet werden. Es wird hier jedoch eindringlich vor der Verwendung dieser Funktionalitäten gewarnt, da sie im allgemeinen totale Datenverluste zur Folge haben (z.B. Dateisystemdaten werden durch Swapdaten überschrieben).

Änderung des Run-Levels

Sollten die Dateien `/etc/utmp` und `/etc/utmpx` nicht angelegt werden können, weil das Dateisystem voll ist, dann können mit `/etc/init[11213]` keine Änderungen des Run-Levels durchgeführt werden. Der Grund hierfür liegt darin, daß die Dateien `/etc/rc[11213]` (diese werden von `init` aufgerufen) das Kommando `who -r` aufrufen, welches in diesem Fall jedoch einen NULL-String liefert. In einem solchen Fall sollte man sich an der Anlage als `root` einloggen und für genügend Platz im Dateisystem sorgen.

Informationen über installierte Produkte

Unter SINIX V5.40 und SINIX V5.41 gibt es eine komfortable Möglichkeit zur Ermittlung der installierten Softwareprodukte und zwar mit dem Kommando `pkginfo`. In diesem Zusammenhang soll darauf hingewiesen werden, daß es das von SINIX V5.2x bekannte Verzeichnis `/usr/admin/.products` unter SINIX V5.40 nicht mehr gibt.

Magnetband-Kassetten-Laufwerk 60/150 MByte

Einschränkung:

Laufwerke verschiedener Kapazität sind aufwärtskompatibel, jedoch nicht abwärtskompatibel. D.h., Kassetten, die mit einem 60-MByte-Laufwerk beschrieben wurden, können von einem 150-MByte-Laufwerk gelesen werden. Jedoch können 150-MByte-Laufwerke keine 60 MByte-Kassetten beschreiben und Kassetten, die mit einem 150-MByte-Laufwerk beschrieben wurden, können von einem 60-MByte-Laufwerk nicht gelesen werden! Die "großen" Laufwerke beschreiben das Band mit der doppelten Anzahl Spuren wie die "kleinen" Laufwerke.

Magnetband-Kassetten-Laufwerk 2,3 Gbyte

Treiber / Kommandos:

Bei SINIX V5.40 und SINIX V5.41 stehen folgende Geräteeinträge zur Verfügung:

Gerätefile	interne Pufferung	Filemark bei close	Kassettenauswurf bei close	rewind
/dev/exa0 bzw. /dev/rmt/exalt0	ja	ja	nein	ja
/dev/exa0r bzw. /dev/rmt/exalt0v	nein	ja	nein	ja
/dev/exa8 bzw. /dev/rmt/exalt0n	ja	ja	nein	nein
/dev/exa8r bzw. /dev/rmt/exalt0vn	nein	ja	nein	nein

Tabelle 7: Geräteeinträge für 2,3 Gbyte Magnetband-Kassetten-Laufwerk

Bei den Gerätefile `exa0` und `exa8` erfolgt eine Zwischenpufferung der Daten im Treiber; dagegen erfolgt bei den Gerätefile `exa0r` und `exa8r` keine Pufferung der Daten. Um den Streaming-Modus des Laufwerks aufrecht zu erhalten, muß eine Zwischenpufferung im Treiber erfolgen. Daher sollten nach Möglichkeit nur die Gerätefile `/dev/exa0` und `/dev/exa8` verwendet werden.

Hinweis

Wird das Video8-Gerät in der ungepufferten Arbeitsweise (Gerätefile `/dev/exa8r` und `/dev/exa0r`) betrieben, so kann der Benutzer die Blocklänge spezifizieren, mit der auf das Band geschrieben wird. In SINIX V5.22 war die maximale Blocklänge auf 64 KByte begrenzt (für MX300 und MX500). Ab der Folgeversion mußte die maximale Blocklänge auf 32 Kbyte limitiert werden, was jedoch zur folgenden Inkompatibilität führt:

Bänder die unter SINIX V5.22 mit Blocklängen größer 32 Kbyte geschrieben wurden, können nicht mehr gelesen werden, da nur die ersten 32 Kbyte geliefert werden, und der Rest verloren geht. Schreib- und Leseaufträge mit einem Blockungsfaktor größer 32 Kbyte können weiterhin abgesetzt werden, jedoch werden diese in kleinere (max. 32 Kbyte) gestückelt.

1/2 Zoll Magnetbandgerät FS1000/FS2000

Neben den Funktionen des Magnetbandgerätes FS1000 (3504-160) unterstützt das Magnetbandgerät FS2000 (3504-625) auch Schreibdichten von 800 und 6250 bpi.

Wird eine Geschwindigkeit von 50 ips gewählt, so wird das Band gleichmäßiger transportiert als bei einer Geschwindigkeit von 100 ips.

Hinweis

Tritt beim Lesen/Schreiben auf das Magnetbandgerät ein Fehler auf, so müssen Sie unter Umständen 30 Minuten auf ein Timeout des Treibers warten bis das Magnetbandgerät wieder ansprechbar ist. Die Zeitdauer bis zum Timeout muß so lang gewählt werden, damit ein ggf. langlaufendes Kommando nicht abgebrochen wird und die folgenden Kommandos fehlerfrei ausgeführt werden können.

Konfigurieren von BA80-Terminals

Im Handbuch "DPTG-2 V1.0 Terminal-Multiplexer-Protokoll, Installationsanleitung" [16] sind im Abschnitt *COLLAGE-Menüs für DPTG-2* unter Menü *'local BA80 configuration'* die Funktionen zum Konfigurieren eines lokalen BA80-Terminal-Anschlusses erklärt. Bisher konnte über COLLAGE kein terminfo-Eintrag für einen über die Leitung betriebenen Terminaltyp generiert werden. Über ein weiteres Menü ist dies jetzt möglich.

Nachdem Sie im Menü *'local BA80 configuration'* die Funktion `configure` gewählt haben, erhalten Sie das Menü *'number of logins'*:

```

3      number of logins

number of login processes 3
label for ttysettings      ba80
value of TERM variable    ba80-06

```

number of login processes

Die Anzahl der Login-Bildschirme festlegen (0 - 8).

label for ttysettings

Das Label festlegen (ba80), über das aus der Datei `/etc/ttydefs` die Leitungsparameter ausgewählt werden.

value of TERM variable

Die TERM-Variable festlegen (ba80-06), mit deren Hilfe die Tastaturvariante bestimmt wird.

Netzwerke

Die in diesem Kapitel beschriebenen Änderungen und Ergänzungen beziehen sich auf das Handbuch "Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung" [4]. Folgende Themen sind betroffen:

- Ladbarer Ethernet-Prozessor 97882-143 (MX500),
- Einhängen der Rechner in das Netz,
- inet - Konfigurationsdatei,
- Asynchrones Schreiben über NFS,
- Netzüberwachung mit SNMP-Agenten.

Ladbarer Ethernet-Prozessor 97882-143 (MX500)

Ab Betriebssystem-Version SINIX V5.41 sind für TCP/IP neben dem bisherigen SCED-Board bis zu vier EXOS-Boards zugelassen und betreibbar. Pro EXOS-Board kann entweder TCP/IP oder ISO/LAN laufen.

Einhängen der Rechner in das Netz

Im Handbuch "Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung" [4] ändert sich im Kapitel *Installation* die Beschreibung des Abschnitts *Einhängen der Rechner in das Netz*.

Die Variable `INTERFACES` in der Datei `/etc/default/inet` kann mehrere Einträge enthalten. In diesem Fall bedeutet das:

- es wurden mehrere Controller-Schnittstellen spezifiziert oder
- eine Schnittstellenbeschreibung enthält neben dem Schnittstellen-Parameter *Schnittstelle* noch andere Angaben (z.B. `exos0:danzig`).

Die Einträge aktivieren Sie, indem Sie das Shell-Skript `/etc/init.d/inet` mit den Parametern `start` oder `stop` starten. Mit `start` können Sie den LAN-Anschluß aktivieren, mit `stop` deaktivieren Sie ihn. Der Aufruf für das Shell-Skript hat folgende Form:

```
sh /etc/init.d/inet <parameter>
```

Wenn `INTERFACES` mehrere Einträge enthält, werden diese Spezifikationen beim Einhängen des Rechners mittels Menüsystem übernommen. Dann erscheint der Hinweis:

Die Schnittstellen werden entsprechend der Spezifikation
in `/etc/default/inet` konfiguriert

Wenn mehrere Controller-Schnittstellen spezifiziert wurden, werden Sie nicht dazu aufgefordert, den Namen des Controllers einzugeben. Er wird in diesem Fall aus der Datei `/etc/default/inet` übernommen.

Wenn keine oder nur eine Controller-Schnittstelle (ohne weitere Angaben) spezifiziert wurde und mehrere Controller vorhanden sind, erscheint folgende Eingabemaske:

Name des Ethernet-Controllers: _____

Durch Drücken der Leertaste bekommen Sie der Reihe nach alle Controller zur Auswahl angezeigt.

inet - Konfigurationsdatei

Dieser Abschnitt wurde dem Handbuch "Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung" [4], Kapitel *Kommandos und Dämonen, inet - Konfigurationsdatei* entnommen. Die Beschreibung des Abschnitts *INTERFACES* wurde um die `slip`-Spezifikation ergänzt (siehe auch "Netzwerke - Leitfaden für Benutzer und Verwalter" [3]).

INTERFACES

Als Wert können Sie eine Liste von Schnittstellen-Spezifikationen angeben. Die einzelnen Einträge der Liste müssen durch Leerzeichen und/oder Tabulatoren voneinander getrennt sein.

Falls der Variablen-Wert nicht existiert oder die Variable nicht in `/etc/default/inet` aufgeführt ist, werden die zu konfigurierenden Schnittstellen mittels `etherstat -e` und `uname -n` automatisch bestimmt. Das Kommando `etherstat -e` ermittelt den Schnittstellen-Parameter *schnittstelle* und `uname -n` den Schnittstellen-Parameter *adresse* für den `ifconfig`-Aufruf.

Jede Schnittstellen-Spezifikation ist wiederum eine Parameterliste. Die Parameter einer Schnittstelle müssen durch einen Doppelpunkt voneinander getrennt werden.

Beginnt die Parameterliste mit `slip`, dann spezifizieren die nachfolgenden Parameter eine `slip`-Konfigurierung. Die auf `slip` folgenden Parameter müssen den Angaben im Kapitel *SLIP - TCP/IP-Verbindung über serielle Leitungen* entsprechen.

Der Aufbau einer `slip`-Spezifikation lautet:

```
slip:tty:[baudrate]:[mtu]:[ip-source]:ip-des[:options]
```

Die folgenden optionalen Angaben haben die Voreinstellwerte:

```
baudrate      9600
mtu           256
ip-source     <hostname>
```

Den Rechnernamen *hostname* bestimmen Sie mittels `uname -n`. Falls die Parameterliste nicht mit `slip` beginnt, müssen die Parameter den Angaben in `ifconfig(1M)` entsprechen.

Existiert nur ein Eintrag in der Parameterliste, dann muß der Eintrag den Parameter *schnittstelle* spezifizieren. Der Parameter *adresse* wird mittels `uname -n` bestimmt.

Beinhaltet die Parameterliste mehr als einen Eintrag, dann werden die ersten zwei Einträge als *schnittstelle* und *adresse* angenommen.

parameter-Einträge in der Parameterliste wie `trailers`, `arp`, etc. überschreiben entsprechende Einträge in `ETHERFLAGS`.

Wenn mehr als eine Schnittstelle konfiguriert werden soll, müssen die Parameter der Schnittstellen immer Werte für die Schnittstellen-Parameter *schnittstelle* und *adresse* beinhalten. Die Beschreibung der Parameter finden Sie in folgenden Handbüchern:

<i>schnittstelle</i>	siehe unten <code>ifconfig(1M)</code> .
<i>parameter</i>	siehe unten <code>ifconfig(1M)</code> .
<i>adresse</i>	siehe unten <code>ifconfig(1M)</code> .
<i>etherstat</i>	siehe "Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung" [4].
<i>uname</i>	siehe "Kommandos, Band 2" [13].
<code>ifconfig(1M)</code>	siehe "Netzwerke - Leitfaden für Benutzer und Verwalter" [3] und "Referenzhandbuch für Systemverwalter" [11].

Asynchrones Schreiben über NFS

Das Handbuch "Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung" [4] wird um die Beschreibung *Asynchrones Schreiben über NFS (Network File System)* ergänzt. Sie erfahren, wie Sie das synchrone Schreiben über NFS in ein asynchrones Schreiben ändern können und umgekehrt.

Als Standard ist auf dem Server synchrones Schreiben eingerichtet. Dabei sendet der Server eine Quittung, sobald die Daten auf Platte geschrieben wurden. Beim asynchronen Schreiben hingegen sendet der Server eine Quittung, sobald die Daten auf dem Server angekommen sind; die Daten müssen noch nicht auf Platte geschrieben sein.

Um synchrones Schreiben in ein asynchrones Schreiben zu ändern (oder umgekehrt), müssen Sie mit dem Programm `adb` den Kernel patchen. Wie Sie das machen, sehen Sie in den beiden folgenden Beispielen:

1. Umschalten auf asynchrones Schreiben

```
# adb -w /unix  
nfs_writeflags?w 0  
END /* oder Tastenkombination CTRL d */
```

Danach müssen Sie das System neu booten.

2. Umschalten auf synchrones Schreiben

```
# adb -w /unix  
nfs_writeflags?w 2  
END /* oder Tastenkombination CTRL d */
```

Danach müssen Sie das System wieder neu booten.

Vorsicht

Bitte beachten Sie, daß die Asynchron-Einstellung auch zu Problemen führen kann. Es ist nicht garantiert, daß ein dem Client als beendeter Schreibvorgang gemeldeter NFS-Auftrag tatsächlich auf die Platte geschrieben wurde. Ein zwischenzeitlicher Systemabsturz kann zu Konsistenzproblemen des Datenbestands führen.

Netzüberwachung mit SNMP-Agenten

Dieses Kapitel wurde dem Handbuch "Netzwerke - Leitfaden für Benutzer und Verwalter, Ergänzung" [4] entnommen und für MX300/MX500 SINIX V5.41 überarbeitet. Im wesentlichen hat sich in diesem die Struktur und der Inhalt der Abschnitte *Installieren des SNMP-Agent*, *Start des SNMP-Agent* und *Rekonfiguration des SNMP-Agent* geändert.

Der Ausfall eines Netzes kann hohe Kosten und große Probleme verursachen. Deshalb ist es wichtig, das Netz und seine Komponenten, z.B. Bridges, Router, Gateways und SINIX-Systeme, zu überwachen, Probleme zu erkennen und entsprechende Maßnahmen rechtzeitig einzuleiten.

Mit dem SNMP-Agent (Simple-Network-Management-Protocol-Agent) wird auf Ihrem SINIX-System ein Hilfsmittel zur Überwachung von Netzen mit TCP/IP-Protokollen bereitgestellt.

Standardmäßig wird die SNMP-Agenten-Software (`Sisnmpd` und `SItcpeXt`) mit der Installation des Betriebssystems installiert.

Beim Systemstart wird der SNMP-Agent automatisch mit Voreinstellungen gestartet.

In diesem Abschnitt finden Sie Beschreibungen zur

- Funktion des SNMP-Agenten,
- Installation des SNMP-Agenten für den Fall, daß der SNMP-Agent auf Ihrem Rechner deinstalliert wurde,
- Rekonfiguration des SNMP-Agenten, falls Sie die aktuellen Einstellungen mit dem Kommando `snmpadm` ändern möchten.

Funktion des SNMP-Agenten

SNMP-Agenten und Netzmanagementstationen

In einem Netz gibt es eine oder mehrere Netzmanagementstationen, von denen aus das Netz überwacht und verwaltet wird. Diese Managementstationen müssen mit den zu überwachenden Netzkomponenten, genauer gesagt, mit deren Agenten, über ein Protokoll kommunizieren. Für TCP/IP-LANs verwendet SINIX hierzu das herstellerunabhängige Protokoll SNMP. Dieses Protokoll muß sowohl in der Managementstation, wie auch in den zu überwachenden Komponenten implementiert sein.

Der SNMP-Agent läuft als Dämonprozeß. Das Prinzip von SNMP ist, daß die Managementstation der aktive Part ist. Dadurch werden die zu überwachenden Komponenten durch das Management so wenig wie möglich belastet. Der SNMP-Agent wartet, bis er eine Nachricht von der Managementstation empfängt. Erst dann wird er aktiv und liefert

die von der Managementstation angeforderte Information. Da SNMP ein herstellerunabhängiges Protokoll ist, kann ein Rechner mit der Betriebssystemversion SINIX V5.41 in das Netzmanagement jeder SNMP-fähigen Netzmanagementstation einbezogen werden.

Die Netzmanagementstation auf SINIX-Systemen ist das Produkt TRANSVIEW-SNMP (siehe Handbuch "TRANSVIEW-SNMP V1.0, Bedienung und Funktion")

Domänen und Community-Namen

Überwachen und verwalten mehrere Managementstationen ein Netz, dann müssen Domänen eingerichtet werden, um die Wirkungskreise der einzelnen Managementstationen abzugrenzen. SNMP bietet dazu ein einfaches Konzept:

Eine Community ist eine Beziehung zwischen einem SNMP-Agenten und einer oder mehrerer Managementstationen. Eine Community wird ausgedrückt durch einen Community-Namen, der mit jeder Nachricht mitgeliefert wird und der damit die Zugehörigkeit zur Community ausdrückt.

Mit einem Community-Namen sind auch die Zugriffsrechte, wie zum Beispiel "read-only" oder "read-write" verbunden. Dieses Recht kann jedoch die in der MIB-Definition (z.B. Standard RFC 1213 MIB II-Management-Information-Base, Management for TCP/IP-based Internets) für die einzelnen Objekte (zu verwaltende Informationen) vorgesehenen Operationen nur einschränken, jedoch nicht erweitern.

Ist für ein Objekt laut MIB Definition "read-only" festgelegt, so kann es nicht als "read-write" verwendet werden, auch wenn der Community-Name mit dem Zugriffsrecht "read-write" verbunden ist.

Folgendes Beispiel soll den Gebrauch von Community-Namen und Zugriffsrechten verdeutlichen:

Ein SNMP-Agent gehört zu einer Community mit Namen `public` und dem Zugriffsrecht "read-only". Zur Community `public` gehören mehrere Managementstationen. Jede dieser Managementstationen kann durch Senden einer entsprechenden Nachricht mit dem Community-Namen `public` von diesem SNMP-Agenten Informationen abfragen. Zugleich gehört der SNMP-Agent zu einer zweiten Community mit Namen `net_5`, mit dem das Zugriffsrecht "read-write" verknüpft ist. Zur Community `net_5` gehört nur eine Managementstation. In diesem Beispiel ist nur die Managementstation der Community `net_5` berechtigt, Schreiboperationen auf dem SNMP-Agenten auszuführen.

Treten bei einer Netzkomponenten besondere Ereignisse auf, so kann der SNMP-Agent eine Trap-Nachricht an eine oder mehrere Managementstationen senden, um diese davon zu unterrichten. Ein solches Ereignis wäre z.B., daß ein neuer SNMP-Agent im Netz die Managementstationen auf seine Existenz aufmerksam machen will. Netzmanagementstationen können dadurch auf solche Ereignisse reagieren ("trap-controlled

polling"). Die Berechtigung eines SNMP-Agenten, an eine Managementstation eine Trap-Nachricht zu senden, wird ebenfalls durch einen Community-Namen ausgedrückt (siehe Abschnitt *Installieren des SNMP-Agenten*). Will der SNMP-Agent eine Trap-Nachricht an eine Managementstation senden, so muß er den dazu nötigen Trap-Community-Namen verwenden, damit die Managementstation diese Nachricht akzeptiert.

Operationen des SNMP-Agenten unter SINIX V5.41

Der mit dem Betriebssystem SINIX V5.41 ausgelieferte SNMP-Agent ermöglicht das Abfragen und Ändern aller Objekte der MIB-II (RFC 1213) durch die Managementstation. Die Informationen der MIB-II sind folgendermaßen gegliedert:

Ebene	Information
System	Name des Knotens, Betriebssystemversion, Name der Kontaktperson, Standort des Rechners, ...
IP-Ebene (Internet-Protocol)	Subnet-Maske, Broadcast-Adresse, Routing-Tabelle und zahlreiche Statistikwerte.
ICMP-Ebene	Statistikwerte.
TCP-Ebene	Anzahl aktiver und passiver Verbindungen, Statistikdaten, TCP-Verbindungstabelle, ...
UDP-Ebene	UDP-Tabelle.
SNMP-Agenten	Statistikwerte.

Tabelle 8: Informationen der MIB-II

Eine genaue Beschreibung aller Objekte der MIB-II (RFC 1213) finden Sie in der Datei `/usr/lib/snmpd/doc/mib-II.doc`.

Installieren des SNMP-Agenten

Dieser Abschnitt ist für Sie nur von Bedeutung, falls der SNMP-Agent auf Ihrem Rechner deinstalliert wurde. Standardmäßig wird der SNMP-Agent mit der Installation des Betriebssystems installiert und automatisch gestartet. Beim Start verwendet der SNMP-Agent Voreinstellungen. Sie können zu einem späteren Zeitpunkt diese Voreinstellungen mit Hilfe des Kommandos `snmpadm` ändern (siehe Abschnitt *Rekonfiguration des SNMP-Agenten*).

Gehen Sie bei der Installation folgendermaßen vor:

- Installieren der SNMP-Software (Softwarepakete `SIsnmpd` und `SItcpevt`)
- Überprüfen der Einträge in der Verwaltungsdatei `/etc/services`.
- Start des SNMP-Agenten.
- Eventuell eine Rekonfiguration des SNMP-Agenten.

Installieren der SNMP-Software

Bei der Installation der SNMP-Software wird vorausgesetzt, daß folgende Software auf Ihrem System bereits vorhanden ist:

- Network-Support-Utilities (NSU),
- Internet-Utilities (INET).

Diese Softwarepakete gehören zum Lieferumfang von SINIX V5.41.

Durch Installation und Start des Betriebssystems wird der SNMP-Agent automatisch installiert bzw. gestartet.

Nur wenn der SNMP-Agent auf Ihrem Rechner deinstalliert wurde, müssen Sie ihn explizit installieren. (Mit Hilfe des Kommandos `pkgadd` können Sie die Softwarepakete `SIsnmpd` und `SItcpevt` installieren.)

Der SNMP-Agent besteht aus zwei Softwarepaketen:

`SIsnmpd`

ist der "Core Agent" und unterstützt die Objekte der Gruppen "Systeminformation" und "Information über den SNMP-Agenten".

`SItcpevt`

ist eine Erweiterung des "Core Agenten", die die anderen Objekte der MIB-II unterstützt.

Erweitern der Verwaltungsdatei `services`

Der SNMP-Agent bietet der Netzmanagementstation Dienste an. Diese Dienste werden in der Datei `/etc/services` eingetragen. In dieser Datei sind die Namen der vorhandenen IP-Dienste mit den dazugehörigen Anschlußnummern aufgeführt. Diese Informationen werden von Programmen gelesen, die Netzdienste anfordern. Die Datei `/etc/services` wird bei der Installation von TCP/IP automatisch angelegt und enthält normalerweise bereits die entsprechenden Einträge für den SNMP-Agent.

Bei der Installation des SNMP-Agenten wird jedoch nochmals geprüft, ob die Einträge vorhanden sind. Für den SNMP-Agent sind folgende Einträge in der Datei `/etc/services` notwendig:

```
snmp          161/udp
snmp-trap     162/udp
```

Die Namen der Netzdienste sind in der ersten Spalte aufgeführt. Die zweite Spalte enthält die Anschlußnummer (port) des Dienstes und den Namen des Transportprotokolls (UDP), unter dem der Dienst arbeitet.

Sind diese Einträge noch nicht vorhanden, dann werden sie automatisch in die lokale Datei `/etc/services` eingetragen. Wird jedoch die Verwaltungsdatei von Ihrem Netzwerkverwalter zentral über NIS verwaltet, so müssen Sie bei ihm einen entsprechenden Eintrag in der Datei `/etc/services` veranlassen. Bei einer Aktualisierung der lokalen Datei `/etc/services` von der zentralen Datei aus würde sonst der für den SNMP-Agenten nötige Eintrag in der lokalen Datei überschrieben. Deshalb erscheint der folgende Hinweis auf dem Bildschirm:

```
If you are running NIS, insert this line manually to
/etc/inet/services in your NIS-Master-Server !
```

Start/Stop des SNMP-Agenten

Der SNMP-Agent wird automatisch gestartet, sobald der Run-Level 2 erreicht wird und automatisch gestoppt, wenn der Rechner in den Run-Level 0 wechselt (siehe "Referenzhandbuch für Systemverwalter" [11], `init(1M)`).

Sie können den SNMP-Agenten aber auch explizit durch Eingabe des folgenden Kommandos starten:

```
/etc/init.d/snmpd start
```

Dadurch wird der SNMP-Agent als Dämonprozeß gestartet. Das System gibt neben einer entsprechenden Meldung die Werte einiger Objekte zur Startzeit aus und meldet sich mit dem Prompt-Zeichen zurück. Der SNMP-Agent wartet jetzt und ist bereit, Nachrichten von den Netzmanagementstationen zu empfangen.

Ist der Dämonprozeß `snmpd` bereits aktiv, wenn Sie das Startskript `/etc/init.d/snmpd` aufrufen, dann ist dies ohne Wirkung, da im Startskript zunächst geprüft wird, ob der Dämonprozeß `snmpd` bereits gestartet wurde.

Sie können den SNMP-Agenten auch explizit mit dem folgenden Kommando anhalten:

```
/etc/init.d/snmpd stop
```

Rekonfiguration des SNMP-Agenten

Der SNMP-Agent wird immer mit Standard-Voreinstellungen gestartet. Um die Einstellungen anschließend Ihren Verhältnissen anzupassen, gehen Sie folgendermaßen vor:

- Festlegen der Community-Namen
- Änderung der Einstellungen mit dem Kommando `/sbin/snmpadm`

Sie sollten die Konfigurationsdateien nicht direkt mit dem Editor verändern, denn nur über das Kommando `/sbin/snmpadm` ist sichergestellt, daß die Änderungen syntaktisch korrekt in die Konfigurationsdateien übernommen werden.

Festlegen der Community-Namen

Sie erhalten den SNMP-Agenten vorkonfiguriert, so daß er jeder Managementstation unter dem Community-Namen `public` lesenden Zugriff gewährt. Trap-Nachrichten werden gemäß dieser Vorkonfiguration nicht an Managementstationen geschickt.

Für eine Ihren Verhältnissen angepaßte Konfiguration erfragen Sie bitte von Ihrem Netzwerk-Verwalter die Ihnen zugeordneten Community-Namen zusammen mit den damit verknüpften Zugriffsrechten. Im allgemeinen Fall sind dies die Fragen:

- Mit welchen Community-Namen soll der SNMP-Agent das Zugriffsrecht "read-only" verbinden?
- Sollen Nachrichten mit diesem Community-Namen von beliebigen Netzmanagementstationen akzeptiert werden oder nur von einer oder speziellen?
- Welche Internet-Adressen haben diese Managementstationen?
- Mit welchen Community-Namen soll der SNMP-Agent das Zugriffsrecht "read-write" verbinden?
- Sollen Nachrichten mit diesem Community-Namen von beliebigen Netzmanagementstationen akzeptiert werden oder nur von einer oder speziellen?
- Welche Internet-Adressen haben diese Managementstationen?
- Wie heißt die Internet-Adresse der Managementstation(en), an die der SNMP-Agent Trap-Nachrichten senden soll?
- Welchen Community-Namen soll er diesen Trap-Nachrichten beifügen?

Änderung der Einstellungen mit dem Kommando `snmpadm`

Mit dem Kommando `/sbin/snmpadm` können Sie die Konfigurationsdateien des SNMP-Agenten verändern. Wenn Sie dieses Kommando aufrufen, wird die entsprechende Information im Dialog abgefragt. Nach der Änderung der Konfigurationsdateien wird der SNMP-Agent automatisch angehalten und neu gestartet. Zum Aufruf von `snmpadm` müssen Sie die Zugriffsrechte der Benutzerkennung `root` haben.

Zuerst wird die Datei `AgentConfig` bearbeitet. Zunächst werden die Einträge für die beiden Parameter `LOCATION` und `CONTACT` angezeigt. Danach wird jeder Eintrag einzeln angezeigt mit der Frage, ob er geändert werden soll. Gegebenenfalls wird dann die benötigte Information abgefragt.

Als nächstes wird die Datei `Communities` bearbeitet. Zunächst wird ihr gesamter Inhalt angezeigt. Jede Zeile enthält eine `SnmpCommunity`, bestehend aus dem Community-Namen, der Internet-Adresse der Managementstation und dem Zugriffsrecht. Aus dieser Datei können Sie Zeilen löschen und später neue hinzufügen, indem Sie auf die Frage

`add a management station?`

mit `y(es)` antworten.

Für jede Managementstation, die hinzuzufügen ist, wird folgende Information abgefragt:

- `/sbin/snmpadm` fragt zunächst nach dem Community-Namen:

```
community_name [?]
```

Geben Sie den Namen der Community ein, z.B. `public`

- Anschließend fragt `/sbin/snmpadm` nach der Internet-Adresse der Managementstation, die zu dieser Community gehört:

```
ip address (form: n.n.n.n, 0.0.0.0 for any address) [?]
```

Geben Sie die Internet-Adresse ein, z.B. in der Form `"129.144.50.56"`. Wenn Sie `"0.0.0.0"` eingeben, so erlaubt der SNMP-Agent jeder Managementstation, die den angegebenen Community-Namen verwendet, den entsprechenden Zugriff.

- Jetzt werden Sie nach den Zugriffsrechten gefragt, die mit diesem Community-Namen verbunden sein sollen:

```
access rights for the management station
```

```
1 READ  
2 WRITE  
3 NONE
```

```
Enter selection [?,?.q]:
```

Wählen Sie eine 1 für "read-only" oder eine 2 für "read-write". Wollen Sie die angegebene Managementstation von jeglichem Zugriff ausschließen, so können Sie dies durch Eingabe einer 3 erreichen.

Wenn Sie verschiedenen Managementstationen den Zugriff über den gleichen Community-Namen erlauben wollen, dann durchlaufen Sie diese drei Punkte mehrmals und geben dabei jeweils den gleichen Community-Namen und die gleichen Zugriffsrechte an.

Als letztes wird die Datei `TrapComm` bearbeitet. Zunächst wird ihr gesamter Inhalt angezeigt. Jede Zeile enthält eine `SnmpTrapCommunity`, bestehend aus dem Community-Namen und der Internet-Adresse der Managementstation, an die unter diesem Community-Namen Trap-Nachrichten gesendet werden sollen. Aus dieser Datei können Sie Zeilen löschen und später neue hinzufügen, indem Sie auf die Frage:

```
add a trap-management station?
```

mit `y(es)` antworten.

Für jede Managementstation, die hinzuzufügen ist, werden die folgenden Informationen abgefragt:

- `/sbin/snmpadm` fragt zunächst nach dem Community-Namen der Managementstation, an die Trap-Nachrichten verschickt werden sollen:

```
trap-community-name of the station to receive traps [?]
```

Geben Sie den Community-Namen ein, z.B. `master_3`.

- Anschließend fragt `/sbin/snmpadm` nach der Internet-Adresse der Management-Station, an die Trap-Nachrichten geschickt werden sollen:

```
ip address of station to receive traps (form: n.n.n.n) [?]
```

Geben Sie die Internet-Adresse ein, z.B. `127.144.50.17`.

Nach der Änderung der Konfigurationsdateien muß der SNMP-Agent angehalten und neu gestartet werden. Dieser sogenannte Restart des SNMP-Agenten wird bei einer Änderung der Konfigurationsdateien mit dem Werkzeug `/sbin/snmpadm` automatisch durchgeführt. Nur wenn Sie die Konfigurationsdatei mit einem Editor verändert haben, müssen Sie den Restart des SNMP-Agenten selbst durchführen.

Dazu geben Sie das folgende Kommando ein:

```
kill -1 Prozeßnummer_des_SNMP-Agenten
```

Die Parameter `CONTACT` und `LOCATION` können auch von einer Managementstation aus verändert werden, die entsprechende Zugriffsrechte hat (Community-Name mit "read-write"-Zugriff). Die Konfigurationsdateien werden dadurch ebenfalls verändert.

COLLAGE

Die in diesem Kapitel beschriebenen Änderungen und Ergänzungen beziehen sich auf die Handbücher

- "COLLAGE-Bediensystem" [5],
- "COLLAGE V4.0 Bedienen und Verwalten" [6].

Folgende Themen sind betroffen:

- Das Menü 'Druckaufträge',
- Neuen Editor eintragen,
- Hinweise zur Bedienung von Alpha-COLLAGE.

Das Menü 'Druckaufträge'

Geändert hat sich im Handbuch "COLLAGE-Bediensystem" [5] im Kapitel *Menüs und Service-Programme* der Abschnitt *Das Menü Druckaufträge*. Der Unterabschnitt *Vordruck* wird wie folgt korrigiert:

Unter dem COLLAGE-Bediensystem V5.41 ist es nicht mehr möglich, Kopfzeilen zu drucken. Das Kommando `pr` kann nicht mehr verwendet werden.

Neuen Editor eintragen

Zum Handbuch "COLLAGE-Bediensystem" [5] gibt es eine Ergänzung zum Kapitel *Einführung in das COLLAGE-Bediensystem*, Abschnitt *Der Editor ced*. Das Kapitel wird um den Unterabschnitt *Neuen Editor eintragen* erweitert.

Standardmäßig steht Ihnen unter dem Befehl *Öffnen* im Menü *Bearbeitung* der Editor *ced* zur Verfügung. Wenn Sie einen weiteren Editor zur Auswahl haben wollen, müssen Sie die beiden PIF-Dateien *METHODS* und *PROGRAMS* ändern.

Wechseln Sie in das sprachabhängige Verzeichnis */usr/lib/col/PIF/De* oder */usr/lib/col/PIF/collmen/De* (siehe "COLLAGE-Bediensystem" [5] im Abschnitt 5.2 *PIF's (Program Information Files) zum Konfigurieren der Bedienoberfläche*. Dort finden Sie die PIF-Dateien.

Öffnen Sie die Datei *METHODS*. Vergeben Sie in einer neuen Zeile einen Editoraufruf und weisen Sie ihm einen Editor zu. Die Zeile muß die Form haben:

```
u_Editor=neuEd $:\
```

Sie können zum Beispiel zusätzlich zum Editor *ced* den Editor *MAXed* als Anwendung eingetragen haben:

```
Editieren=ced $:\
M_editieren=MAXed $:\
```

Schließen Sie nun *METHODS* und öffnen Sie *PROGRAMS*. Kopieren Sie die Zeile, in der der Editor *ced* (oder ein anderer Editor) eingetragen ist. Ersetzen Sie am Anfang der kopierten Zeile den Namen *ced* durch den Namen des neuen Editors. Wenn Sie z.B. den Editor *MAXed* eingetragen haben, können die Zeilen diese Form haben:

```
ced:OS:ISAR16,25,45,-1,-1
MAXed:OS:ISAR16,25,45,-1,-1
```

Schließen Sie die Datei *PROGRAMS* wieder.

Hinweise zur Bedienung von Alpha-COLLAGE

Die in diesem Abschnitt beschriebenen Hinweise zur Bedienung von Alpha-COLLAGE beziehen sich auf das Kapitel *Alpha-COLLAGE bedienen* im Handbuch "COLLAGE V4.0" [6]. Dort finden Sie auch weitergehende Informationen und Details zu den Themen dieses Abschnitts.

In einem Shell-Fenster arbeiten

Wenn Sie unter dem COLLAGE-Bediensystem im Menü *Desktop* mit dem Befehl *Anwendungen* ein Fenster mit einer SINIX-Shell öffnen, ist es sicherer, auch die Sondertaste *gesamter Bildschirm* zu drücken. Grund: Unter SINIX werden die Ausgaben auf die Größe des gesamten Bildschirms abgestimmt. Wenn Sie in einem kleineren Fenster arbeiten, gehen unter Umständen Teile der Ausgaben verloren.

Sondertaste 'Ausgabe unterdrücken - ganz'

Die Funktionstaste *Ausgabe unterdrücken - ganz* arbeitet nur fehlerfrei in Fenstern, in denen eine SINIX-Shell aufgerufen ist (Menü *Desktop*, Befehl *Anwendungen*). In anderen Fenstern des COLLAGE Bediensystems ist eine Verwendung dieser Taste nicht zu empfehlen.

C-Programmierung

CES V5.41 erhält funktionale Erweiterungen, die in den folgenden Handbüchern noch nicht enthalten sind:

- "CES, Leitfaden und Werkzeuge für die Programmierung mit C" [7],
- "CES, cc-Kommando für MX300 und WX200" [8],
- "CES V5.40/V5.41, Referenzhandbuch für den Programmierer" [9].

Die Erweiterungen betreffen:

- ABI386-Konformität,
- Optionen zur Optimierung und Codegenerierung,
- CES-Abweichungen vom ANSI-Standard,
- Installation von Fortran77-Komponenten,
- Post&Wait-Treiber (MX500),
- Asynchronous Raw I/O (MX500).

Hinweis

Die Beschreibung im Handbuch "CES, cc-Kommando für MX300 und WX200" [8] gilt auch für MX500.

ABI386-Konformität

CES V5.41 verhält sich ab SINIX V5.41 ABI386-konform. Das bedeutet, daß bei der Funktionsrückgabe von `char`- und `short`-Funktionen die entsprechenden Register vorzeichengerecht auf 32 Bit erweitert werden. Durch diese Registererweiterung ist das problemlose Mischen von AT&T- und CES-Objekten sichergestellt.

Optionen zur Optimierung und Codegenerierung

Die bisherige Beschreibung in den Handbüchern "CES, Leitfaden und Werkzeuge für die Programmierung mit C" [7] und "CES, cc-Kommando für MX300 und WX200" [8] wird erweitert um die Optionen für *Inline Expandierung von Benutzerfunktionen* (OI) und *Verbesserte Sprungoptimierung* (FB).

Inline Expandierung von Benutzerfunktionen

-OI

(Optimizer Inline). Der Optimierer versucht, Funktionen `inline` zu expandieren. Das heißt, Funktionen werden an ihren Aufrufstellen durch den Funktionscode ersetzt. Durch diese Optimierung kann der Entry- und Exit-Code eingespart werden, der vor allem bei kurzen Funktionen relativ zeitaufwendig ist.

Funktionen, die nicht `static` deklariert sind oder deren Adresse angesprochen wurde, bleiben mit ihrer Definition im Modul erhalten. An Aufrufstellen innerhalb des Moduls können sie jedoch `inline` expandiert werden.

Es hat sich gezeigt, daß der Inliner nur bei bestimmten Programmstrukturen zu Performance-Verbesserungen führt:

- Der Inliner bringt Performance-Verbesserung, wenn der Linkage-Anteil gegenüber dem eigentlichen Funktionscode einen großen Teil einnimmt. D.h. bei Verwendung von kurzen Funktionen kann ein Inliner helfen. Dies ist insbesondere bei Entwicklung abstrakter Datentypen der Fall, da dort auch einfache Zugriffe auf Datenstrukturen in Funktionen eingeschaltet werden.
- Der Inliner kann nur arbeiten, wenn die Definition der Funktion zugreifbar ist, d.h. die Funktionsdefinition muß im aktuellen Modul liegen. Bei Verwendung von `static` Funktionen ist dies immer der Fall. Bei extern sichtbaren Funktionen kann der Inliner nur dann die meisten Aufrufe ersetzen, wenn die Definition in dem Modul liegt, wo auch die Aufrufe dieser Funktion stattfinden. D.h. bei starker Modularisierung kann ein Inliner nur dann wirken, wenn er modulübergreifend arbeitet, also durch einen Linker unterstützt wird.
- Probleme beim Inlining können dadurch entstehen, daß der Registerdruck stärker wird. Falls z.B. innerhalb von Schleifen Funktionsinlining durchgeführt wird, kann sich die Zahl der Registerkandidaten erhöhen.
- Durch Inlining kann auch ein ungünstigeres Cash-Verhalten hervorgerufen werden.

Aus obigen Gründen kann bei Nutzung des Inliners nicht immer von einer Performance-Verbesserung ausgegangen werden. Das Ergebnis ist stark von der Programmstruktur abhängig.

Verbesserte Sprungoptimierung

Mit der Option `-F` schalten Sie spezielle Optimierungsmaßnahmen ein und steuern damit die Codegenerierung. Zu `-F` können Sie optional mehrere Unteroptionen angeben. Neu ist die Unteroption `B`:

`-FB`

Sprünge werden optimiert und Sprungdistanzen werden verkürzt. Außerdem werden nicht erreichbare Codeteile eliminiert.

CES-Abweichungen vom ANSI-Standard

Die im Handbuch "CES, Leitfaden und Werkzeuge für die Programmierung mit C" [7] im Kapitel *C-Übersetzungssystem* im Abschnitt *CES-Abweichungen vom ANSI-Standard* beschriebenen Präprozessor-Anweisungen werden erweitert. Neu hinzu kommt die Präprozessor-Anweisung `#pragma weak`.

Präprozessor-Anweisung `#pragma weak`

Standard-Syntax:

```
#pragma weak identifizier[=identifizier2]
```

identifizier

wird als globales Symbol mit dem Binderattribut `weak` gekennzeichnet. Auch wenn der Binder (`ld`) keine Definition für *identifizier* findet, bindet er ein Modul oder ein Programm. Referenzen auf *identifizier* führen entweder zum entsprechenden Wert (falls definiert) oder zum Wert Null.

identifizier2

muß angegeben werden, wenn Bibliotheksfunktionen umdefiniert werden und der Binder keine Fehlermeldung erzeugen soll.

Beispiel

```
#pragma weak=_read
```

Die benutzerdefinierte Funktion `read()` überschreibt die Funktion `read()` der Standard-C-Bibliothek.

Installation von Fortran77-Komponenten

In der CES-Version 5.41B wird eine Entkoppelung von CES und F77 hinsichtlich gemeinsamer Komponenten (ulsmch, cc, cpp, oc23) vorgenommen. Deswegen arbeitet F77 V1.2A nach der Installation von CES V5.41B nicht mehr korrekt, weil F77 versucht, auf (noch) nicht existierende, separat installierte Komponenten zuzugreifen.

Um diese Situation zu bereinigen, enthält dieses CES-Package auch die von F77 benötigten Komponenten. Diese werden bei der normalen CES-Installation nicht installiert. Um diese Komponenten bei Bedarf zu installieren, gehen Sie wie folgt vor:

```
pkgadd -d gerät -s /var/spool/pkg Sices  
pkgadd -d /var/spool/pkg -r /var/spool/pkg/Sices/install/f77files Sices
```

(*gerät* kann z.B. /dev/fd0 sein). Voraussetzung für dieses Vorgehen ist allerdings, daß in der Datei /var/sadm/install/admin/default der Parameter `instance` auf `overwrite` gesetzt ist.

Post&Wait-Treiber (MX500)

Das Handbuch "Referenzhandbuch für Programmierer" [9] wird um den Abschnitt *Post&Wait-Treiber* ergänzt.

Der Post&Wait-Treiber wurde insbesondere für den Einsatz mit Datenbanken entwickelt. Er kann in einem Subsystem, das eine Reihe von Prozessen umfaßt, eingesetzt werden, wenn die Prozesse synchronisiert werden müssen und für die Synchronisierung von Timeout-Mechanismen erforderlich sind.

Ziel der Entwicklung ist es, den großen Aufwand mit Semaphoren in Verbindung mit Timing-Routinen zu vermeiden. In den Fällen, in denen gewöhnlich bis zu vier Systemaufrufe für die Synchronisierung mit Timeouts erforderlich waren, wird jetzt nur noch ein Systemaufruf verwendet.

Der neue Mechanismus kann nur in Subsystemen, die eine Reihe von Prozessen umfassen, eingesetzt werden, wenn die zusätzliche Verwaltung von Prozeßnummern subsystemintern bereitgestellt wird.

Post&Wait - Grundoperation

Symbolische Beschreibung der Post&Wait-Operationen:

```
status = PW_WAIT (pwid, timeout)
```

Auf den Prozeß mit der Nummer `pwid` wird maximal `timeout` Clock-Einheiten gewartet. Die Nummer `pwid` ist eine eindeutige interne Nummer für einen Einzelprozeß im Subsystem. Der Rückgabewert `status` gibt an, ob der Prozeß geweckt wurde oder ob der Timeout ablief. Gültige Werte sind `{0 == Prozeß wurde geweckt}` und `{PW_TIMED_OUT == Timeout für Prozeß abgelaufen}`.

```
PW_POST (pwid)
```

Wenn Sie einen Prozeß mit der Nummer `pwid` aufwecken (`posten`), werden alle Prozesse geweckt, die auf den Prozeß `pwid` warten.

Verwendet ein Prozeß die Operation `PW_WAIT(pwid,timeout)`, wartet er in Schlafstellung auf das Posting des Prozesses `pwid`. Ist die Operation `PW_POST(pwid)` vor Ablauf des Timeouts erfolgt, wird der Prozeß geweckt und die Operation `PW_WAIT()` meldet den Rückgabewert 0. Erfolgt innerhalb `timeout` Clock-Einheiten keine Operation `PW_POST(pwid)`, wird der Prozeß ebenfalls geweckt, aber die Operation `PW_WAIT()` meldet den Status `PW_TIMED_OUT`.

Wird die Operation `PW_WAIT()` auf einen Prozeß angewendet, der bereits geweckt ist, wird von `PW_WAIT()` sofort der Rückgabewert `Status = 0` gemeldet.

Wird ein Prozeß über `PW_POST(pwid)` geweckt, werden alle Prozesse geweckt, die auf den Prozeß `pwid` warten.

Implementierung von Post&Wait-Operationen mit einem Pseudogerät

Da es keine Möglichkeit gibt, Systemaufrufe problemlos in den UNIX-Kernel einzufügen, werden die Post&Wait-Operationen über ein Pseudogerät implementiert. Dieses Gerät (z.B. `/dev/pw/0`) wird von sämtlichen Prozessen des jeweiligen Subsystems, das die Post&Wait-Operationen verwendet, gemeinsam benutzt.

Ein weiteres Problem ist die Definition der Prozeßnummern im Subsystem der Prozesse. Aufgrund der Implementierungseinschränkungen für den Treiber muß die Nummer im Bereich `0...PW_MAX_PID` (max. 255) liegen. D.h. die Nummern müssen im jeweiligen Subsystem von einem Master-Prozeß verwaltet werden, der sämtliche Prozesse in diesem Subsystem kennt.

Die Post&Wait-Operationen selbst werden als `ioctl`-Aufrufe an den Treiber implementiert, nachdem das entsprechende Pseudogerät geöffnet wurde.

Post&Wait-Schnittstellen

```
#include <sys/pw.h>

char *pw_device = "/dev/pw/0";      /* Pseudogerät für dieses Subsystem */
                                     /* 0 bzw. dev/pw/n mit */
                                     /* n=0 bis PW_INSTANCES-1 */
int   pw_fd;                          /* Dateideskriptor */
int   pw_id;                          /* Nummer eines Prozesses im Subsystem */
```

Pseudogerät (z.B. `/dev/pw/0`) des Subsystems öffnen:

```
pw_fd = open (pw_device, O_RDWR);
```

Operation `PW_WAIT`:

```
status = ioctl (pw_fd, PW_WAIT, ((timeout<<8) | pw_id));
```

Rückgabewert:

```
status == 0           - Prozeß wurde geweckt
status == PW_TIMED_OUT - Timeout des Prozesses lief ab
```

Operation `PW_POST`:

```
ioctl (pw_fd, PW_POST, pw_id);
```

Bei ungültiger `pw_id` einer `ioctl()`-Operation wird von der Operation der Rückgabewert `(-1)` gemeldet, wobei `errno` auf `EINVAL` gesetzt wird.

Konfiguration des Post&Wait-Treibers

Der Pseudotreiber kann durch Änderung der Datei `space.c` in `/etc/conf/pack.d/pw` geändert werden. Folgende Parameter sind von der Konfiguration betroffen:

```
#define PW_INSTANCES
```

Die maximale Zahl von verschiedenen Subsystemen, die vom Treiber unterstützt werden. Jedem Subsystem wird ein spezieller Geräteknotten, der in `/dev/pw` eingerichtet ist, zugeordnet. Die Nummer des Subsystems ist eine Zahl, die mit 0 beginnt. Sie ist auch die Minorgerätenummer des Geräteknottens.

```
#define PW_MAX_PID
```

Die maximale Zahl von Prozeßnummern im jeweiligen Subsystem. Die Nummer darf nicht größer als 255 (Implementierungslimit) sein.

Beispiel

Ein Master-Prozeß erzeugt über `master_pwid=255` 5 Slave-Prozesse. Die `slave_pwid` wird an den Slave weitergeleitet.

```
int pw_fd;                /* shared between master and slaves */
int master_pwid = 255;    /* global known by master and slaves */
int my_proc_table[5];     /* private table to map pw_id's to process id's */

/* open the communication pseudo device */
pw_fd = open("/dev/pw/0",0_RDWR);
/* create slave processes */
for (i=0; i < 5; i++)
{
    pid_t pid;
    switch (pid=fork())
    {
        case -1: /* error */
            report_error("fork failed");
            exit(1);
            break;
        case 0: /* child process, slave_pwid = i */
            slave_task(i);
            /* no return */
            break;
        default: /* father */
            /* remember the process id */
            my_proc_table[i] = pid;
            break;
    }
}
master_task();
```

Synchronisierung von Master-Prozeß und einem einzelnen Slave

```

master_task()
{
    int status;

    /* do some communication with a slave given by its slave_id */
    ...
    /* synchronize with slave, grant a timeout of 5 seconds */
    status = ioctl(pw_fd,PW_WAIT,((5*HZ)<<8|slave_id);
    switch (status)
    {
    case -1:          /* error */
        if (errno == EINVAL)
            report_error("PW_WAIT: invalid slave id %d",slave_id);
        else
            report_error("PW_WAIT: error %d",errno);
        /* do some recovery */
        break;
    case 0:          /* slave process was posted */
        /* do some action, continue normal processing */
        break;
    case PW_TIMED_OUT: /* timeout expired */
        report_error("PW_WAIT: slave %d timed out%d",slave=id);
        /* do recovery actions, restart operation on slave */
        break;
    default:
        report_error("PW_WAIT: invalid response %d",status);
        exit(1);
    }
}

slave_task(my_id)
int my_id;
{
    /* do some communication with the master */
    ...
    /* do the job the master wants us to do */
    ...
    /* signal the master that our job is complete */
    (void)ioctl(pw_fd,PW_POST,my_id);
}

```

Zusätzliche Funktion des Post&Wait-Pseudotreibers

Abbildung der time-Variablen des Kernels im Adreßraum des Benutzers

Der Post&Wait-Pseudotreiber stellt die zusätzliche Schnittstelle `mmap()` zur Verfügung, so daß die Benutzerprozesse die interne `time`-Variable des Kernels im eigenen Adreßraum abbilden können. Dadurch erhält man auf einfache Weise sekundengenau eindeutige Zeitstempel innerhalb des Anwendungsprogramms.

Treiberschnittstelle mmap():

```
#include <sys/types.h>
#include <sys/mman.h>
#include <sys/pw.h>

char    *pw_device = "/dev/pw/0"; /* 0 bzw. dev/pw/n mit */
                                     /* n=0 bis PW_INSTANCES-1 */
int     pw_fd;
caddr_t addr;
off_t   page_offset;
```

Pseudogerät des Subsystems öffnen:

```
pw_fd = open(pw_device,O_RDONLY);
```

Kernelseite mit der time-Variable abbilden:

```
addr = mmap (NULL, PAGE_SIZE, PROT_READ, MAP_SHARED, pw_fd, 0);
```

Relativzeiger der time-Variable innerhalb der abgebildeten Seite positionieren:

```
ioctl (pw_fd, PW_TIME_OFFSET, &page_offset);
```

Treiberfunktion mmap():

```
int     pw_device = "/dev/pw/0"; /* Name des Pseudogeräts */
int     pw_fd;
caddr_t addr;
int     time_pg_offset; /* Relativzeiger der time-Variablen
                           innerhalb einer Seite */
long    *time_ptr;
long    time_stamp;
```

Pseudogerät öffnen:

```
pw_fd = open(pw_device,O_RDONLY);
```

Seite mit der time-Variable abbilden:

```
addr = mmap(NULL, PAGE_SIZE, PROT_READ, MAP_SHARED, pw_fd, 0);
```

Relativzeiger der time-Variable innerhalb der abgebildeten Seite positionieren:

```
ioctl(pw_fd, PW_TIME_OFFSET; &time_pg_offset);
```

Zeiger auf die time-Variable setzen:

```
time_ptr = (long *) (addr + time_pg_offset);
```

Die time-Variable wird jetzt einfach durch Ablesen des Zeigerinhalts gelesen:

```
time_stamp = *time_ptr;
```

Asynchronous Raw I/O (MX500)

This document describes the usage and the implementation of asynchronous raw I/O for the MX500.

The user interface is based on the description of the user interface in the "Asynchronous I/O Reference Manual" [10] for the RM600 and RM400. Asynchronous I/O is done via `ioctl()` commands.

Introduction

With asynchronous raw I/O (AIO in the following) a process can start an I/O operation and instead of blocking until the I/O completes, the process can continue executing. Later the process can request the state of previously sent I/O requests. It is also possible that the process might be notified by a signal, when a previous I/O request is completed.

The following `ioctl` commands can be used:

- `DKIOCAIOVERS` : Return the version number of the AIO interface.
- `DKIOCASTRT` : Start an asynchronous I/O request.
- `DKIOCASTAT` : Return the state of previous requests.

The following (new) flag can be used with `fcntl(2)`:

- `FRAIOSIG` : Send a signal to the process when requests are completed.

A maximum number of 500 simultaneous asynchronous I/O (AIO) requests is possible. If the maximum number of simultaneous AIO exceeds 500, `EAGAIN` is sent. Another AIO request will not be accepted, unless an AIO request in the buffer has been terminated and its status has been checked by `DKIOCASTAT`. When the device is not closed and the status is not checked, the AIO is temporarily locked.

The ioctl Commands

The ioctl Command DKIOCAIOVERS:

Determine the version of the AIO interface. Example:

```
#include <sys/async.h>
int fd, version;
fd = open(DEVICE, O_RDONLY);
if (ioctl(fd, DKIOCAIOVERS, &version) < 0 ||
    version != AIOVERSION) {
    perror("AIO not supported or mismatched version");
    exit(2);
}
```

The ioctl Command DKIOCASTRT:

Start an asynchronous request. A variable of type struct areqbuf should be initialized:

```
typedef struct areqbuf {
    long au_cmd;        /* AU_READ, AU_WRITE, AU_ORDWRITE */
    long au_daddr;     /* location on disk */
    char *au_maddr;    /* memory address of buffer */
    long au_size;      /* number of bytes to transfer */
    char *au_ref;      /* caller specified reference */
} AREQBUF;
```

The au_cmd's are defined as

```
#define AU_READ      01 /* read request */
#define AU_WRITE     02 /* unordered write request */
#define AU_ORDWRITE  04 /* ordered write request */
#define AU_CMDMASK  07 /* mask of command bits */
```

In the present version an ordered write request is treated as an unordered write request.

The caller specified reference value may be any value which can be stored in char *. The kernel does not use this value, it simply returns this value in the asyncstatus data structure where it can be used to identify the returned request.

The error EIORESID (defined in async.h) is returned and no bytes are read or written if one of the following is false:

- The location on disk must be specified in DEVBSIZE units.
- The memory address must be on a 16 byte boundary.
- The number of bytes to transfer must be a multiple of DEVBSIZE.
- The number of bytes to transfer must be less or equal to 32 * 1024.

The ioctl Command DKIOCASTAT:

To retrieve the state of up to MAXSTATUS requests the DKIOCASTAT command is used:

```
struct asyncstatus iostat;
...
ioctl(fd, DKIOCASTAT, &iostat);
```

The variable `iostat` will be filled out by the AIO driver and consists of the number of request returned and a data structure `struct aiostat` for each completed request but not more than MAXSTATUS. If there are no completed requests, the number of completed requests is 0.

```
#define MAXSTATUS      15
typedef struct asyncstatus
{
    long    account;          /* # of requests being returned */
    IOSTAT  astatus[MAXSTATUS]; /* completion status per request */
} ASYNCSTATUS;
```

The structure `IOSTAT` is defined as:

```
typedef struct aiostat
{
    short iostatus; /* I/O completion status for request */
    short iobsize; /* verification (from AREQBUF dbsize) */
    char *iomaddr; /* verification (from AREQBUF dbmaddr) */
    char *ioref;   /* return caller specified reference */
} IOSTAT;
```

The `ioref` value is always identical with the value `au_ref` in `struct areqbuf`. This may be any value which can be stored in `char *`.

Errors

All errors except `EIORESID` are the same as returned via the normal I/O system calls as there are `read`, `write`, `ioctl`. Since `EIORESID` is specific to asynchronous I/O it is defined in `sync/async.h` rather than in `errno.h`. See the `ioctl` command `DKIOCASTRT` when `EIORESID` is returned.

Performance Results

In the following the transfer rates of asynchronous raw disk I/O are compared with synchronous raw disk I/O. In the 1st example 2 x 1MB was read from the same or a different raw device. The program for this is:

```
for (i = 0; i < 32; i++) {
  read(fd0, buf0[i], 32KB) or ioctl(fd0, DKIOCASTRT, reqbuf0[i]);
  read(fd1, buf1[i], 32KB) or ioctl(fd1, DKIOCASTRT, reqbuf1[i]);
}
```

---> *1st time after read measured here*

```
close(fd0);
```

```
close(fd1);
```

---> *2nd time after close measured here*

In the synchronous case 1st and 2nd time are almost the same, as `close()` immediately returns. In the asynchronous case `close()` has to wait, till all reads are done. Timing was done with the function `gettimeofday()`.

Synchronous Read

1 disk	: Time after Read	: 1.85 sec
	: Time after Close	: 1.85 sec
2 disks	: Time after Read	: 1.35 sec
	: Time after Close	: 1.35 sec

Asynchronous Read

1 disk	: After read	: 0.45 sec
	: After Close	: 1.45 sec
2 disks	: After read	: 0.45 sec
	: After Close	: 1.00 sec

Two Examples for demonstrating asynchronous I/O**Example 1:**

```

/*
** Asynchronous IO:
** read 32KB synchron + asynchron, close device, and compare
*/
#include <stdio.h>
#include <sys/fcntl.h>
#include <sys/errno.h>
#include "async.h"

#define COMMAND      AU_READ
#define SIZE         (32 * 1024)
#define DADDR       0;
#define DEVICE1      "/dev/rdskc0d0s6"

AREQBUF      reqbuf[2];      /* request buffer */
ASYNCSTATUS  iostat;        /* status buffer */
char         *buf1, *buf2;   /* transfer buffer */
extern int    errno;

main(argc, argv)
char **argv;
{
    int    i, fd1;
    long   version;
    AREQBUF *req;
    IOSTAT *reqstat;

    AIO_MALLOCC(buf1, SIZE);
    AIO_MALLOCC(buf2, SIZE);

    if ((fd1 = open(DEVICE1, O_RDONLY)) == -1) {
        perror(DEVICE1);
        exit(1);
    }
    read(fd1, buf2, SIZE);

    /* Determine if AIO is supported for DEVICE */

    if (ioctl(fd1, DKIOCAIOVERS, &version) < 0 || version != AIOVERSION) {
        perror("AIO not supported or mismatched version\n");
        exit(2);
    }

    /* Setup the request */

    reqbuf[0].au_cmd      = COMMAND;
    reqbuf[0].au_daddr    = DADDR;
    reqbuf[0].au_maddr    = &buf1[0];
    reqbuf[0].au_size     = SIZE;
    reqbuf[0].au_ref      = (char *)&reqbuf[0];

    if (ioctl(fd1, DKIOCASTRT, &reqbuf[0]) < 0) {
        perror("ioctl failed on DKIOCASTRT"); /* read async */
        exit(3);
    }
}

```

```
    }
    close(fd1);
    for (i = 0; i < SIZE; i++) {
        if (buf1[i] != buf2[i]) {
            printf("AIO: read 32KB mismatch\\n");
            exit(4);
        }
    }
    printf("AIO: successfully read 32KB from %s\\n", DEVICE1);
}
```

Example 2:

```

/*
** Asynchronous IO:
** read 32KB synchron + asynchron, poll and compare
*/
#include <stdio.h>
#include <sys/fcntl.h>
#include <sys/errno.h>
#include "async.h"

#define COMMAND      AU_READ
#define DADDR        0
#define SIZE          (32 * 1024)          /* Disk Block / Page Size */
#define DEVICE        "/dev/rdisk/c0d0s6"

AREQBUF      reqbuf[2];      /* request buffer */
ASYNCSTATUS  iostat;        /* status buffer */
char         *buffer;        /* transfer buffer */
char         *busync;        /* transfer buffer */
extern int    errno;

main()
{
    int    i, fd, ncomplete;
    long   version;
    AREQBUF *req;
    IOSTAT *reqstat;

    AIO_MALLOCC(busync, 2 * SIZE);
    AIO_MALLOCC(buffer, 2 * SIZE);

    if ((fd = open(DEVICE, O_RDONLY)) == -1) {
        perror(DEVICE);
        exit(1);
    }
    read(fd, busync, 2 * SIZE);

    /* Determine if AIO is supported for DEVICE */

    if (ioctl(fd, DKIOCAIOVERS, &version) < 0 || version != AIOVERSION) {
        perror("AIO not supported or mismatched version\\n");
        exit(2);
    }

    /* Setup the request */

    reqbuf[0].au_cmd      = COMMAND;
    reqbuf[0].au_daddr    = DADDR;
    reqbuf[0].au_maddr    = buffer;
    reqbuf[0].au_size     = SIZE;
    reqbuf[0].au_ref      = (char *)&reqbuf[0];

    reqbuf[1].au_cmd      = COMMAND;
    reqbuf[1].au_daddr    = DADDR + SIZE/512;
    reqbuf[1].au_maddr    = buffer + SIZE;
    reqbuf[1].au_size     = SIZE;
    reqbuf[1].au_ref      = (char *)&reqbuf[1];

```

```

    /* start to read */
    for (i = 0; i < 2; i++) {
        if (ioctl(fd, DKIOCASTRT, &reqbuf[i]) < 0) {
            perror("ioctl failed on DKIOCASTRT");
            exit(3);
        }
    }
    ncomplete = 0;
marke:
    iostat.acount = 0;

    while (iostat.acount == 0) {
        sleep(1);
        if (ioctl(fd, DKIOCASTAT, &iostat) < 0) {
            perror("ioctl failed on DKIOCASTAT");
            exit(4);
        }
    }
    reqstat = &iostat.astatus[0];
    while (iostat.acount-- > 0) {
        req = (AREQBUF *)reqstat->ioref;

        if (reqstat->iobsize != req->au_size)
            printf("FAILURE: reqstat->iobsize = %d, reqbuf.au_size = %d\n",
                   reqstat->iobsize, req->au_size);
        else if (reqstat->iomaddr != req->au_maddr)
            printf("FAILURE: reqstat->iomaddr = %x, reqbuf.au_maddr = %x\n",
                   reqstat->iomaddr, req->au_maddr);
        else if (reqstat->iostatus)
            printf("FAILURE: reqstat->iostatus = %x\n",
                   reqstat->iostatus);
        else
            printf("AIO: Sucessfully returned requests\n");
        ncomplete++;
        reqstat++;
    }

    if (ncomplete < 2)
        goto marke;

    close(fd);

    for (i = 0; i < (2 * SIZE); i++) {
        if (buffer[i] != busync[i]) {
            printf("AIO: read 32KB mismatch\n");
            exit(4);
        }
    }
    printf("AIO: Sucessfully read 2 * 32KB from %s\n", DEVICE);
}

```


Manual Pages

In diesem Kapitel sind Ergänzungen und Änderungen zu Manual Pages beschrieben. Sie finden diese Informationen in der Reihenfolge: Benutzerkommandos, Netzkommandos, Systemverwalter- und Netzverwalter-Kommandos, Gerätetreiber-Dateien.

In den Handbüchern "Kommandos, Band 1, A - K" [12] und "Kommandos, Band 2, L - Z" [13] sind betroffen:

- das Kommando `acctcom`
- das Kommando `ar`
- das Kommando `awk`
- das Kommando `bc`
- das Kommando `cal`
- das Kommando `copy`
- das Kommando `cpio`
- das Kommando `crypt`
- das Kommando `csh`
- das Kommando `df`
- das Kommando `diff`
- das Kommando `diff3`
- das Kommando `dirname`
- das Kommando `flchk`
- das Kommando `gettxt`
- das Kommando `ksh`
- das Kommando `login`
- das Kommando `lp`
- das Kommando `lpr`
- das Kommando `lpstat`
- das Kommando `ls`
- das Kommando `mail`
- das Kommando `mailx`
- das Kommando `mt`
- das Kommando `mv`
- das Kommando `newgrp`
- das Kommando `passwd`
- das Kommando `pr`
- das Kommando `ps`
- das Kommando `rm`

- das Kommando `sar`
- das Kommando `secure`
- das Kommando `sed`
- das Kommando `sh`
- das Kommando `shl`
- das Kommando `sort`
- das Kommando `tail`
- das Kommando `talk`
- das Kommando `tar`
- das Kommando `time`
- das Kommando `timex`
- das Kommando `uniq`
- das Kommando `vi`
- das Kommando `wcheck`
- das Kommando `wc`
- das Kommando `who`

Im Handbuch "Referenzhandbuch für Systemverwalter" [11] sind folgende Kommandos bzw. Gerätetreiber-Dateien betroffen:

- das Kommando `addbad(1M)` (MX500)
- das Kommando `bind(1M)` (MX500)
- das Kommando `boot(1M)` (MX300)
- das Kommando `colltbl(1M)`
- das Kommando `edvtoc(1M)`
- das Kommando `swap(1M)`
- das Kommando `smtper(1M)`
- die Gerätetreiber-Datei `exa(7)`
- die Gerätetreiber-Datei `shd(7)`

Im Handbuch "Netzwerke - Leitfaden für Benutzer und Verwalter" [3] sind betroffen:

- das Kommando `tcpdump(1)`
- das Kommando `gated(1M)`
- das Kommando `traceroute(1M)`

acctcom - Suchen und Drucken von Prozeßabrechnungsstatistiken

Für acctcom gelten folgende Änderungen und Ergänzungen:

1. Text des Ausgabefeldes COMMAND NAME geändert

COMMAND NAME

Name des ausgeführten Kommandos, zu dem die folgende Prozeßinformation gehört. Ein Nummernzeichen # vor dem Kommandonamen bedeutet, daß das Kommando mit Systemverwalter-Privilegien ausgeführt wurde.

2. Spaltentitel der Ausgabefelder REAL (SEC), CPU (SEC) geändert

Die Spaltentitel, die Sie mit dem Kommando acctcom erhalten, haben folgendes Aussehen:

COMMAND NAME	USER	TTYNAME	START TIME	END TIME	REAL (SECS)	CPU (SECS)	MEAN SIZE(K)
-----------------	------	---------	---------------	-------------	----------------	---------------	-----------------

3. Ausgabefelder CPU SYS, (SECS) USER neu

Wenn Sie das Kommando acctcom mit der Option -t angeben, erhalten Sie zwei neue Ausgabefelder:

CPU SYS)

CPU-Zeit, die das System verbraucht hat.

(SECS) USER

CPU-Zeit, die der Benutzer verbraucht hat.

ar - Bibliotheken verwalten

Für ar ergeben sich folgende Änderungen bezüglich der internationalen Umgebung:

Die Umgebungsvariable *LC_TIME* bestimmt das Format der Datums- und Zeitangaben bei der Auflistung des Archivinhalts mit der Option v.

Wenn *LC_TIME* nicht oder als leere Zeichenkette definiert ist, wird der Wert von *LANG* als Standardwert für die jeweils nicht gesetzte oder leere Variable herangezogen. Ist auch *LANG* nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert.

Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

awk - Programmierbare Bearbeitung von Textdateien

Beim `awk` sind folgende Änderungen zu beachten:

1. Besonderheit bei Feldtrennzeichen `t`

Wenn Sie als Trennzeichen zwischen den Feldern eines Eingabesatzes das Zeichen `t` verwenden wollen, so müssen Sie es beim `awk`-Aufruf oder im `BEGIN`-Teil des `awk`-Programms folgendermaßen angeben:

```
awk -F"[t]" ...           oder           BEGIN {FS="t" ...}
```

2. Internationalisierung

In geklammerten regulären Ausdrücken bestimmt die Umgebungsvariable `LC_COLLATE` die Bedeutung von Zeichenbereichen, Äquivalenzklassen und Zeicheneinheiten, die Umgebungsvariable `LC_CTYPE` die Bedeutung von Zeichenklassen. `LC_COLLATE` bestimmt das Verhalten von Vergleichsoperatoren beim Vergleich von Zeichenketten.

Wenn `LC_COLLATE` oder `LC_CTYPE` nicht oder als leere Zeichenkette definiert ist, wird der Wert von `LANG` als Standardwert für die jeweils nicht gesetzte oder leere Variable herangezogen. Ist auch `LANG` nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert. Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

bc - Arithmetische Sprache

Bei bc sind folgende Änderungen und Ergänzungen zu beachten:

1. Verwendungsbereich für Operator % erweitert

% Rest bei ganzzahliger Division (Modulo), kann auch für Gleitkommazahlen verwendet werden.

2. Angabe der x -Werte bei mathematischen Funktionen ergänzt

Die Werte für x bei den mathematischen Funktionen müssen im absoluten Bogenmaß angegeben werden.

3. Abschnitt *Nachkommastellen* erweitert

Jedem Ausdruck E in bc ist eine Zahl von Nachkommastellen zugeordnet. Diese Zahl können Sie mit der Variablen *scale* ändern und abfragen oder mit der Funktion *scale(E)* abfragen.

Wenn Sie zwei Ausdrücke durch einen Operator verknüpfen, so wird die dem Ergebnis zugeordnete Anzahl von Nachkommastellen durch eine Regel bestimmt, die für den Operator spezifisch ist. Im folgenden werden die Regeln für die bc-Operatoren beschrieben. Dabei werden einige Abkürzungen verwendet:

a = erster Operand

b = zweiter Operand

R = Anzahl der Nachkommastellen des Ergebnisses einer Rechenoperation

A = *scale(a)*

B = *scale(b)*

$-, ++, --$

Das unäre Minuszeichen $-$ sowie die Inkrement- und Dekrement-Operatoren $++$ und $--$ (in Präfix- und Postfix-Notation) ändern die Zahl der Nachkommastellen nicht.

Regel: $scale(E) = scale(-E) = scale(--E) = scale(++E) \dots$

$+, -$

Für die binären Operatoren $+$ und $-$ ist R gleich der Anzahl der Stellen des Operanden, der die meisten Nachkommastellen hat. Dabei spielt es keine Rolle, ob Sie *scale* zuvor einen anderen Wert zugewiesen haben.

Regel: $R = \max(A, B)$

*

Bei einer Multiplikation spielt ein zuvor gesetzter Wert von *scale* eine Rolle: bc ermittelt zunächst den höchsten Wert *max* aus den Werten von *scale*, *A* und *B*. Anschließend bildet bc die Summe von *A* und *B*. *R* ist dann der kleinere Wert aus dem Vergleich von *max* und dieser Summe.

Regel: $R = \min(A+B, \max(\text{scale}, A, B))$

/

Bei einer Division bestimmt ausschließlich der Wert von *scale* die Genauigkeit des Ergebnisses:

Regel: $R = \text{scale}$

^

Beim Potenzieren berechnet sich *R* wie folgt:

– wenn der ganzzahlige Exponent $e \geq 0$ ist:

bc ermittelt den höchsten Wert *max* aus den Werten von *scale* und *A*. Anschließend multipliziert es *A* mit dem Betrag *m* des Exponenten, vergleicht das Ergebnis mit *max* und nimmt von beiden den kleineren Wert.

Regel: $R = \min(A^m, \max(\text{scale}, A))$

– wenn der ganzzahlige Exponent $e < 0$ ist:

Die Anzahl Nachkommastellen des Ergebnisses entspricht dem Wert von *scale*.

Regel: $R = \text{scale}$

=, =@

Bei den Zuweisungsoperatoren entspricht der Wert von *R* demjenigen von *A* nach der Zuweisung. Dabei gilt für einen zusammengesetzten Operatoren =@ jeweils die gleiche Regel zur Berechnung der Anzahl Nachkommastellen, wie für den einfachen Operatoren @.

Regeln: $R = \text{scale}(b)$
 $R = \text{scale}(a@b)$

%

Ist bei einer %-Berechnung der Wert von *scale* ungleich 0, so errechnet sich das Ergebnis folgendermaßen:

$a\%b = a - (a/b) * b$

Dabei wird zunächst die Division mit der Genauigkeit von *scale* berechnet. Die Genauigkeit der Multiplikation mit *b* beträgt:

`scale + B`

Das heißt, daß die Multiplikation exakt ausgeführt wird. Der *z*-Operator kann hier also als Maß für die Genauigkeit verwendet werden, mit der die Division durchgeführt wird.

R schließlich ist der höchste Wert aus *A* und dem Additionsergebnis von *scale* und *B*.

Regel: $R = \max ((\text{scale} + B), A)$

cal - Kalender ausgeben

Bei `cal` wurde bei der Ausgabe für September bei der Jahresangabe 1752 eingefügt.

Wenn Sie `cal` mit dem Argument 1752 oder den Argumenten `9 1752` aufrufen, erhalten Sie für den Monat September eine verkürzte Ausgabe. Im September 1752 gab es eine Zeitkorrektur von 11 Tagen, die von `cal` berücksichtigt wird.

copy - Dateien gruppenweise kopieren

Bei copy wurde die Option `-o` korrigiert.

Wenn dies Option gesetzt ist, werden Eigentümer und Gruppenzugehörigkeit der Quelldatei übernommen. Um auch die letzte Änderungs- und Zugriffszeit zu übernehmen, müssen Sie zusätzlich `-m` angeben.

cpio - Dateien und Dateiverzeichnisse ein- und auslagern

Für `cpio` gibt es folgende Neuerungen:

1. Option `-k` zum Sichern von Daten auf Folgebändern

Wenn Sie große Datenmengen sichern wollen, die nicht alle auf ein Band passen, können Sie mit der Option `-k` Folgebänder anfordern. Die Option `-k` müssen Sie zusammen mit der Option `-c` in folgender Form angeben:

`-k bandlänge -c 1024*n`

bandlänge bezeichnet die Länge des zu verwendenden Bandes in Kbyte abzüglich eines Sicherheitspuffers. Wenn Sie z.B. ein Band mit einer Länge von 155 Kbyte verwenden wollen, geben Sie für *bandlänge* 150 an. Wieviel Sicherheitspuffer Sie benötigen, hängt von Ihren Erfahrungswerten mit den Bändern ab.

Mit der Option `-c` geben Sie 1024 oder ein Vielfaches (*n*) von 1024 an.

Ist das Band mit *bandlänge* Kbyte beschrieben, werden Sie aufgefordert ein neues Band einzulegen und RETURN zu drücken.

2. Sichern von Dateien mit I-Nodes, die größer als 65535 sind

Aus Gründen der Kompatibilität zu SYSTEM V Release 3.2 oder früher werden Dateien mit I-Nodes, die größer als 65535 sind, standardmäßig nicht mit `cpio` gesichert. Wenn Sie solche Dateien sichern wollen, so geben Sie die Zusatzoption `c` an.

3. Internationalisierung

Die Umgebungsvariable `LC_TIME` bestimmt das Format der Datums- und Zeitausgabe bei der Auflistung des Archivinhalts mit der Option `-v`.

Wenn `LC_TIME` nicht oder als leere Zeichenkette definiert ist, wird der Wert von `LANG` als Standardwert herangezogen. Ist auch `LANG` nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert. Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

crypt - Text verschlüsseln und entschlüsseln

Für `crypt` gibt es die neue Option `-k` und die neue Variable `CrYpTkEy`.

Statt `-c[_schlüssel]` können Sie auch die Option `-k` angeben. `-k` en- und dekodiert den angegebenen Text durch den Wert der Variablen `CrYpTkEy`. Die Klein- und Großschreibung des Variablennamens ist wie hier angegeben obligatorisch.

Beispiel

```
$ CrYpTkEy=flattix; export CrYpTkEy
$ crypt -k < datei > datei.secr
```

csh - C-Shell

Für csh wurde das Kommando `setenv` korrigiert.

Die Argumentbeschreibung zu `VAR_wort` muß folgendermaßen beginnen: `setenv VAR wort` weist der Umgebungsvariablen `VAR` den Wert `wort` zu.

df - Anzahl der freien und belegten Plattenblöcke und I-Nodes ausgeben

Bei `df` sind folgende Optionen ergänzt bzw. geändert worden:

- `-n`
Diese Option kann nicht mit `s5_option`, `-g`, `-k`, `-v` und `-t` kombiniert werden. Kombinieren Sie `-n` mit `-o` oder `-f`, erhalten Sie eine Fehlermeldung.
- `-e`
Die Anzahl der freien I-Nodes wird ausgegeben. Diese Option kann nicht mit `-o` benutzt werden und setzt die Option `-l` außer Kraft.
- `-g`
Diese Option kann nicht mit `s5_option` oder mit `-o` benutzt werden und setzt die Optionen `-b`, `-e`, `-k`, `-l`, `-n`, `-r` und `-v` außer Kraft.
- `-k`
Diese Option kann nicht mit `s5_option` oder mit `-o` benutzt werden und setzt die Optionen `-b`, `-e`, `-k`, `-l`, `-n`, `-r` und `-v` außer Kraft.
- `-l`
Diese Option gibt für alle lokal montierten Dateisysteme die Anzahl freier Blöcke und I-Nodes aus. `-l` ist die voreingestellte Option. Sie kann nicht mit `s5_option`, `-e` oder `-o` benutzt werden.
- `-t`
Diese Option gibt pro montiertem oder angegebenem Dateisystem die Anzahl freier Blöcke und Dateien sowie die Gesamtanzahl der zur Verfügung stehenden Blöcke und Dateien aus. `-t` setzt die Option `-b` außer Kraft. Die Kombination von `-t` mit `-o i` ergibt eine Total-Zeile aller `-o i` Ausgaben.
- `-v`
Diese Option kann nicht mit `-o` benutzt werden und setzt die Optionen `-b`, `-e`, `-k`, `-l`, `-n`, `-r` und `-v` außer Kraft.
- `-o`
Diese Option ist sinnvoll nur mit den Optionen `-F` und `-t` kombinierbar.

diff - Dateien zeilenweise vergleichen

Die Optionen `-h` und `-i` sollten nicht gleichzeitig angegeben werden.

diff3 - Drei Dateien zeilenweise vergleichen

`diff3` kann auch Dateien bearbeiten, die größer sind als 64 Kbyte. `diff3` findet jedoch maximal 200 Unterschiede.

dirname - Pfad-Präfix vom Dateinamen trennen

Enthält bei `dirname` die Angabe *zeichenkette* keinen Schrägstrich, so wird nur ein Punkt auf die Standard-Ausgabe ausgegeben.

flchk - Labelbereich einer Diskette überprüfen (floppy check)

Bei flchk gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable *LANG* bestimmt das Eingabeformat für die Antwort auf Ja/Nein-Abfragen in der internationalen Umgebung.

Die Antworten auf Ja/Nein-Abfragen müssen dabei immer in der durch *LANG* festgelegten Sprache eingegeben werden, auch dann, wenn die Abfragen vom System in einer anderen Sprache gestellt werden. Ist z.B. kein Meldungskatalog für flchk vorhanden und hat *LANG* den Wert *De*, wird die Abfrage in englisch ausgegeben, die Antwort muß dagegen in deutsch eingegeben werden.

Wenn *LANG* nicht oder als leere Zeichenkette definiert ist, verhält sich das System so, als wäre es nicht internationalisiert.

Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich System so, als wäre keine Variable gesetzt.

gettxt - Zeichenketten in einer Datenbasis für Meldungstexte suchen

Bei gettxt wurde das Argument "standard-text" korrigiert.

Wenn gettxt die gesuchte Zeichenkette nicht in der Meldungsdatei finden kann, deren Pfadname durch die Variable *LC_MESSAGES* festgelegt ist, sucht gettxt in der Datei */usr/lib/locale/C/LC_MESSAGES/meldungsdatei* nach der *meldungsnummer* zugeordneten Zeichenkette. Wenn entweder *meldungsdatei* oder *meldungsnummer* nicht existiert, gibt gettxt den angegebenen *standard-text* aus.

ksh - Korn-Shell

Zu ksh gibt es folgende Korrekturen und Neuerungen:

1. Kommando `typeset`: Leerzeichen zwischen Option und Argument gestrichen

Bei folgenden Optionen des `ksh`-Kommandos `typeset` wird kein Leerzeichen zwischen Option und zugehörigem Argument angegeben:

```
-L[zahl], -R[zahl], -Z[zahl], -i[zahl]
```

2. Führendes Nummernzeichen `#` bei Parametern gestrichen

Bei folgenden Parametern wird innerhalb der geschweiften Klammern kein führendes Nummernzeichen `#` angegeben:

```
#{parameter:-wort}  
#{parameter:=wort}  
#{parameter:?wort}  
#{parameter:+wort}  
#{parameter#muster}  
#{parameter##muster}  
#{parameter%muster}  
#{parameter%%muster}
```

3. Ein-/Ausgabeumlenkung bei bidirektionalen Pipelines ergänzt

Sie können aus einem Skript Kommandos oder Pipelines im Hintergrund starten. Diese können dann mit Ihrem Programm kommunizieren. Um einen solchen Ko-Prozeß zu starten, stellen Sie dem Kommando den Operator `|&` nach. Verwenden Sie bidirektionale Pipes nur in Skripts, jedoch nicht auf der Kommandozeile.

Kommandofolgen lassen sich nur einmal als bidirektionale Pipes aufrufen. Wenn Sie den ursprünglichen Prozeß abgebrochen haben (z.B. mit `kill-9 PID`) und zu einem späteren Zeitpunkt nochmals versuchen, ein Kommando in eine bidirektionale Pipe zu schreiben, wird zwar eine Subshell aufgerufen, aber der Prozeß gestoppt. Sie erhalten die Fehlermeldung `ksh: bad file unit number.`

4. Suchanweisung `G` korrigiert

Wenn Sie bei der Suchanweisung `G` kein Argument *zahl* angeben, wird standardmäßig auf das älteste eingegebene Kommando zugegriffen.

login - Sich neu am System anmelden

Beim `login` gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable *LANG* bestimmt die Sprache der Meldungstexte.

Wenn *LANG* nicht oder als leere Zeichenkette definiert ist, verhält sich das System so, als wäre es nicht internationalisiert.

Es wird empfohlen, für Kennungen und Kennwörter nur Zeichen aus dem für portable Dateinamen vereinbarten Zeichensatz (Portable Filename Character Set) zu verwenden; 8-Bit-Dateien sind unter Umständen nicht portabel zu nicht internationalisierten Systemen.

lp - Dateien ausdrucken

lpr - Dateien ausdrucken und Druckaufträge steuern

lpstat - Informationen über Druckaufträge ausgeben

Bei den Kommandos `lp`, `lpr` und `lpstat` gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable `LANG` bestimmt die Sprache der Meldungstexte.

Wenn `LANG` nicht oder als leere Zeichenkette definiert ist, verhält sich das System so, als wäre es nicht internationalisiert.

ls - Informationen über Dateiverzeichnisse und Dateien ausgeben

Beim Kommando `ls` gibt es folgende Neuerungen:

1. Korrektur der Option `-L`

Bei symbolischen Verweisen wird statt des Namens der Ursprungsdatei oder des Ursprungsverzeichnisses der Verweisname ausgegeben.

2. Internationalisierung

Die Umgebungsvariable `LC_COLLATE` bestimmt die Sortierreihenfolge der Ausgabe von `ls`, die Umgebungsvariable `LC_CTYPE` bestimmt, welche Zeichen im Zusammenhang mit der Option `-q` als nicht druckbare Zeichen definiert werden, und die Umgebungsvariable `LC_TIME` bestimmt das Format und den Inhalt der Datums- und Zeitausgabe bei den Optionen `-g`, `-l`, `-n` und `-o`.

Wenn `LC_COLLATE`, `LC_CTYPE` oder `LC_TIME` nicht oder als leere Zeichenkette definiert ist, wird der Wert von `LANG` als Standardwert für die jeweils nicht gesetzte oder leere Variable herangezogen. Ist auch `LANG` nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert. Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

mail - Nachrichten senden oder lesen

Zu `mail` gibt es folgende Neuerungen:

1. Eingabe-Umlenkung im Sendemodus (Punkt als Textabschluß) ergänzt

Liest `mail` aus einer Datei, dann wird ein alleinstehender Punkt am Anfang einer Zeile nicht als Ende-Zeichen erkannt.

2. Entwertung des Zeichens `@` bei Netzadressierung gestrichen

Bei der domänen-orientierten Adressierung muß das Zeichen `@` nicht durch Gegenschrägstrich entwertet werden.

mailx - Nachrichten interaktiv bearbeiten

Bei `mailx` muß das Zeichen `@` innerhalb der Netzadressen im 2. Beispiel nicht mit einem Gegenschrägstrich entwertet werden.

mt - Magnetband oder Magnetbandkassette bearbeiten

Bei mt wird die Abfrage von Statusinformationen über *status* nicht mehr unterstützt.

mv - Dateien versetzen oder umbenennen

In der Beschreibung von `mv` wurde der Abschnitt *Format 2: Dateien ein anderes Dateiverzeichnis versetzen* geändert in *Format 2: Dateien und Dateiverzeichnisse in ein anderes Dateiverzeichnis versetzen* und ergänzt um:

`mv_dvz1_datei_..._dvz`

datei

Namen der Dateien oder des Dateiverzeichnisse, die in das Dateiverzeichnis *dvz* übertragen werden sollen. Wenn Sie ein Dateiverzeichnis als Quelle angeben, werden alle darin enthaltenen Dateien und Unterverzeichnisse versetzt.

dvz

Name des Dateiverzeichnisses, in das die Dateien oder Dateiverzeichnisse übertragen werden sollen. Sie brauchen für das Ziel-Dateiverzeichnis Schreibrecht. Wenn *dvz* schreibbar ist, aber das t-Bit (sticky-Bit) gesetzt hat, muß eine der folgenden Bedingungen erfüllt sein, um die Dateien oder Dateiverzeichnisse nach *dvz* zu übertragen:

- die Datei muß dem Benutzer gehören
- das Ziel-Dateiverzeichnis muß dem Benutzer gehören
- der Benutzer muß Schreibberechtigung für die Datei haben
- der Benutzer muß ein privilegierter Benutzer sein

newgrp - Gruppenzugehörigkeit ändern

newgrp wird durch eine Signalbehandlung beim Kommandoaufruf ergänzt. Wird newgrp mit der Taste **DEL** in jener Phase abgebrochen, in der /bin/newgrp die aktuelle Shell überlagert, wird auch die Shell abgebrochen, von der newgrp aufgerufen wurde.

passwd - Login-Kennwort und Kennwortattribute eintragen oder ändern

Bei `passwd` gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable `LANG` bestimmt die Sprache der Meldungstexte.

Wenn `LANG` nicht oder als leere Zeichenkette definiert ist, verhält sich das System so, als wäre es nicht internationalisiert.

Es wird empfohlen, für Kennungen und Kennwörter nur Zeichen aus dem für portable Dateinamen vereinbarten Zeichensatz (Portable Filename Character Set) zu verwenden; 8-bit-Dateien sind unter Umständen nicht portabel zu nicht internationalisierten Systemen.

pr - Dateien formatieren und auf die Standardausgabe ausgeben

Bei `pr` wurden folgende Optionen ergänzt und geändert:

`-a`

(a - across) Füllt die Spalten einer Seite von links nach rechts. Die Anzahl der Spalten muß größer als 1 sein. Diese Anzahl können Sie mit `-anzahl_spalten` oder `-w` bestimmen.

Die Option `-a` kann nicht zusammen mit der Option `-m` verwendet werden.

`-m`

Wenn Sie `-m` benutzen, können Sie maximal neun Dateien für `datei` angeben. Die Option `-m` kann nicht zusammen mit der Option `-a` verwendet werden.

ps - Prozeßdaten abfragen (process status)

Beim Kommando `ps` gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable `LANG` bestimmt die Sprache der Meldungstexte.

Wenn `LANG` nicht oder als leere Zeichenkette definiert ist, verhält sich das System so, als wäre es nicht internationalisiert.

rm - Dateien löschen

Bei `rm` gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable `LANG` bestimmt das Eingabeformat für die Antwort auf Ja/Nein-Abfragen in der internationalen Umgebung.

Die Antworten auf Ja/Nein-Abfragen müssen dabei immer in der durch `LANG` festgelegten Sprache eingegeben werden, auch dann, wenn die Abfragen vom System in einer anderen Sprache gestellt werden. Ist z.B. kein Meldungskatalog für `rm` vorhanden und hat `LANG` den Wert `De`, wird die Abfrage in englisch ausgegeben, die Antwort muß dagegen in deutsch eingegeben werden.

Wenn `LANG` nicht oder als leere Zeichenkette definiert ist, verhält sich das System so, als wäre es nicht internationalisiert.

Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich System so, als wäre keine Variable gesetzt.

sar - Über Systemtätigkeit berichten

Die Beschreibung von `sar` wird um das Ausgabefeld `%busy` ergänzt.

Bei einem System, das für seine Platten `look-ahead-caching` benutzt, kann der Wert für `%busy` die Hundertprozent-Marke überschreiten, wenn vor allem sequentielle Daten gelesen werden.

secure - Benutzerkennung überprüfen

`secure` überprüft eine Benutzerkennung und gibt mögliche Sicherheitsrisiken auf dem Bildschirm aus.

Der Funktionsumfang von `secure` hängt davon ab, ob ein nicht-privilegiertes Benutzer oder der Systemverwalter das Kommando aufruft.

Wird das Kommando `secure` von einem nicht-privilegierten Benutzer aufgerufen, werden die Informationen nur für die Benutzerkennung des aufrufenden Benutzers ausgegeben. Ein Aufruf durch den Systemverwalter überprüft das gesamte Rechnersystem.

Für den nicht-privilegierten Benutzer bietet `secure` folgenden Funktionsumfang:

- Es werden alle Dateien ausgegeben, bei denen das s-Bit für den Eigentümer oder die Gruppe gesetzt ist. Diese Dateien stellen ein Sicherheitsrisiko dar, da jeder, der eine solche Datei ausführt die Zugriffsrechte des Eigentümers besitzt und mit diesen Rechten beliebige Aktionen ausführen kann.
- Es werden alle im Dateibaum des Benutzers befindliche Dateiverzeichnisse aufgelistet, bei denen für alle Systembenutzer die Schreibberechtigung erteilt ist.
- `secure` sucht nach Programmen mit dem Namen `su`.
- Die Shell-Variablen werden analysiert. Ist die Datei `.profile` in der analysierten Benutzerkennung nicht vorhanden, so wird die Datei `/etc/profile` verwendet.

Für den Systemverwalter bietet `secure` einen erweiterten Funktionsumfang. Folgende Informationen werden ausgegeben:

- Benutzerkennungen, die kein Passwort haben.
- "Pseudo-Benutzerkennungen", für die ein ungültiges Passwort angegeben ist, und die daher kein `login` erlauben.
- Benutzerkennungen mit doppelter Benutzernummer.
- Benutzerkennungen, für die kein Login-Dateiverzeichnis definiert ist.
- Benutzerkennungen, an denen sich in den letzten 30 Tagen niemand angemeldet hat.
- Dateien mit gesetztem s-Bit. Handelt es sich bei diesen Dateien um Textdateien, also möglicherweise um Shell-Prozeduren, so wird eine zusätzliche Warnung ausgegeben.
- Dateiverzeichnisse, bei denen für alle Systembenutzer die Schreibberechtigung erteilt ist.
- Dateien im Dateiverzeichnis `/dev`, die keine Gerätedateien sind.
- Gerätedateien im Dateiverzeichnis `/dev`, die für alle Systembenutzer schreibbar sind.

- Gerätedateien, die nicht im Dateiverzeichnis `/dev` stehen.
- Kopien von `/bin/sh`.
- Programme mit dem Namen `su`.
- Escape-Sequenzen in der Mail.

`secure` [option]

OPTIONEN

- h
Die Hilfe-Texte werden ausgegeben.
- l
`secure` versucht, das Passwort der Benutzerkennung zu ermitteln, um festzustellen, ob ein zu einfaches Passwort vergeben wurde. Dazu enthält `secure` eine interne Kennwortliste. Gesucht wird nach Benutzernamen, Vornamen, Firmennamen etc. Wird diese Option vom Systemverwalter aufgerufen, untersucht `secure` alle Benutzerkennungen. Dieser Vorgang kann dann sehr viel Zeit in Anspruch nehmen.
- s
Der Prozeß zur Ermittlung der Passwörter (siehe Option `-l`) läuft im Hintergrund ab. `secure` informiert während des Prozeßablaufs darüber, wieviel Prozent der Benutzer bereits überprüft wurden. Die Option `-s` kann nur vom Systemverwalter aufgerufen werden.
- t
Die Option `-t` ermöglicht es dem Systemverwalter, eine beliebige Benutzerkennung im Status eines nicht-privilegierten Benutzers mit `secure` zu analysieren. Nach Aufruf von `secure -t` wird der Systemverwalter aufgefordert, ein Passwort einzugeben. Mit Hilfe dieses Passworts kann er sich anschließend an einer beliebige Benutzerkennung über `login` anmelden und diese Benutzerkennung im Status eines nicht-privilegierten Benutzers mit `secure` analysieren. Die Option `-t` kann nur vom Systemverwalter aufgerufen werden.

DATEIEN

Temporäre Dateien, die während des Ablaufs von `secure` angelegt werden:

`/tmp/sec_st`, `/tmp/no_dev`, `/tmp/out_dev`, `/tmp/s_bit`, `/tmp/t_bit`, `/tmp/w_dev`,
`/tmp/sh_cp`, `/tmp/f_su`, `/tmp/mail_e`

SIEHE AUCH

chmod, login, passwd, su

sed - Editor im Prozedurbetrieb

Bei den `sed`-Kommandos `a`, `c` und `i` ändert sich die Ausgabe des Arguments *text*. Führende Tabulator- oder Leerzeichen in *text* werden von `sed` nicht entfernt.

sh - Bourne-Shell

Für `sh` wurde die Variable `TZ` (Beginn und Ende der Sommerzeit) korrigiert. Der Wechsel von Orts- zu Sommerzeit findet am letzten Sonntag im März; von Sommerzeit zu Ortszeit wird am letzten Sonntag im September gewechselt (jeweils um 2 Uhr morgens).

Beispiel

```
TZ=MET-1MDT;M3.5.0/02:00:00,M9.5.0/03:00:00
```

sh1 - Schichtenverwaltung für Shells

In der Beschreibung des `sh1`-Kommandos `create` muß der Beispiel-Pfadname für ein virtuelles Terminal lauten: `/dev/sxt/003`.

sort - Dateien sortieren und/oder mischen

Bei `sort` gibt es folgende Ergänzung bezüglich der internationalen Umgebung:

Die Umgebungsvariable `LC_COLLATE` bestimmt die vorgegebene Sortierreihenfolge, die vom Kommando `sort` angewendet wird. `LC_CTYPE` bestimmt das Verhalten von Zeichenklassen beim Verwenden der Optionen `-d`, `-f` oder `-i`. `LC_NUMERIC` bestimmt das Format des Dezimalzeichens beim Verwenden der Option `-n`.

Wenn `LC_COLLATE`, `LC_CTYPE` oder `LC_TIME` nicht oder als leere Zeichenkette definiert ist, wird der Wert von `LANG` als Standardwert für die jeweils nicht gesetzte oder leere Variable herangezogen. Ist auch `LANG` nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert.

Hat eine der Variablen für die internationale Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

tail - Den letzten Teil einer Datei ausgeben

`tail` erfährt folgende Ergänzung bezüglich der Option `r`:

Wird die Option `r` zusammen mit einem Zahlenwert `[zahl]` verwendet, so darf nur ein negativer Zahlenwert angegeben werden. Das heißt, `+` darf in diesem Fall nicht mit `[zahl]` kombiniert werden.

talk - Dialog mit anderem Benutzer führen

In der Beschreibung zu `talk` muß das Zeichen `@` innerhalb der Netzadresse im 2. Beispiel nicht mit einem Gegenschrägstrich entwertet werden.

tar - Archivieren von Dateien auf Magnetbandkassette, Band oder Diskette und Archive bearbeiten

Durch die Internationalisierung gibt es beim tar-Kommando folgende Neuerungen:

Die Umgebungsvariable *LC_TIME* bestimmt das Format der Datums- und Zeitausgabe bei der Auflistung des Archivinhalts mit der Option *v*.

Die Umgebungsvariable *LANG* bestimmt das Eingabeformat für die Antwort auf Ja/Nein-Abfragen in der internationalen Umgebung.

Wenn *LC_TIME* nicht oder als leere Zeichenkette definiert ist, wird der Wert von *LANG* als Standardwert herangezogen. Ist auch *LANG* nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert. Hat eine der Variablen für die internationalisierte Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

Der Zeichensatz ISO 8859-1 wird bei der Datenübertragung zwischen Systemen verwendet, die dem X/Open-Standard entsprechen. 8-bit Daten und Dateinamen unterliegen gewissen Einschränkungen hinsichtlich der Portierung auf nicht-internationalisierte Systeme. Es ist deshalb empfehlenswert, daß Sie beim Datentransfer zwischen verschiedenen Systemen nur solche Zeichen verwenden, die im 7-bit-ASCII-Zeichenbereich (Zeichensatz ISO 646) definiert sind. Achten Sie darauf, daß Sie bei der Benennung von Dateien nur solche Zeichen aus dem für portable Dateinamen vereinbarten Zeichensatz (Portable Filename Character Set) verwenden.

time - Laufzeit eines Kommandos messen

In der Beschreibung zu `time` wurde der Absatz über Multiprozessor-Systeme ergänzt.

Wenn Sie `time` auf einem Multiprozessor-System aufrufen, dann ist die Summe aus der Zeit im Benutzermodus und der Zeit im Systemmodus deutlich größer als die reale Laufzeit. Die Angabe einer scheinbaren CPU-Auslastung von mehr als 100% ist also die Folge der Verteilung der Sohnprozesse auf verschiedene Prozessoren.

timex - Laufzeit eines Kommandos messen, Prozeßdaten und Systemaktivitäten anzeigen

In die Beschreibung zu `timex` wurde der folgende Absatz über Multiprozessor-Systeme eingefügt.

Wenn Sie `timex` auf einem Multiprozessor-System aufrufen, dann ist die Summe aus der Zeit im Benutzermodus und der Zeit im Systemmodus deutlich größer als die reale Laufzeit. Die Angabe einer scheinbaren CPU-Auslastung vom mehr als 100% ist also die Folge der Verteilung der Sohnprozesse auf verschiedene Prozessoren.

uniq - Mehrfache Zeilen suchen

Bei `uniq` ändert sich die Beschreibung der Option `+n`:

`+n`

Die ersten n Zeichen ab Zeilenanfang werden beim Vergleich der Zeilen nicht berücksichtigt.

Wird die Option `+n` zusammen mit der Option `-m` verwendet, so werden die ersten n Zeichen nach dem m -ten Feld beim Vergleich nicht berücksichtigt. Leerzeichen nach dem m -ten Feld werden nicht ignoriert, sondern müssen im Wert n berücksichtigt werden.

vi - Bildschirmorientierter Editor

Beim `vi` sind folgende Korrekturen und Ergänzungen zu beachten:

1. Kommando `g` entfällt

Das Kommando `[zeile]g` entfällt. Der Absatz ist zu streichen.

2. Text beim `vi`-Kommando "`Puffer[zahl]dposition`" bzw. "`Puffer[zahl]dd`" ergänzt.

Der Text wird von der aktuellen Position der Schreibmarke bis zum Ende des angegebenen Bereichs gelöscht, zusätzlich aber in den mit *Puffer* bezeichneten alphabetischen Puffer gesichert. *Puffer* ist ein Groß- oder ein Kleinbuchstabe.

3. Mißverständliche Formulierung bezüglich der Funktionsweise des Kommandos `L` korrigiert

Die Schreibmarke wird zu dem ersten Zeichen in der letzten Textzeile auf dem Bildschirm bewegt, bei dem es sich nicht um ein Leer- oder Tabulatorzeichen handelt. Wird *zahl* angegeben, so wird die Schreibmarke zu der Textzeile bewegt, die sich *zahl* Textzeilen oberhalb des unteren Bildschirmrandes befindet.

4. Änderung beim Kommando `p`

Dem Kommando `p` kann keine Zahl vorangestellt werden. Der entsprechende Hinweis `[zahl]` ist zu streichen.

5. Änderung beim Kommando `P`

Dem Kommando `P` kann keine Zahl vorangestellt werden. Der entsprechende Hinweis `[zahl]` ist zu streichen.

6. Mißverständliche Formulierungen beim `vi`-Kommando `t` korrigiert

(`t` - `to`) Auf `t` muß ein einzelnes Zeichen folgen. `vi` durchsucht die aktuelle Zeile in Vorwärtsrichtung (nach rechts) nach diesem Zeichen. Wenn das Zeichen gefunden wurde, positioniert der `vi` die Schreibmarke unmittelbar vor das Zeichen. Wenn Sie einen Wiederholungsfaktor angeben, positioniert `vi` vor das *zahl*-te Zeichen. Eine Wiederholung des Suchkommandos mit `,` bzw. `;` ist nicht sinnvoll.

7. Mißverständliche Formulierungen beim `vi`-Kommando `T` korrigiert

(`T` - `to`) Auf `T` muß ein einzelnes Zeichen folgen. `vi` durchsucht die aktuelle Zeile in Rückwärtsrichtung (nach links) nach diesem Zeichen. Wenn das Zeichen gefunden wurde, positioniert der `vi` die Schreibmarke unmittelbar hinter das Zeichen. Wenn Sie einen Wiederholungsfaktor angeben, positioniert `vi` hinter das *zahl*-te Zeichen. Eine Wiederholung des Suchkommandos mit `,` bzw. `;` ist nicht sinnvoll.

8. Mißverständliche Formulierung bezüglich der Ausgabe des `vi`-Kommandos `set all` korrigiert.

Es gibt zwei Typen von Optionen: Optionen mit Booleschen Werten und Optionen mit nicht-Booleschen Werten. Ein Beispiel für den Booleschen Typ ist die Option `showmode`. Ist sie gesetzt, gibt `set all` die Zeichenkette `showmode` aus, ist sie nicht gesetzt, gibt `set all` die Zeichenkette `noshowmode` aus. Bei nicht-Booleschen Typen gibt `set all` den Wert aus, auf den die jeweilige Option gesetzt ist. Ein Beispiel für den nicht-Booleschen Typ ist `scroll`.

wc - Wörter, Zeichen und Zeilen zählen (word count)

Durch die Internationalisierung hat sich für das Kommando `wc` folgendes geändert:

Die Umgebungsvariable `LC_CTYPE` bestimmt, welche Zeichen als Zwischenraumzeichen eingestuft werden.

Wenn `LC_CTYPE` nicht oder als leere Zeichenkette definiert ist, wird der Wert von `LANG` als Standardwert herangezogen. Ist auch `LANG` nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert. Hat eine der Variablen für die internationalisierte Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

wcheck - Veränderte Dateien suchen (write check)

wcheck gibt alle jene Dateien eines Benutzers aus, die zwischen der letzten Abmeldung und der Anmeldung der aktuellen Sitzung verändert wurden.

Meldet sich ein Benutzer zum erstenmal am System an, wird als Zeitpunkt der letzten Abmeldung der 1. Januar 1970, 1:00 Uhr festgelegt.

Das Kommando wcheck sollte aus Sicherheitsgründen in die Datei `.profile` eingetragen werden. Dadurch kann sich der Benutzer bei jeder Anmeldung am System vergewissern, daß in der Zwischenzeit keine Manipulationen an seinen Dateien vorgenommen wurden.

Wird eine Datei, die in der angegebenen Zeitspanne verändert oder gelesen wurde, während der aktuellen Sitzung erneut verändert oder gelesen, so wird der Zeitstempel erneuert. Diese Datei wird dann von wcheck nicht mehr als veränderte Datei erkannt. Aus diesem Grund sollte wcheck unmittelbar nach der Anmeldung am System aufgerufen werden.

`wcheck [option]`

OPTION

-a

wcheck gibt zusätzlich jene Dateien aus, die zwischen der letzten Abmeldung und der Anmeldung der aktuellen Sitzung gelesen wurden.

DATEI

`/etc/wtmpx`

Die Datei `/var/adm/wtmpx` enthält Benutzer- und Verwaltungsinformationen. In dieser Datei sind die An- und Abmeldezeiten vermerkt.

who - Aktive Benutzerkennungen anzeigen

Die Beschreibung zu `who` enthält folgende Ergänzungen und Korrekturen:

1. Hinweis für Systemverwalter bezüglich `who` im Single-User-Modus.

Bei jedem Login werden die entsprechenden Informationen in der Datei `/etc/utmp` auf den neuesten Stand gebracht. Im Single-User-Modus erfolgt kein Login. Nach einem Shutdown in den Single-User-Modus kann `who` deshalb keine korrekten Informationen über den augenblicklichen Login-Status liefern. Verwenden Sie statt dessen `who am i`.

2. Option `-p` korrigiert und ergänzt.

Das Feld *Leitung* (LINE) hat im Zusammenhang mit `-p` keine Bedeutung, daher wird in diesem Feld nur ein Punkt ausgegeben. Das Feld *Kommentar* (COMMENT) wird nur dann ausgegeben, wenn Sie `-p` mit `-a` oder mit `-l` und/oder `-u` kombinieren. In einem dieser Fälle enthält das Kommentarfeld den Eintrag des Identifikationsfeldes der entsprechenden Zeile aus `/etc/inittab`.

3. Internationalisierung

Die Umgebungsvariable `LC_TIME` bestimmt das Format der Datums- und Zeitausgabe in der Spalte *Zeit*.

Wenn `LC_TIME` nicht oder als leere Zeichenkette definiert ist, wird der Wert von `LANG` als Standardwert herangezogen. Ist auch `LANG` nicht oder als leere Zeichenkette definiert, verhält sich das System so, als wäre es nicht internationalisiert. Hat eine der Variablen für die internationalisierte Umgebung einen ungültigen Wert, verhält sich das System so, als wäre keine Variable gesetzt.

tcpdump(1)

Dieses Kommando sollte nur zu Diagnosezwecken benutzt werden.

BEZEICHNUNG

tcpdump - Informationen über Datenverkehr in einem Netzwerk ausgeben

ÜBERSICHT

```
tcpdump [-deflnNOpqStvx] [-c count] [-F File] [-i interface] [-r file] [-s snapshotlen]  
        [-w file] expression
```

BESCHREIBUNG

tcpdump gibt die Kopfsätze der Pakete auf einer Netzwerk-Schnittstelle aus, die zum Booleschen Wert *expression* passen. Zum Aufrufen von tcpdump benötigen Sie die root-Berechtigung.

OPTIONEN

- c *count*
Kommandoausführung nach dem Empfang von *count* Paketen beenden.
- d
Übersetzten Paketvergleichs-Code auf der Standardausgabe ausgeben und Kommandoausführung beenden.
- e
Gibt in jeder Zeile den Kopfsatz auf Verbindungsebene (link-level header) aus.
- f
foreign (ferne) Internet-Adressen im numerischen und nicht im symbolischen Format ausgeben (durch diese Option soll eine schwerwiegende Funktionsstörung (Sun) yp-Server der Sun verhindert werden - in der Regel gerät er beim Übersetzen von nicht-lokalen Internet-Nummern in einen unendlichen Hängezustand).
- F *File*
File als Eingabe für den Filter-Ausdruck (*expression*) benutzen. Weitere in der Kommandozeile angegebene Ausdrücke werden gegebenenfalls ignoriert.
- i *interface*
Schnittstelle *interface* überwachen (listen). Wenn diese Option nicht angegeben wird, sucht tcpdump in der System-Schnittstellenliste nach der konfigurierten Schnittstelle mit der niedrigsten Nummer (Loopback-Schnittstellen werden jedoch nicht berücksichtigt).

- l
Versetzt die Standardausgabe in den zeilengepufferten Modus. Diese Option ist immer dann nützlich, wenn Sie sich die Daten beim Erfassen ansehen möchten.
- Beispiel*
- ```
tcpdump -l | tee dat
oder
tcpdump -l > dat & tail -f dat
```
- n  
Adressen (d.h. Hostadressen, port numbers (Anschlußnummern) usw.) nicht in Namen umwandeln.
- N  
Hostnamen ohne Domännennamen ausgeben. Dadurch zeigt tcpdump bei Angabe dieses Parameters beispielsweise nur *nic* an und nicht *nic.ddn.mil*.
- O  
Optimierer für Paketvergleichs-Code nicht aufrufen. Diese Option ist nur dann hilfreich, wenn Sie vermuten, daß der Optimierer einen Fehler hat.
- p  
Die Schnittstelle soll *nicht* in den "promiscuous mode" versetzt werden. Beachten Sie bitte, daß die Schnittstelle sich aus irgendeinem anderen Grund im "promiscuous mode" befinden könnte; somit kann -p nicht als Abkürzung für *ether host {localhost} or broadcast* benutzt werden.
- q  
Schnelle Ausgabe. Gibt weniger Protokollinformationen aus, so daß die Zeilen kürzer werden.
- r *file*  
Pakete aus *file* einlesen (mit der Option -w angelegt). Bei der Angabe von - anstatt *file* wird die Standardeingabe benutzt.
- s *snaplen*  
Aus jedem Paket sollen *snaplen* Byte geholt werden und nicht die Standardanzahl von 68 Byte (unter NIT beträgt die Mindestanzahl in Wirklichkeit 96 Byte). 68 Byte eignen sich für IP, ICMP, TCP und UDP; allerdings kann es dabei vorkommen, daß Protokollinformationen aus Namenserver- und NFS-Paketen abgeschnitten werden (siehe unten). Pakete, die wegen eines zu kleinen *snaplen*-Werts abgeschnitten werden, werden auf der Ausgabe mit [*proto*] gekennzeichnet. *proto* bezeichnet den Namen der Protokollebene, auf dem die Kürzung erfolgte.
- Beachten Sie, daß sich bei einer Erhöhung von *snaplen* die Verarbeitungszeit der Pakete verlängert und letztendlich die Anzahl der gepufferten Pakete verringert. Dadurch kann es vorkommen, daß Pakete verlorengehen.

Daher sollten Sie für *snapslen* den kleinsten Wert angeben, bei dem die benötigten Protokollinformationen noch erfaßt werden.

-S

Absolute und keine relativen TCP-Folgenummern ausgeben.

-t

Die einzelnen Zeilen sollen *ohne* Zeitstempel ausgegeben werden.

-tt

Jede Zeile mit einem unformatierten Zeitstempel ausgeben.

-v

(Geringfügig) ausführlichere Ausgabe. Beispielsweise werden die Lebensdauer (time to live) und der Typ der Service-Informationen in einem IP-Paket ausgegeben.

-w *file*

Die Raw-Pakete sollen nach *file* geschrieben und nicht analysiert und ausgegeben werden. Sie können die Pakete später mit der Option *-r* ausgeben. Bei der Angabe eines Bindestrichs (-) anstelle von *file* wird die Standardausgabe benutzt.

-x

Jedes Paket (ohne seinen Kopfsatz auf Verbindungsebene) im Hexadezimalformat ausgeben. Es werden der kleinere Teil des Gesamtpakets bzw. *snapslen* Bytes ausgegeben.

*expression*

Gibt die Pakete an, über die Informationen ausgegeben werden sollen. Ohne die Angabe von *expression* werden Informationen über sämtliche Pakete im Netzwerk ausgegeben. Andernfalls werden lediglich die Pakete berücksichtigt, für die *expression* zutrifft.

*expression* besteht aus ein oder mehreren *primitives* (Teilausdrücken). Ein *primitive* besteht im Normalfall aus der Kennung *id* (Name oder Nummer), der ein oder mehrere Bezeichner vorangestellt sind. Man unterscheidet zwischen drei Arten von Bezeichnern:

*type*

Ein derartiger Bezeichner gibt an, auf was sich die alphanumerische bzw. numerische Kennung bezieht. *type* kann sein: *host*, *net* oder *port* (z.B. *host foo*, *net 128.3*, *port 20*). Ohne die Angabe eines Bezeichners wird *host* eingesetzt.

*dir*

Ein derartiger Bezeichner gibt eine bestimmte Übertragungsrichtung von/zu *id* an. Mögliche Richtungen sind: *src*, *dst*, *src or dst* und *src and dst* (z.B. *foo*, *dst net 128.3*, *src or dst port ftp-data*). Ohne die Angabe eines *dir*-Bezeichners wird *src or dst* eingesetzt.

*proto*

Ein derartiger Bezeichner wählt ein bestimmtes Protokoll aus. *proto* kann sein: ether, ip, arp, rarp, tcp oder udp (z.B. ether src foo, arp net 128.3, tcp port 21). Wird kein Protokoll-Bezeichner angegeben, werden sämtliche Protokolle eingesetzt, die zum Protokolltyp konsistent sind.

*Beispiele*

src foo steht für (ip or arp or rarp) src foo (letzteres ist allerdings keine zulässige Syntax), net bar steht für (ip or arp or rarp) net bar und port 53 steht für (tcp or udp) port 53.

Zusätzlich gibt es eine Reihe spezieller Teilausdruck-Schlüsselwörter, die nicht nach diesem Muster aufgebaut sind: gateway, broadcast, less, greater sowie arithmetische Ausdrücke. Diese Schlüsselwörter werden unten beschrieben.

Komplexere Filter-Ausdrücke werden durch die Verknüpfung von Teilausdrücken mit Hilfe der Wörter and, or und not gebildet (z.B. host foo and not port ftp and not port ftp-data). Um den Eingabeaufwand zu begrenzen, müssen identische Bezeichnerlisten nur jeweils einmal eingegeben werden.

*Beispiel*

tcp dst port ftp or ftp-data or domain hat genau dieselbe Bedeutung wie tcp dst port ftp or tcp dst port ftp-data or tcp dst port domain.

**Zulässige Teilausdrücke**dst host *host*

Wahr, wenn im IP-Empfängerfeld des Pakets die Angabe *host* enthalten ist; dabei darf es sich entweder um eine Adresse oder um einen Namen handeln.

src host *host*

Wahr, wenn das IP-Senderfeld des Pakets die Angabe *host* enthält.

host *host*

Wahr, wenn entweder das IP-Sender- oder Empfängerfeld des Pakets die Angabe *host* enthält. Jedem der oben aufgeführten Hostrechner-Ausdrücke können die Schlüsselwörter ip, arp oder rarp vorangestellt werden (z.B. ip host *host* ist identisch mit ether proto \ip and host *host*). Wenn *host* ein Name mit mehreren IP-Adressen ist, wird jede Adresse auf Übereinstimmung überprüft.

ether dst *ehost*

Wahr, wenn die Ethernet-Zieladresse *ehost* ist. *ehost* kann entweder ein Name aus /etc/ethers oder eine Nummer sein (numerisches Format siehe ethers(3N)).

ether src *ehost*

Wahr, wenn die Ethernet-Ausgangsadresse *ehost* ist.

ether host *ehost*

Wahr, wenn die Ethernet-Ausgangs- oder Zieladresse *ehost* ist.

gateway *host*

Wahr, wenn das Paket *host* wie einen Gateway verwendet hat; das bedeutet, wenn die Ethernet-Ausgangs- oder Zieladresse *host* war, aber die Angabe *host* weder im Ausgangs- noch im Empfängerfeld des IP-Kopfsatzes enthalten war. *host* muß alphanumerisch und sowohl in `/etc/hosts` als auch in `/etc/ethers` enthalten sein. (Ein gleichbedeutender Ausdruck ist ether host *ehost* and not host *host*; dieser kann entweder für Namen oder Zahlen für *host/ehost* benutzt werden.)

dst net *net*

Wahr, wenn die IP-Zieladresse des Pakets die Netzwerknummer *net* hat; dabei kann es sich entweder um eine Adresse oder einen Namen handeln.

src net *net*

Wahr, wenn die IP-Ausgangsadresse des Pakets die Netzwerknummer *net* hat.

net *net*

Wahr, wenn entweder die IP-Ausgangs- oder die Zieladresse des Pakets die Nummer *net* hat.

dst port *port*

Wahr, wenn das Paket mit `ip/tcp` oder `ip/udp` übermittelt wird und als "destination port value" (Zielanschluß) *port* hat. *port* kann entweder eine Nummer oder ein Name aus `/etc/services` sein (siehe `tcp(7)` und `udp(7)`). Bei der Angabe eines Namens werden sowohl die Portnummer als auch das Protokoll überprüft. Wird eine Nummer oder ein zweideutiger Name benutzt, wird lediglich die Portnummer (Anschlußnummer) überprüft (so wird z.B. mit `dst port 513` sowohl `tcp/login traffic` als auch `udp/who traffic` ausgegeben, mit `port domain` sowohl `tcp/domain` als auch `udp/domain traffic`).

src port *port*

Wahr, wenn das Paket am Ausgangsport (Ausgangsanschluß) *port* abgeschickt wurde.

port *port*

Wahr, wenn *port* entweder der Ausgangs- oder der Zielport des Pakets ist. Jedem der obengenannten Ausdrücke zur Angabe eines Ports kann das Schlüsselwort `tcp` oder `udp` vorangestellt werden (z.B. paßt `tcp src port port` nur zu `tcp`-Paketen).

**less *length***

Wahr, wenn die Länge des Pakets kleiner oder gleich *length* ist. Dies entspricht der folgenden Angabe:

*len* <= *length*.

**greater *length***

Wahr, wenn das Paket größer oder gleich *length* ist. Dies entspricht der folgenden Angabe:

*len* >= *length*.

**ip proto *protocol***

Wahr, wenn das Paket ein ip-Paket (siehe ip(7)) mit dem Protokoll-Typ *protocol* ist. *protocol* kann eine Nummer oder einer der Namen *icmp*, *udp*, *nd* oder *tcp* sein. Beachten Sie, daß die Bezeichner *tcp*, *udp* und *icmp* ebenfalls Schlüsselwörter sind, die mit einem Gegenschrägstrich (\) entwertet werden müssen; in der C-Shell müssen hierzu zwei Gegenschrägstriche (\\) angegeben werden.

**broadcast**

Wahr, wenn es sich bei dem Paket um ein Broadcast-Paket handelt.

**ether proto *protocol***

Wahr, wenn es sich bei dem Paket um ein Ethernet-Paket vom Typ *protocol* handelt. *protocol* kann eine Nummer oder ein Name wie z.B. *ip*, *arp* oder *rarp* sein. Beachten Sie, daß es sich auch bei diesen Bezeichnern um Schlüsselwörter handelt, die mit einem Gegenschrägstrich (\) entwertet werden müssen.

**ip, arp, rarp**

Sind Abkürzungen für ether proto *p*, wobei *p* eines der genannten Protokolle ist.

**tcp, udp, icmp**

Sind Abkürzungen für ip proto *p*, wobei *p* eines der genannten Protokolle ist.

***expr relop expr***

Wahr, wenn der Vergleich zutrifft; *relop* kann dabei sein: >, <, >=, <=, =, !=; *expr* ist ein arithmetischer Ausdruck aus ganzzahligen Konstanten (die im Standard-C-Format angegeben werden), den normalen binären Operatoren [+ , - , \* , / , & , | ], einem Längen-Operator sowie speziellen Zugriffsfunktionen für die Paket-Daten. Der Zugriff auf die Daten im Paket erfolgt in der Syntax:

*proto* [ *expr* : *size* ]

*proto* kann entweder ether, ip, arp, rarp, tcp, udp oder icmp sein und gibt die Protokollebene für die Indizierungs-Operation an. Der Byte-Offset relativ zur angegebenen Protokollebene wird durch *expr* angegeben.

*size* ist optional und steht für die Anzahl der Bytes im betreffenden Feld (1, 2 oder 4; Standard: 1). Der Längen-Operator (Schlüsselwort *len*) gibt die Länge

des Pakets an.

So wird z.B. mit `ether[0] & 1 != 0` sämtlicher Multicast-Datenverkehr erfaßt. Der Ausdruck `ip[0] & 0xf != 5` erfaßt sämtliche IP-Pakete mit Optionen. Der Ausdruck `ip[2:2] & 0x1fff = 0` erfaßt ausschließlich unfragmentierte Datagramme sowie Fragment 0 von fragmentierten Datagrammen. Diese Überprüfung wird implizit auf die `tcp`- und die `udp`-Indizierungsoperationen angewandt. So steht z.B. `tcp[0]` in jedem Fall für das erste Byte des TCP-Kopfsatzes, in keinem Fall für das erste Byte eines der nachfolgenden Fragmente.

Teilausdrücke können folgendermaßen miteinander verknüpft werden:

- Durch Klammerung von Teilausdrücken und Operatoren (Klammern müssen entwertet werden, damit sie nicht von der Shell interpretiert werden).
- Durch Negation (! oder `not`).
- Durch logisches UND (`and`).
- Durch logisches ODER (`or`).

Die Negation hat die höchste Priorität. Die ODER- und die UND-Verknüpfung haben die gleiche Priorität und werden ihrer Reihenfolge entsprechend ausgewertet. Beachten Sie, daß eine UND-Verknüpfung jetzt explizit durch den Token `and` angegeben werden muß; eine Aneinanderreihung ist nicht mehr ausreichend.

Wenn ein Bezeichner ohne ein Schlüsselwort angegeben wird, wird das jeweils zuletzt angegebene Schlüsselwort eingesetzt (z.B. ist `not host vs and ace` die Kurzform für `not host vs and host ace`, was nicht mit `not (host vs or ace)` verwechselt werden darf).

Ausdrucks-Argumente werden an `tcpdump` entweder als einzelnes Argument oder als Mehrfach-Argument. übergeben; Sie können die Möglichkeit benutzen, die in Ihrem Fall am effizientesten ist. Ein Ausdruck mit Shell-Metazeichen wird im allgemeinen am besten als einzelnes, in Hochkomma eingeschlossenes Argument übergeben. Werden mehrere Argumente angegeben, so werden sie vor der Analyse durch Leerzeichen miteinander verknüpft.

## BEISPIELE

Alle Pakete ausgeben, die von `sundown` empfangen bzw. abgeschickt werden:

```
tcpdump host sundown
```

Datenverkehr zwischen `helios` und `hot` oder `ace` ausgeben:

```
tcpdump host helios and \(hot or ace \)
```

Sämtliche IP-Pakete zwischen `ace` und allen Hostrechnern außer `helios` ausgeben:

```
tcpdump ip host ace and not helios
```

Datenverkehr zwischen den lokalen Hostrechnern und den Hostrechnern in Berkeley ausgeben:

```
tcpdump net ucb-ether
```

Sämtlichen `ftp`-Datenverkehr ausgeben, der über das Internet-Gateway `snup` läuft (beachten Sie, daß der Ausdruck in Hochkomma eingeschlossen ist, um ihn vor einer (versehentlichen) Interpretation durch die Shell zu schützen):

```
tcpdump 'gateway snup and (port ftp or ftp-data)'
```

Sämtlichen Datenverkehr ausgeben, der weder von einem lokalen Hostrechner ausging noch an einen lokalen Hostrechner gerichtet war (wenn es sich um eine Weiterleitung zu einem anderen Netzwerk handelt, sollten die betreffenden Daten auf keinen Fall in Ihr lokales Netzwerk eingespeist werden).

```
tcpdump ip and not net localnet
```

Das erste und letzte Paket (die SYN- und FIN-Pakete) jeder TCP-Kommunikation ausgeben, an der ein nicht-lokaler Hostrechner beteiligt war.

```
tcpdump 'tcp[13] & 3 != 0 and not src and dst net localnet'
```

Alle IP-Pakete ausgeben, die länger als 576 Byte waren und über das Gateway `snup` geschickt wurden:

```
tcpdump 'gateway snup and ip[2:2] > 576'
```

Alle IP-Broadcast- oder Multicast-Pakete ausgeben, die *nicht* als Ethernet-Broadcast bzw. -Multicast-Pakete geschickt wurden:

```
tcpdump 'ether[0] & 1 = 0 and ip[16] >= 224'
```

Alle ICMP-Pakete ausgeben, die keine Anforderungen/Antworten zurückgeben (d.h., es sollen keine Pakete an Netzwerk-Hostrechner geschickt werden):

```
tcpdump 'icmp[0] != 8 and icmp[0] != 0'
```

## AUSGABEFORMAT

Die Ausgabe von `tcpdump` ist protokollabhängig. Nachfolgend finden Sie eine kurze Beschreibung und Beispiele für einen Großteil der Formate.

### Kopfsätze auf Verbindungsebene (Link-level headers)

Wenn die Option `-e` angegeben wird, wird der Kopfsatz auf Verbindungsebene ausgegeben. In Ethernet-Netzwerken werden die Ausgangs- und Zieladressen, das Protokoll sowie die Paketlänge ausgegeben.

#### *Hinweis*

In der nachfolgenden Beschreibung werden Kenntnisse des SLIP-Komprimierungsalgorithmus vorausgesetzt, wie er im RFC-1144 beschrieben wird.

In SLIP-Verbindungen werden ein Richtungsanzeiger (I für inbound, O für outbound), der Pakettyp sowie Komprimierungsinformationen ausgegeben. Der Pakettyp steht auf der Ausgabe an erster Stelle. Es gibt die drei Typen `ip`, `utcp` und `ctcp`. Für `ip`-Pakete werden keine weiteren Informationen über die Verbindung ausgegeben. Für TCP-Pakete wird der Verbindungsbezeichner nach dem Pakettyp ausgegeben.

Der Kopfsatz eines komprimierten Pakets wird in kodierter Form ausgegeben. Die Sonderfälle werden im Format `*S+n` und `*SA+n` ausgegeben, wobei `n` für das Delta steht, um das sich die Folgenummer (sequence number) (bzw. die Folgenummer und `ack`) geändert hat. Wenn es sich nicht um einen Sonderfall handelt, werden null oder mehr Änderungen ausgegeben. Bei einer Änderung wird ein U (urgent pointer), W (window), A (`ack`), S (sequence number) oder I (Paket-ID) ausgegeben, gefolgt von einem Deltawert (`+n` oder `-n`) oder einem neuen Wert (`=n`). Abschließend wird die Menge der Daten im Paket sowie die Länge des komprimierten Kopfsatzes ausgegeben.

Im folgenden Beispiel zeigt die Zeile einen abgehenden komprimierten TCP-Paket mit einem impliziten Verbindungsbezeichner; `ack` hat sich um den Wert 6 geändert, die Folgenummer um den Wert 49 und die Paket-ID um den Wert 6; die Daten haben eine Länge von 3 Byte, der komprimierte Kopfsatz hat eine Länge von 6 Byte:

```
0 ctcp * A+6 S+49 I+6 3 (6)
```

### ARP/RARP-Pakete

In der Ausgabe für ARP/RARP-Pakete werden die Art der Anforderung sowie die zugehörigen Argumente angezeigt. Das Format ist selbsterklärend. Der Anfang einer `rlogin`-Anforderung von Hostrechner `rtsg` an Hostrechner `csam` kann z.B. folgendermaßen aussehen:

```
arp who-has csam tell rtsg
arp reply csam is-at CSAM
```

Die erste Zeile bedeutet, daß `rtsg` ein `arp`-Paket abschickt, das die Ethernet-Adresse des Internet-Hostrechners `csam` anfordert. `csam` schickt als Antwort seine Ethernet-Adresse (in diesem Beispiel werden die Ethernet-Adressen in Großbuchstaben und die Internet-Adressen in Kleinbuchstaben angegeben).

Mit `tcpdump -n` wären weniger überflüssige Informationen enthalten:

```
arp who-has 128.3.254.6 tell 128.3.254.68
arp reply 128.3.254.6 is-at 02:07:01:00:01:c4
```

Mit `tcpdump -e` wäre erkennbar, daß das erste Paket ein Broadcast- und das zweite ein POINT-TO-POINT-Paket ist:

```
RTSG Broadcast 0806 64: arp who-has csam tell rtsg
CSAM RTSG 0806 64: arp reply csam is-at CSAM
```

Dies bedeutet für das erste Paket, daß RTSG die Ethernet-Ausgangsadresse ist, das Ziel die Broadcast-Adresse ist, das Typfeld den Hexadezimalwert 0806 (Typ `ETHER_ARP`) enthält und die Gesamtlänge 64 Byte beträgt.

## TCP-Pakete

### *Hinweis*

In der folgenden Beschreibung werden Kenntnisse des TCP-Protokolls vorausgesetzt, das im RFC-793 beschrieben wird. Ohne derartige Kenntnisse werden Sie weder mit der Beschreibung noch mit `tcpdump` allzuviel anfangen können.

Eine `tcp`-Protokollzeile hat folgendes allgemeines Format:

*src* > *dst*: *flags data-seqno ack window urgent options*

*src* und *dst* sind die Ausgangs- und Ziel-IP-Adressen und -Anschlüsse.

*flags* ist entweder eine beliebige Kombination aus den Buchstaben S (SYN), F (FIN), P (PUSH) oder R (RST) oder ein einzelner Punkt (.) für "keine Flags".

*data-seqno* beschreibt denjenigen Teil des Folgenummern-Bereichs, der durch die Daten in diesem Paket abgedeckt wird (siehe Beispiel unten).

*ack* ist die Folgenummer der Daten, die auf dieser Verbindung als nächstes in der anderen Richtung erwartet werden.

*window* ist die Anzahl der Bytes im Empfangspuffer, die auf dieser Verbindung in der anderen Richtung vorhanden sind.

*urg* gibt an, daß eilige Daten (*urgent*) im Paket enthalten sind.

*options* steht für `tcp`-Optionen, die in spitze Klammern eingeschlossen sein müssen(z.B. `<mss 1024>`).

*src*, *dst* und *flags* sind in jedem Fall vorhanden. Der Inhalt der übrigen Felder hängt vom Kopf des tcp-Protokolls ab und wird nur bei Bedarf ausgegeben.

Der erste Abschnitt einer *rlogin*-Anforderung von Hostrechner *rtsg* an Hostrechner *csam* sieht z.B. folgendermaßen aus:

```
rtsg.1023 > csam.login: S 768512:768512(0) win 4096 <mss 1024>
csam.login > rtsg.1023: S 947648:947648(0) ack 768513 win 4096 <mss 1024>
rtsg.1023 > csam.login: . ack 1 win 4096
rtsg.1023 > csam.login: P 1:2(1) ack 1 win 4096
csam.login > rtsg.1023: . ack 2 win 4096
rtsg.1023 > csam.login: P 2:21(19) ack 1 win 4096
csam.login > rtsg.1023: P 1:2(1) ack 21 win 4077
csam.login > rtsg.1023: P 2:3(1) ack 21 win 4077 urg 1
csam.login > rtsg.1023: P 3:4(1) ack 21 win 4077 urg 1
```

Die erste Zeile bedeutet, daß am tcp-Port 1023 von Hostrechner *rtsg* ein Paket an Port *login* von Hostrechner *csam* abgeschickt wurde. *S* bedeutet, daß das SYN-Flag gesetzt war. Das Paket hatte die Folgenummer 768512 und enthielt keine Daten (dabei wird das Format '*first:last(nbytes)*' mit der folgenden Bedeutung benutzt: Folgenummern von *first* bis ausschließlich *last*, was *nbytes* Byte von Benutzerdaten entspricht. Da es kein "piggy-backed" *ack* (begleitende Rückmeldung) gab, war ein Empfangsfenster mit einer Größe von 4096 Byte verfügbar, und es gab eine Option zur Angabe der maximalen Segmentgröße, die eine Meldung (*mss*) mit einer Länge von 1024 Byte anforderte.

*csam* antwortet mit einem ähnlichen Paket, enthält aber ein "piggy-backed" *ack* für das SYN-Flag von *rtsg*. *rtsg* bestätigt dann das SYN-Flag von *csam*. '.' bedeutet, daß keine Flags gesetzt waren. Da das Paket keine Daten enthielt, gibt es keine Daten-Folgenummer. Beachten Sie, daß die *ack*-Folgenummer eine Ganzzahl vom Typ *small* ist (1). Sobald *tcpdump* eine tcp-Kommunikation feststellt, gibt es die Folgenummer aus dem Paket aus. In den nachfolgenden Paketen der Kommunikation wird die Differenz zwischen der Folgenummer des aktuellen Pakets und dieser ersten Folgenummer ausgegeben. Somit können die Folgenummern nach der ersten als relative Byte-Positionen innerhalb des Datenstroms der Kommunikation interpretiert werden (wobei 1 das erste Daten-Byte in jeder Richtung ist). *-s* hebt diese Funktion auf, wodurch die ursprünglichen Folgenummern ausgegeben werden.

In der sechsten Zeile schickt *rtsg* an *csam* Daten mit einer Länge von 19 Byte (Byte 2 bis 20 im *rtsg* (-> *csam* Teil der Kommunikation). Im Paket wird das PUSH-Flag gesetzt. In der siebten Zeile meldet *csam*, daß er Daten bis ausschließlich Byte 21 von *rtsg* empfangen hat. Die meisten dieser Daten befinden sich offensichtlich im Socket-Puffer, da das Empfangsfenster von *csam* um 19 Byte geschrumpft ist. Außerdem schickt *csam* in diesem Paket ein Datenbyte an *rtsg*. In der achten und neunten Zeile schickt *csam* zwei "eilige" Datenbyte (*urg*) mit gesetztem PUSH-Flag an *rtsg*.

## UDP-Pakete

Das UDP-Format wird anhand des folgenden `rwho`-Pakets erläutert:

```
actinide.who > broadcast.who: udp 84
```

Diese Zeile bedeutet, daß der Port (Anschluß) `who` auf dem Hostrechner `actinide` ein `udp`-Datagramm an den Port `who` auf dem Hostrechner `broadcast` (der Internet-Broadcast-Adresse) geschickt hat. Das Paket enthielt Benutzerdaten mit einer Länge von 84 Byte.

Es werden einige UDP-Dienste erkannt (aufgrund der Ausgangs- oder Zielportnummer) und die Protokollinformationen der darüberliegenden Ebene ausgegeben. Dabei handelt es sich speziell um die Anforderungen des Domain Name-Service (RFC-1034/1035) und die Sun RPC-Aufrufe (RFC-1050) von NFS.

## Anforderungen des UDP-Namenservers

### *Hinweis*

In der folgenden Beschreibung werden Kenntnisse des Domain Service-Protokolls vorausgesetzt, das im RFC-1035 beschrieben wird. Ohne derartige Kenntnisse werden Sie mit der Beschreibung absolut nichts anfangen können.

Anforderungen des Namenservers werden in folgendem Format angegeben:

```
src > dst: id op? flags qtype qclass name (len)
```

```
h2opolo.1538 > helios.domain: 3+ A? ucbvax.berkeley.edu. (37)
```

Der Hostrechner `h2opolo` fragte beim Domänenserver auf `helios` nach einem Adreßdatensatz (*qtype*=A), der dem Namen `ucbvax.berkeley.edu` zugeordnet ist. Die Anfrage hatte die Kennung `3. +` bedeutet, daß das Flag `recursion desired` gesetzt war. Die Länge der Anfrage betrug 37 Byte, die Kopsätze des UDP- und des IP-Protokolls nicht mitgerechnet. Bei der Anfrage handelte es sich um eine normale Anfrage (*Query*), so daß das *op*-Feld nicht enthalten war. Jeder andere Inhalt des *op*-Felds wäre zwischen den Kennungen `3` und `+` ausgegeben worden. Auch bei *qclass* handelt es sich um eine normale Anfrage (C\_IN im Feld *qclass*) und wurde daher weggelassen. Jeder andere Inhalt von *qclass* wäre unmittelbar hinter `A` ausgegeben worden.

Die Anfrage wird auf bestimmte Besonderheiten überprüft; sind Besonderheiten vorhanden, so werden sie in zusätzlichen Feldern ausgegeben, die in eckige Klammern eingeschlossen sind. Wenn eine Anfrage einen Antwort-, Namenserver- oder Zonen-Teil enthält, werden `ancount`, `nscount` oder `arcount` im Format `[na]`, `[nn]` oder `[nau]` ausgegeben, wobei *n* der entsprechende Zählerwert ist. Wenn ein oder mehrere der Antwort-Bits (AA, RA oder rcode) gesetzt sind oder ein Teil der `must be zero`-Bits in Byte 2 und 3 gesetzt sind, wird `[b2&3=x]` ausgegeben, wobei *x* der Hexadezimalwert der Kopsatz-Bytes 2 und 3 ist.

## Antworten des UDP-Namenservers

Die Antworten des Namenservers werden in folgendem Format ausgegeben:

```
src > dst: id op rcode flags a/n/au type class data (len)
```

```
helios.domain > h2opolo.1538: 3 3/3/7 A 128.32.137.3 (273)
```

```
helios.domain > h2opolo.1537: 2 NXDomain* 0/1/0 (97)
```

Im ersten Beispiel antwortet `helios` auf die Anfrage mit der Kennung 3 von `h2opolo` mit drei Antwortsätzen, drei Namenserver-Sätzen und sieben Zonen-Sätzen. Der erste Antwortsatz ist vom Typ A (*address*) und enthält die Internet-Adresse 128.32.137.3. Die Gesamtlänge der Antwort beträgt 273 Byte, wobei die UDP- und die IP-Kopfsätze nicht mitgerechnet werden. Der Operationscode (*Query*) und der Antwortcode (`NoError`) wurden ebenso wie die Klasse (`C_IN`) des A-Datensatzes weggelassen.

Im zweiten Beispiel antwortet `helios` auf die Anfrage 2 mit einem Antwortcode für eine nicht vorhandenen Domäne (`NXDomain`), mit keinen Antwortsätzen, einem Namenserver-Satz und keinen Zonen-Sätzen. Der Stern (\*) bedeutet, daß das Bit *authoritative answer* gesetzt war. Da es keine Antwort gab, wurden weder der Typ noch die Klasse oder Daten ausgegeben.

Zusätzlich können die Flag-Zeichen – (*recursion available*, `RA`, *nicht* gesetzt) und `I` (*truncated message*, `TC`, gesetzt) enthalten sein. Wenn der Frage-Teil nicht exakt einen Eintrag enthält, wird `[nq]` ausgegeben.

Beachten Sie, daß die Namenserver-Anforderungen und -Antworten häufig sehr umfangreich sind und, wenn für *snapslen* der Standardwert (96 Byte) benutzt wird, die Größe des erfassten Paketteils für eine Ausgabe nicht ausreicht. Wenn Sie ausführliche Informationen über die Namenserver-Anforderungen und -Antworten benötigen, können Sie *snapslen* mit dem Schalter `-s` erhöhen. Gute Erfahrungen wurden mit der Einstellung `-s 128` gemacht.

## NFS-Anforderungen

Sun NFS- (Network File System-) Anforderungen und -Antworten werden in folgendem Format ausgegeben:

```
src.xid > dst.nfs: len op args
```

```
src.nfs > dst.xid: reply stat len
```

```
vs.e2766 > helios.nfs: 136 readdir fh 6.5197 8192 bytes @ 0
```

```
helios.nfs > vs.e2766: reply ok 384
```

```
vs.e2767 > helios.nfs: 136 lookup fh 6.5197 'RCS'
```

In der ersten Zeile schickt Hostrechner `vs` die Transaktion mit der Kennung `e2766` an `helios` (beachten Sie bitte, daß die Nummer nach dem Ausgangs-Hostrechner eine Transaktions-Kennung und *nicht* der Ausgangsport ist). Die Anforderung hatte eine Länge von 136 Byte, die UDP- und IP-Kopfsätze nicht eingerechnet. Bei der Operation handelte es sich um `readdir` (read directory) für das Dateikennzeichen (*file handle*, `fh`)

6.5197. Es werden 8192 Byte ab Offset 0 gelesen. *helios* gibt die Antwort *ok* mit 384 Daten-Bytes zurück.

In der dritten Zeile fordert *vs* den Rechner *helios* auf, den Namen *RCS* in der Verzeichnisdatei 6.5197 zu suchen. Beachten Sie, daß die ausgegebenen Daten vom Typ der Operation abhängen. Das Format sollte eigentlich selbsterklärend sein, wenn es zusammen mit einer NFS-Protokollspezifikation gelesen wird.

Beachten Sie bitte, daß NFS-Anforderungen sehr umfangreich sind und die obige Ausgabe erst nach einer Erhöhung von *snaps* erzeugt wird. Zur Überwachung des NFS-Datenverkehrs wurde *-s 192* benutzt.

### KIP-Appletalk (DDP in UDP)

Appletalk DDP-Pakete, die in UDP-Datagrammen verkapselt sind, werden entkapselt und als DDP-Pakete ausgegeben (d.h., sämtliche Informationen aus dem UDP-Kopfsatz werden verworfen). Anhand der Datei */etc/ataalk.names* werden Appletalk-Netzwerk- und Knotennummern in Namen umgesetzt. Die Zeilen in dieser Datei haben das folgende Format:

| <u>number</u> | <u>name</u> |
|---------------|-------------|
| 1.254         | ether       |
| 16.1          | icsd-net    |
| 1.254.110     | ace         |

In den ersten beiden Zeilen sind die Namen von Appletalk-Netzwerken enthalten. Die dritte Zeile enthält den Namen eines bestimmten Hostrechners (eine Hostnummer kann von einer Netzwerknummer durch die dritte Achtergruppe innerhalb der Zahl unterschieden werden - eine Netzwerknummer *muß* aus zwei Achtergruppen bestehen, eine Hostrechner-Nummer aus drei). Die Nummer und der Name sollten durch Zwischenraumzeichen (Leer- oder Tabulatorzeichen) voneinander getrennt werden. Die Datei */etc/ataalk.names* kann Leer- oder Kommentarzeilen enthalten (Zeilen, die mit den Zeichen *#* beginnen).

Appletalk-Adressen werden in folgendem Format ausgegeben:

*net.host.port*

```
144.1.209.2 > icsd-net.112.220
office.2 > icsd-net.112.220
jssmag.149.235 > icsd-net.2
```

(Wenn die Datei */etc/ataalk.names* nicht vorhanden ist oder keinen Eintrag für eine Appletalk-Hostrechner-/Netznummer enthält, werden die Adressen im numerischen Format ausgegeben). Im ersten Beispiel schickt *NBP* (DDP-Port 2) im Netzwerk 144.1 am Knoten 209 Daten an den Netzwerkdienst, der gerade am Port 220 am *icsd*-Netzwerk-knoten 112 wartet. Die zweite Zeile hat dieselbe Bedeutung, nur ist der Ausgangsknoten bekannt (*office*). Bei der dritten Zeile handelt es sich um einen Sendevorgang vom Port 235 am *jssmag*-Netzwerkknoten 149 zum Rundsenden an *NBP*-Port im

icsd-Netzwerk (beachten Sie, daß die Broadcast-Adresse (255) über einen Netzwerknamen ohne Netzwerknummer angegeben wird- aus diesem Grund sollten die Knotennamen in `/etc/ata/k.names` möglichst von den Netzwerknamen getrennt werden).

Der Inhalt von NBP- (name binding protocol) und ATP-(Appletalk transaction protocol) Paketen wird interpretiert. Bei anderen Protokollen werden lediglich der Protokollname (bzw. die Nummer, wenn für das Protokoll kein Name eingetragen ist) sowie die Paketgröße ausgegeben.

Das Format für NBP-Pakete sieht wie in den folgenden Beispielen aus:

```
icsd-net.112.220 > jssmag.2: nbp-lkup 190: "=:LaserWriter@*"
jssmag.209.2 > icsd-net.112.220: nbp-reply 190: "RM1140:LaserWriter@*" 250
techpit.2 > icsd-net.112.220: nbp-reply 190: "techpit:LaserWriter@*" 186
```

Die erste Zeile enthält eine Namenssuch-Anforderung für Laserdrucker (LaserWriter), die durch den Hostrechner 112 im Netzwerk `icsd` abgeschickt und im Netzwerk `jssmag` rundgesendet wird. Die Suche hat die `nbp`-Kennung 190. In der zweiten Zeile wird eine Antwort auf diese Anforderung angezeigt (beachten Sie, daß die Kennung dieselbe geblieben ist), die vom Hostrechner `jssmag.209` abgeschickt wurde. Diese Antwort bedeutet, daß eine Laserdrucker-Ressource namens `RM1140` an Port 250 eingetragen ist. Die dritte Zeile ist eine weitere Antwort auf dieselbe Anforderung; sie bedeutet, daß auf dem Hostrechner `techpit` der Laserdrucker `techpit:LaserWriter` am Port 186 registriert ist.

Das Format für ATP-Pakete sieht wie im folgenden Beispiel aus:

```
jssmag.209.165 > helios.132: atp-req 12266<0-7> 0xae030001
helios.132 > jssmag.209.165: atp-resp 12266:0 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:1 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:2 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:3 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:4 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:5 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:6 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp*12266:7 (512) 0xae040000
jssmag.209.165 > helios.132: atp-req 12266<3,5> 0xae030001
helios.132 > jssmag.209.165: atp-resp 12266:3 (512) 0xae040000
helios.132 > jssmag.209.165: atp-resp 12266:5 (512) 0xae040000
jssmag.209.165 > helios.132: atp-rel 12266<0-7> 0xae030001
jssmag.209.133 > helios.132: atp-req* 12267<0-7> 0xae030002
```

`jssmag.209` startet die Transaktion mit der Kennung 12266 mit dem Hostrechner `helios`, wobei er bis zu 8 Pakete anfordert (<0-7>). Bei der Hex-Nummer am Ende der Zeile handelt es sich um den Wert des `userdata`-Felds in der Anforderung.

`helios` antwortet mit 8 Paketen zu je 512 Byte. `:zahl` nach der Transaktions-Kennung gibt die Paket-Folgenummer innerhalb der Transaktion an; die Zahl innerhalb der Klammern steht für die Datenmenge im Paket, den `atp`-Kopfsatz ausgenommen. Das Zeichen `*` bei Paket 7 bedeutet, daß das EOM-Bit gesetzt war.

`jssmag.209` fordert dann die erneute Übertragung von Paket 3 und 5 an. `helios` schickt sie dann erneut ab; daraufhin gibt `jssmag.209` die Transaktion frei. Schließlich startet `jssmag.209` die nächste Anforderung. Das Zeichen \* bei der Anforderung bedeutet, daß **XO** (*exactly once*) *nicht* gesetzt war.

### IP-Fragmentierung

Fragmentierte Internet-Datagramme werden in folgendem Format ausgegeben:

```
(frag id:size@offset+)
(frag id:size@offset)
```

(Das erste Format wird verwendet, wenn es weitere Fragmente gibt. Das zweite Format wird für das letzte Fragment verwendet).

*id* ist die Fragment-Kennung (im Hex-Format). *size* ist die Fragment-Größe (in Byte), den IP-Kopf nicht eingerechnet. *offset* ist der Offset dieses Fragments (in Byte) innerhalb des ursprünglichen Datagramms.

Die Fragment-Informationen werden für jedes Fragment ausgegeben. Das erste Fragment enthält den Kopfsatz des darüberliegenden Protokolls; auf die Protokoll-Informationen folgen die Fragment-Informationen. Die Fragmente, die auf das erste Fragment folgen, enthalten keinen Kopfsatz des darüberliegenden Protokolls; die Fragment-Informationen werden nach den Ausgangs- und Empfänger-Adressen ausgegeben. Nachfolgend finden Sie z.B. einen Teil eines ftp von `arizona.edu` an `lbl-rtsg.arpa` über eine CSNET-Verbindung, die offensichtlich keine 576-Byte-Datagramme verarbeiten kann:

```
arizona.ftp-data > rtsg.1170: . 1024:1332(308) ack 1 win 4096
(frag 595a:328@0+)
arizona > rtsg: (frag 595a:204@328)
rtsg.1170 > arizona.ftp-data: . ack 1536 win 2560
```

Zu diesem Beispiel ist folgendes anzumerken: Erstens enthalten die Adressen in der zweiten Zeile keine Portnummern. Dies liegt daran, daß sämtliche TCP-Protokollinformationen im ersten Fragment enthalten sind und die Port oder Folgenummern bei der Ausgabe der darauffolgenden Fragmente vollkommen offen sind. Zweitens werden die tcp-Folgeinformationen in der ersten Zeile ausgegeben, als hätten die Benutzerdaten eine Länge von 308 Byte; in Wirklichkeit beträgt die Gesamtlänge jedoch 512 Byte (308 Byte im ersten Fragment und 204 im zweiten). Dies kann verwirrend sein, wenn Sie bei den Folgenummern nach Lücken suchen oder versuchen, Bestätigungen (*acks*) Paketen zuzuordnen.

Ein Paket mit dem IP-Flag `don't fragment` wird mit einem nachstehenden (DF) markiert.

### Zeitstempel

Standardmäßig wird allen Ausgabezeilen ein Zeitstempel vorangestellt. Der Zeitstempel zeigt die aktuelle Uhrzeit im Format

*hh:mm:ss.frac*

an und entspricht in seiner Genauigkeit der Uhrzeit im Systemkern (z.B.  $\pm 10\text{ms}$ ). Der Zeitstempel gibt den Zeitpunkt an, zu dem der Systemkern das Paket zum ersten Mal feststellte. Es wird nicht versucht, den Zeitversatz zwischen dem Zeitpunkt, zu dem die Ethernet-Schnittstelle das Paket aus der Leitung entfernte und dem Zeitpunkt, zu dem der Systemkern das `new packet-interrupt` (-Unterbrechung) bediente, auszugleichen.

## **addbad(1M) (MX500)**

Das `addbad`-Kommando kann ab SINIX V5.41 auch den Storage-Controller bedienen. Dazu stehen folgende Gerätetreiber-Dateien zur Verfügung:

```
/dev/rdisk/c8d0s0
```

```
/dev/rdisk/c9d0s0
```

```
/dev/rdisk/c9d1s0
```

Ansonsten gilt die Beschreibung im Handbuch "Referenzhandbuch für Systemverwalter" [11].

## bind(1M) (MX500)

### BEZEICHNUNG

bind - Prozeß an einen Prozessor binden

### ÜBERSICHT

/sbin/bind *proc kommando [parameter]*  
/sbin/bind -q

### BESCHREIBUNG

Das Kommando `bind` wird erst ab der Version SINIX V5.41B unterstützt. `bind` führt das angegebene Kommando auf einem bestimmten Prozessor aus und sorgt dafür, daß das Kommando und seine Sohnprozesse während der Ablaufzeit diesem Prozessor zugeordnet bleiben. Wird das Kommando mit der Option `-q` (für `query`) aufgerufen, so werden lediglich die Prozessoren aufgelistet, an die bereits Prozesse gebunden sind.

### OPTIONEN

#### *proc*

Nummer des gewünschten Prozessors. Die Prozessoren werden dabei mit 0 beginnend gezählt.

#### *kommando*

Das Kommando, das an einen bestimmten Prozessor gebunden werden soll. Auch alle von diesem Kommando erzeugten Sohn-Prozesse werden an diesen Prozessor gebunden.

#### *parameter*

Die Parameter des angegebenen Kommandos.

#### `-q`

Es werden alle Prozessoren aufgelistet, an die bereits Prozesse gebunden sind.

#### *Hinweis*

Das Kommando kann nur der Systemverwalter ausführen.

Es muß zumindest ein Prozessor ohne gebundene Prozesse im System verbleiben. Das Kommando `bind` bricht daher in folgenden Fällen mit einer Fehlermeldung ab:

- `proc` ist der einzige Prozessor im System
- `proc` ist der letzte Prozessor ohne gebundene Prozesse
- `proc` existiert nicht
- `proc` ist offline

*Beispiel*

```
/sbin/bind 0 1s -1
```

Das Kommando `1s -1` wird im obigen Beispiel auf dem Prozessor 0 ausgeführt.

## boot(1M) (MX300)

### BEZEICHNUNG

boot - Ladeprogramm des SINIX-Systems

### BESCHREIBUNG

Das Programm `boot` lädt interaktiv Programme, die eigenständig d.h. ohne Unterstützung durch ein Betriebssystem, auf der Hardware ablaufen können, und veranlaßt deren Ausführung. Typischerweise wird `boot` zum Laden und Ausführen des Systemkerns benutzt. Im Prinzip kann aber auch ein beliebiges anderes Programm geladen und ausgeführt werden, vorausgesetzt, es ist für eine eigenständige Ausführung gebunden worden (siehe `1d(1)`).

Eine weitere wichtige Funktion von `boot` ist das Kopieren des Hauptspeichers auf ein Massenspeicher-Medium, z.B. eine Festplatte oder ein Magnetband. Dieser Programmteil wird nur unter bestimmten Umständen (s.u.) aktiviert und im folgenden als "dump" bezeichnet.

`boot` selbst ist ein eigenständig ausführbares Programm und befindet sich auf dem Zylinder 0 der ersten Festplatte an einer vorgegebenen Adresse. Außerdem wird auf Diskette `SINIX0` ein speziell auf die Installation von SINIX zugeschnittenes `boot` mitgeliefert. Das Programm `boot` wird durch die Firmware der Maschine geladen. Standardmäßig werden zuerst die Diskettenlaufwerke und dann die erste Festplatte überprüft, ob sie `boot` enthalten.

### Ablauf von boot

Beim Starten des Systems ist im NVRAM (non-volatile RAM) die Information hinterlegt, ob die letzte Systembeendigung regulär oder irregulär war. Im letzteren Fall wird normalerweise ein Dump gezogen. Ist `AUTODUMP` (siehe Abschnitt *Boot-Optionen*) auf `YES` gesetzt, geschieht das automatisch. Bei `AUTODUMP=NO` fordert `boot` interaktiv Bestätigung an:

```
Do you want a memory dump [yn]?
```

Bei Eingabe `RETURN` wird der Standardwert `y` angenommen, ebenso, wenn die `TIMEOUT`-Zeit (siehe Abschnitt *Boot-Optionen*) abgelaufen ist. In diesem Fall wird `dump` aktiviert. `dump` stellt zunächst fest, ob an seiner vorgegebenen Zieladresse bereits ein älterer Speicherabzug existiert. Wenn ja, muß der neue Speicherabzug explizit bestätigt werden:

```
dump: Dump already in default slice,
Do you want to overwrite it?
yes: type 'y<ret>' for overwriting previous dump
no : type 'n<ret>' for no dump
```

Nach der Eingabe von *y* wird der Speicherinhalt auf eine vorgegebene Adresse kopiert. Dabei wird in 0.25 MB-Intervallen angezeigt, wieviel schon kopiert wurde.

Nach Beendigung von `dump` wird abhängig von `AUTOBOOT` mit den Standardwerten automatisch geladen, oder aber die Eingabeaufforderung

Enter name of programm to boot:

erscheint an der Konsole.

Bei RETURN oder abgelaufener TIMEOUT-Zeit wird `DEFBOOTSTR` (siehe Abschnitt *Boot-Optionen*) als Eingabe angenommen. Wenn das Programm erfolgreich lokalisiert werden kann, wird es geladen. Währenddessen erscheinen auf der Konsole die Ausgaben

```
loading text segment...
loading data segment...
```

Zu diesem Zeitpunkt kann der Ladevorgang durch Drücken der DEL-Taste unterbrochen werden. Kann das Programm aus irgendwelchen Gründen beim ersten Versuch nicht erfolgreich geladen werden, wird unabhängig von `AUTOBOOT` eine Eingabe verlangt:

Enter name of programm to boot:

Beim zweiten und allen folgenden Versuchen ist die TIMEOUT-Uhr nicht mehr aktiv, so daß auf jeden Fall eine Benutzereingabe erforderlich ist.

## Boot-Optionen

`boot` wird durch eine Reihe von Optionen gesteuert. Standardwerte dafür stehen entweder in einer Datei, üblicherweise `/stand/boot`, oder werden von `boot` vorbelegt. Die Optionen werden unverändert an das Programm weitergereicht, sofern sie nicht durch die Benutzereingabe überschrieben werden (s.u.). Über die Optionen `AUTOBOOT`, `AUTODUMP` und insbesondere `TIMEOUT` kann der komplette Lade- und Dumpvorgang automatisiert werden.

`boot` kennt folgende Optionen:

`AUTOBOOT` = YES oder NO

Legt fest, ob `boot` den Systemkern sofort lädt (YES) oder erst eine Eingabeaufforderung an der Konsole ausgibt (NO). Standardwert: YES.

`AUTODUMP` = YES oder NO

Gibt an, ob ein Speicherabzug (Dump) bei jeder nicht regulären Systembeendigung (Reset, Spannungsausfall, Systemabsturz) automatisch zu ziehen ist (YES) oder nur nach vorheriger Bestätigung durch den Systemverwalter (NO). Standardwert: YES.

`BOOTPROMPT` = *string*

Ändert die standardmäßige Eingabeaufforderung des Ladeprogramms zu der Zeichenkette *string*. Standardwert: Enter name of program to boot:

DEFBOOTSTR = *bootstring*

Der Name des zu ladenden Programms (und Parameter) werden auf *bootstring* eingestellt. Dieser Bootstring wird verwendet, wenn Sie die Eingabeaufforderung von boot nur mit RETURN quittieren oder aber wenn die Wartezeit gemäß TIMEOUT abläuft. Standardwert: hd(10,0)unix root=hd(1) swap=hd(2). Auf MX300 mit Onboard-SCSI-Hostadapter: shd(10,0)unix root=shd(1) swap=shd(x). Dabei ist x abhängig von der Auswahl der Systemplatte.

MEMRANGE = *bereich* [*bereich...*]

Teilt dem Programm boot diejenigen Speicherbereiche mit, deren Vorhandensein getestet werden soll. *bereich* hat das Format *start-ende*:0. *start* und *ende* sind Dezimaladressen, die durch einen Bindestrich getrennt sind (zum Beispiel 1M-4M, 1K-65536). Aufsteigende (z.B. 15M-16M) und absteigende (z.B. 16M-15M) Adreßbereiche werden unterstützt. Die Startadresse zählt mit zum Speicherbereich (inklusive), die Endadresse nicht (exklusive). Standardwert: 0-64M:0.

MEMREQ = *größe*

Weist boot an, eine Mindestspeichergröße von *größe* anzufordern. Findet boot nicht mindestens *größe* Byte Speicherplatz, gibt es eine Meldung aus und beendet sich. *größe* hat dasselbe Format wie die Adressen bei MEMRANGE. Standardwert: 0.

MREQMSG1 = *zeichenkette*

Diese Option legt die Nachricht fest, die ausgegeben wird, wenn die in MEMREQ geforderte Speichergröße nicht verfügbar ist. Standardwert: Insufficient memory for a useful system.

MREQMSG2 = *zeichenkette*

Gibt eine optionale zweite Zeile als Ergänzung zur Nachricht MREQMSG1 an. Standardwert: leer.

TIMEOUT = *anz*

Wenn TIMEOUT gesetzt ist, wartet boot genau *anz* Sekunden auf eine Benutzereingabe. Nach Ablauf dieser Zeit verwendet boot stattdessen den Bootstring, der DEFBOOTSTR zugewiesen ist. Ist *anz* gleich 0, dann ist die Uhr zum Warten nicht gesetzt. Standardwert: 0.

### Eingaben für boot (Bootstring)

Die wichtigsten Eingaben für boot sind Adresse und Name des zu ladenden Programms sowie Parameter dafür. Adresse und Name werden im "Standalone-Format" angegeben. Die Syntax der Eingabe lautet wie folgt:

```
gtyp(nr[,dist])[name] [par1] [par2]... [parn]
```

Dabei gilt:

*gtyp*

gibt an, von welchem Gerätetyp das Programm zu laden ist. Die unterstützten Gerätetypen sind *hd* (Festplatte) und *fd* (Diskette). Das gilt für Maschinen mit Interphase Storer. Für Maschinen mit Onboard-SCSI-Hostadapter können alternativ auch die Gerätetypen *shd* (Festplatte) und *sfd* (Diskette) benutzt werden.

*nr*

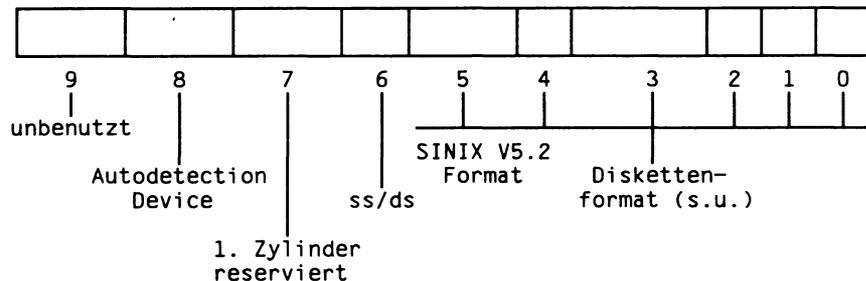
bezeichnet die Gerätenummer (minor device number) des Geräts. Sie wählt ein bestimmtes Gerät des angegebenen Typs aus und bestimmt darüber hinaus eine Adresse auf diesem Gerät.

Gerätenummern für Geräte am Interphase Storer werden nach dem folgenden Schema angegeben:

|           |                           |                         |                  |
|-----------|---------------------------|-------------------------|------------------|
| 31-17     | 16-14                     | 13-10                   | 9-0              |
| unbenutzt | Nummer<br>des Controllers | Nummer<br>des Laufwerks | gerätespezifisch |

Die Bits 9-0 geben im Fall einer Festplatte eine Slice-Nummer an. Die Numerierung von Controllern, Laufwerken und Slices beginnt immer bei 0. So bezeichnet z.B. *hd(1034)* die Slice Nr. 10 auf der zweiten Platte am ersten Controller.

Im Fall eines Diskettenlaufwerks sind die Bits 9-0 wie folgt zu interpretieren:



Bits 4-0 geben das Diskettenformat an. Folgende Disketten werden unterstützt:

| Format-Nr | Bytes/Sektor | Spuren/Zoll | Sektoren/Spur |                  |
|-----------|--------------|-------------|---------------|------------------|
| 0         | 512          | 96/135      | 15            | ss/ds            |
| 1         | 512          | 48          | 9             | ss/ds            |
| 2         | 512          | 48          | 8             | ss/ds            |
| 3         | 1024         | 48          | 4             | ss/ds            |
| 4         | 256          | 48          | 16            | ss/ds            |
| 5         | 512          | 96          | 9             | ss/ds            |
| 6         | 512          | 135         | 9             | ss/ds            |
| 7         | 512          | 135         | 18            | ss/ds            |
| 8         | 512          | 96          | 8             | ss               |
| 32        | 256          | 96          | 16            | SINIX 5.2 Format |
| 33        | 512          | 96          | 15            | SINIX 5.2 Format |

Gerätenummern für Geräte am Onboard-SCSI-Hostadapter NCR 53C710 werden nach dem folgenden Schema angegeben:

Festplatte hd (shd)

| 31-18        | 17-10 | 9-7    | 6-4 | 3-0   |
|--------------|-------|--------|-----|-------|
| major number | Ctlr  | Target | Lun | Slice |

Diskettenlaufwerk fd (sfd)

| 31-18        | 17-11      | 10-7 | 6-4    | 3   | 2     | 1-0   |
|--------------|------------|------|--------|-----|-------|-------|
| major number | Formatcode | Ctlr | Target | Lun | ss/ds | Slice |

Die Bedeutung der Felder innerhalb der Minor-Nummern sind für Geräte am NCR 53C710:

Ctlr

Die Controller-Nummer des NCR 53C710. Hier kann nur die Nummer 0 angegeben werden.

Target

Die SCSI-Targetnummer des Platten- oder Diskettenlaufwerks (0-7).

Lun

Die Lun-Nummer des SCSI-Targets. Für Festplatten wird nur die Lun 0 unterstützt, Diskettenlaufwerke können die Luns 0 oder 1 haben. Lun 0 ist üblicherweise die 3 1/2-Zoll-Diskette, Lun 1 die 5 1/4-Zoll Diskette.

Slice

Die Slicenummer der Festplatte oder Diskette.

ss/ds

Dieses Bit wird für "single-sided"-Disketten auf 1 gesetzt.

Formatcode

Der Formatcode spezifiziert das detaillierte Format der verwendeten Disketten.

| Formatcode | Bytes/Sektor | Spuren/Zoll | Zylinder |              |
|------------|--------------|-------------|----------|--------------|
| 0          | 512          | 15          | 80       | SINIX 5.4 HD |
| 1          | 512          | 9           | 40       | SINIX 5.4 DD |
| 2          | 512          | 9           | 80       | SINIX 5.4 HD |
| 3          | 512          | 8           | 40       | SINIX 5.4 DD |
| 4          | 1024         | 4           | 40       | SINIX 5.4 DD |
| 5          | 256          | 16          | 40       | SINIX 5.4 DD |
| 6          | 256          | 16          | 80       | SINIX 5.2    |
| 7          | 512          | 15          | 80       | SINIX 5.2    |
| 8          | 512          | 18          | 80       | SINIX 5.4 HD |
| 9          | 512          | 36          | 80       | SINIX 5.4 HD |
| 10         | 512          | 9           | 80       | SINIX 5.4 HD |
| 11         | 256          | 61          | 80       | SINIX 5.4 HD |

Der Formatcode 0x7F löst die automatische Formaterkennung (Autodetection) aus.

#### *dist*

gibt eine Distanz in KB an, die zu der durch *nr* ermittelten Adresse addiert wird. Wenn *dist* weggelassen wird, wird der Wert 0 angenommen.

#### *name*

bezeichnet den Namen des zu ladenden Programms in dem Dateisystem, dessen Startadresse durch *nr* und *dist* ermittelt wird. Ist *name* angegeben, sollte *dist* 0 sein. Wenn kein Name angegeben ist, wird das Programm direkt von der durch *nr* und *dist* bezeichneten Geräteadresse geladen. Bezeichnet *name* ein Dateiverzeichnis, so werden die Namen der darin enthaltenen Dateien angezeigt. Das Root-Verzeichnis wird durch den Namen "." angegeben.

#### *pari*

sind Parameter (s.u.) für das zu ladende Programm. Im Fall des SINIX-Kerns sind diese Parameter Zuweisungen der Form *vari=wert*. Für diese Zuweisungen kann es auch Einträge in der Standardwerte-Datei `/stand/boot` geben. Ist ein Parameter sowohl in der Standardwerte-Datei als auch in der Eingabezeile definiert, wird die Definition in der Eingabezeile benutzt.

### Parameter für den SINIX-Kern

Im Bootstring können Sie folgende Parameter für den SINIX-Kern angeben:

`root = gtyp(nr)`

legt fest, wo das Root-Dateisystem liegt. *gtyp* und *nr* werden vom Kern interpretiert. Die Angaben haben die gleiche Bedeutung wie oben beschrieben. Im Regelfall gilt `root=hd(1)` für einen Kern auf der Festplatte (erste Platten-Partition am Interphase Storage) und `root=ramd(0)` für einen Installationskern (Root-Dateisystem liegt in einem logisch als Platte verwalteten Hauptspeicherbereich). Für einen Kern auf der ersten Partition der Festplatte am Onboard-SCSI-Hostadapter gilt `root=shd(1)`.

`swap = gtyp(nr)`

legt den Swapbereich fest. Typisch sind `swap=hd(2)` bzw. `swap=shd(2)` für einen regulären Kern und `swap=ramd(1)` für einen Installationskern. `gtyp` und `nr` werden wie beschrieben vom Kern interpretiert.

`rootfstype = fstype`

informiert, welcher Dateisystem-Typ als Root-Dateisystem zu verwenden ist. Erlaubt sind "s5" und "ufs". `fstype` muß natürlich mit dem Typ des Dateisystems übereinstimmen, das als `root` angegeben ist.

`initargs = -s`

veranlaßt, daß der SINIX-Kern unabhängig von den in `/etc/inittab` angegebenen Standardvorgaben in den Single-User-Modus hochfährt.

## DATEIEN

`/stand/boot`            Datei mit Standardwerten zum Laden

## SIEHE AUCH

`fdisk(1M)`, `init(1M)`, `fd(7)`, `hd(7)`.

## colltbl(1M)

Die korrekte Pfadangabe für die Datei LC\_COLLATE muß lauten:

### DATEIEN

/usr/lib/locale/*sprache*/LC\_COLLATE

LC\_COLLATE-Datenbank für  
*sprache*

Alle weiteren Angaben der Beschreibung im Handbuch "Referenzhandbuch für Systemverwalter" [11] sind korrekt.

## edvtoc(1M)

### BEZEICHNUNG

edvtoc - Dienstprogramm zur Modifikation des VTOC einer Platte

### ÜBERSICHT

```
/etc/edvtoc -f vtoc_datei raw_gerät
```

### BESCHREIBUNG

Mit edvtoc kann der Benutzer das VTOC einer Platte (VTOC: volume table of contents, Inhaltsverzeichnis eines Datenträgers) ändern. Er muß dazu die folgenden drei Schritte ausführen: Zuerst muß er `prvtoc -f vtoc_datei` aufrufen (siehe `prvtoc(1M)`). `prvtoc` schreibt das aktuelle VTOC in die Datei `vtoc_datei`. Anschließend muß er diese Datei editieren und die gewünschten Änderungen am VTOC eingeben. Schließlich muß er `edvtoc -f vtoc_datei` aufrufen, damit die Änderungen übernommen werden.

edvtoc führt folgendes aus: Lesen/Interpretieren der Datei `vtoc_datei`, begrenzte Plausibilitätsprüfung des neuen VTOC, Anzeigen des neuen VTOC und Schreiben des VTOC auf Platte, sofern der Benutzer dies wünscht.

`-f vtoc_datei`

schreibt ein neues VTOC auf die aktive UNIX Partition. Das VTOC wird aus `vtoc_datei` aufgebaut. Der Benutzer muß `vtoc_datei` vorher mit `prvtoc` erstellt und mit einem Editor geändert haben. Jede Zeile der Datei `vtoc_datei` beschreibt eine Slice anhand von fünf Angaben: Slicenummer (`#slice`), Tag der Slice (`tag`), Flags der Slice (`flags`), Startsektor der Slice (`start`), Größe der Slice in Sektoren (`size`).

`raw_gerät`

Im Raw-Modus betriebenes Gerät für das gewünschte Plattenlaufwerk. Damit das gesamte Gerät angesprochen wird, muß das Gerät auf Slice 0 angegeben werden:

```
/dev/rdisk/crdys0
```

bei ESDI-Platten am Interphase Storageer und SCSI-Platten am Onboard-SCSI-Hostadapter NCR 53C710.

```
/dev/rdisk/shdxtys0
```

bei SCSI-Platten am NCR Host Adapter (ADP32 Controller).

**DATEIEN**

/dev/rdisk/crdys0  
/dev/rdisk/shdxtys0  
/etc/vfstab  
/etc/partitions

**SIEHE AUCH**

prvtoc(1M)

**Hinweis**

edvtoc zerstört i.a. den Platteninhalt und damit die Benutzerdaten. Zusätzlich zum VTOC müssen bei einer Änderung der Plattenaufteilung noch Dateien wie /etc/vfstab und /etc/partitions auf den neuesten Stand gebracht werden. Zur Plattenverwaltung ist deshalb diskadd besser geeignet.

## gated(1M)

### BEZEICHNUNG

gated - Routing-Dämon im Gateway

### ÜBERSICHT

gated [ *-t*[ierpuRH] ] [*protokolldatei*]

### BESCHREIBUNG

gated ist ein Routing-Dämon für die Abwicklung von mehreren Routing-Protokollen. Er ersetzt routed(1M) und sonstige Routing-Dämonen, die die Routing-Protokolle HELLO und EGP verstehen. Derzeit wickelt gated die Routing-Protokolle RIP, EGP und HELLO ab. gated kann so konfiguriert werden, daß entweder alle Routing-Protokolle oder eine beliebige Kombination der drei Protokolle ausgeführt werden. Die Konfiguration von gated ist standardmäßig in der Datei `/etc/inet/gated.conf` gespeichert.

### Protokollierungsoptionen in der Aufrufzeile

gated kann mit einer Reihe von Protokollierungsoptionen und/oder einer Protokolldatei aufgerufen werden. Protokollierungsoptionen können auch mit der Klausel `traceflags` in der Konfigurationsdatei festgelegt werden. gated trennt sich von dem steuernden Terminal, außer es wurden Protokollierungsoptionen aber keine Protokolldatei angegeben; in diesem Fall wird die gesamte Protokollierung am steuernden Terminal ausgegeben. Folgende Protokollierungsoptionen sind gültig:

`-t`

Alle Fehlermeldungen, Routenänderungen und gesendeten bzw. empfangenen EGP-Pakete protokollieren, wenn keine anderen Optionen angegeben sind. Ist nur die Option `-t` angegeben, werden automatisch die Protokollierungsoptionen `i`, `e`, `r` und `p` eingeschaltet. In Verbindung mit anderen Optionen ist `-t` wirkungslos, und es werden lediglich die begleitenden Optionen erkannt.

Beachten Sie, daß die anderen Optionen nur in Kombination mit `-t` verwendet werden dürfen!

`i`

Alle internen Fehler und internen Routingfehler protokollieren.

`e`

Alle von EGP verursachten externen Fehler, alle externen Routingfehler und EGP-Zustandswechsel protokollieren.

- r Alle Routenänderungen protokollieren.
- p Alle gesendeten und empfangenen EGP-Pakete protokollieren.
- u In Verbindung mit p, R, H oder N den gesamten Inhalt der gesendeten und empfangenen Routingpakete anzeigen.
- R Alle gesendeten oder empfangenen RIP-Pakete protokollieren.
- H Alle gesendeten oder empfangenen HELLO-Pakete protokollieren.

gated protokolliert in jedem Fall die nicht behebbaren Fehler. Ist keine Protokolldatei angegeben und sind keine Protokollierungsoptionen gesetzt, werden alle Meldungen an `/dev/null` gesendet.

### Signalverarbeitung

gated fängt eine Reihe von Signalen auf und führt bestimmte Aktionen durch. Derzeit führt gated eine spezielle Verarbeitung der Signale `SIGHUP`, `SIGINT` und `SIGUSR1` durch.

Wird `SIGHUP` an gated gesendet und wird gated mit Protokollierungsoptionen und einer Protokolldatei aufgerufen, wird die Protokollierung abgeschaltet und die Protokolldatei geschlossen. Die Protokolldatei kann zu diesem Zeitpunkt versetzt/umbenannt oder gelöscht werden. Das nächste `SIGHUP` an gated schaltet die Protokollierung wieder ein. gated liest die Konfigurationsdatei und setzt die Protokollierungsoptionen entsprechend der Einstellungen der Klausel `traceflags`. Ist keine Klausel `traceflags` eingetragen, wird die Protokollierung entsprechend der in der Aufrufzeile angegebenen Protokollierungsoptionen wieder aufgenommen. Wenn die in der Aufrufzeile angegebene Protokolldatei nicht existiert, wird sie angelegt, und das Protokoll wird dann in die genannte Datei geschrieben. Existiert sie schon, wird das Protokoll daran angehängt. Nützlich ist dies bei zyklischen Protokolldateien, z.B. in Verbindung mit dem `syslog(1M)`-Dämon.

Wenn gated das Signal `SIGINT` erhält, wird festgelegt, daß innerhalb der nächsten 60 Sekunden ein Speicherabzug in die Datei `/usr/tmp/gated_dump` geschrieben werden muß. Bevor gated den Speicherabzug erstellt, arbeitet er alle noch ausstehenden Routenänderungen ab. Der Speicherabzug enthält eine Momentaufnahme des aktuellen Zustands von gated, einschließlich der Schnittstellenkonfigurationen, des Zustands des EGP-Nachbars und der Routing-Tabellen.

Ist die Datei `/usr/tmp/gated_dump` bereits vorhanden, wird der Speicherabzug an die bestehende Datei angehängt.

Beim Empfang von SIGUSR1 liest gated erneut bestimmte Angaben in der Konfigurationsdatei. Diese Angaben sind derzeit `announcetoAS`, `noannouncetoAS` und `validAS`. Werden keine Fehler festgestellt, werden die neuen Konfigurationsangaben umgesetzt, werden jedoch Fehler festgestellt, werden die Konfigurationsangaben nicht geändert. Beim Empfang von SIGUSR1 prüft gated auch den Zustand der Netzwerk-Schnittstellen.

### Optionen in der Konfigurationsdatei zur Steuerung der Protokollausgabe

`traceflags prot_option [prot_option] [prot_option] ...`

Diese Klausel teilt gated mit, welche Stufe der Protokollausgabe gewünscht wird. Sie wird während der Initialisierung von gated und bei jedem Empfang des Signals SIGHUP durch gated gelesen. Sie wird zum Zeitpunkt der Initialisierung ignoriert, wenn in der Aufrufzeile Protokollierungsoptionen angegeben sind. Folgende Protokollierungsoptionen sind gültig:

`internal`

Alle internen Fehler und internen Routingfehler protokollieren.

`external`

Alle von EGP verursachten externen Fehler, alle externen Routingfehler und EGP-Zustandswechsel protokollieren.

`route`

Alle Routenänderungen protokollieren.

`egp`

Alle gesendeten und empfangenen EGP-Pakete protokollieren.

`update`

In Verbindung mit `egp`, `rip` oder `hello` den gesamten Inhalt der gesendeten und empfangenen Routingpakete anzeigen.

`rip`

Alle gesendeten oder empfangenen RIP-Pakete protokollieren.

`hello`

Alle gesendeten oder empfangenen HELLO-Pakete protokollieren.

`icmp`

Alle empfangenen ICMP-REDIRECT-Pakete protokollieren.

`stamp`

Alle 10 Minuten in der Protokolldatei einen Zeitstempel eintragen.

`general`

Kombination aus `internal`, `external`, `route` und `egp`.

all

Alle oben genannten Protokollierungsoptionen aktivieren.

Wird mehr als eine Klausel `traceflags` verwendet, werden die Protokollierungsoptionen akkumuliert.

### Standardkonfiguration

In der Regel liest `gated` die eingetragenen Konfigurationsangaben in der Konfigurationsdatei, die bei der Compilierung angegeben wurde. Ist diese Datei nicht vorhanden, verwendet `gated` folgende Standardwerte:

```
RIP yes
HELLO no
EGP no
```

Wenn die Konfigurationsdatei nicht vorhanden ist, wenn darüber hinaus nur eine Netzwerkschnittstelle existiert und im Kernel eine Standardroute installiert ist, dann beendet sich `gated` in der Annahme, daß eine einfache Standardroute ausreicht.

### Optionen in der Konfigurationsdatei zur Abwicklung von Routing-Protokollen

In diesem Abschnitt werden die zahlreichen Konfigurationsoptionen erläutert. Bei jedem Starten liest `gated` die Datei `/etc/inet/gated.conf`, um festzustellen wie das Routing bei jedem Protokoll abzuwickeln ist.

Die folgenden Konfigurationsoptionen stehen zur Verfügung:

```
RIP {yes | no | supplier | pointopoint | quiet | gateway #}
```

Damit wird `gated` die Ausführung des Routing-Protokolls `RIP` angezeigt. Nach dem Schlüsselwort `RIP` ist nur eins der oben genannten `RIP`-Argumente zulässig. Ist mehr als eins angegeben, wird nur das erste erkannt.

Es folgt eine Liste der in der `RIP`-Klausel möglichen Argumente:

yes

Das `RIP`-Protokoll ausführen. Alle ankommenden `RIP`-Pakete verarbeiten und alle 30 Sekunden `RIP`-Informationen nur dann verteilen, wenn mindestens zwei Netzwerkschnittstellen vorhanden sind.

no

Das `RIP`-Protokoll nicht ausführen. `RIP` nicht ausführen.

supplier

Das `RIP`-Protokoll ausführen. Alle ankommenden `RIP`-Pakete verarbeiten und alle 30 Sekunden unabhängig von der Zahl der vorhandenen Netzwerkschnittstellen die `RIP`-Informationen verteilen.

**pointopoint**

Das RIP-Protokoll ausführen. Alle ankommenden RIP-Pakete verarbeiten und alle 30 Sekunden unabhängig von der Zahl der vorhandenen Netzwerkschnittstellen die RIP-Informationen verteilen. Bei Angabe dieses Arguments wird die RIP-Information nicht in einem Nachrichtenpaket gesendet. Die RIP-Information wird direkt an die Gateways gesendet, die bei der unten beschriebenen Option `sourceripgateways` aufgelistet sind.

**quiet**

Alle ankommenden RIP-Pakete verarbeiten, aber unabhängig von der Zahl der vorhandenen Netzwerkschnittstellen keine RIP-Information weiterleiten.

**gateway #**

Alle ankommenden RIP-Pakete verarbeiten, alle 30 Sekunden die RIP-Information verteilen und die Standardroute (0.0.0.0) mit einem metrischen Wert von # bekanntmachen. Der metrische Wert sollte ein Wert sein, der eine Anzahl von RIP-Knoten widerspiegelt. Ist diese Option gesetzt, werden alle anderen Standardrouten von anderen RIP-Gateways ignoriert. Die Standardroute wird nur bekanntgemacht, wenn wenigstens mit einem EGP-Nachbar eine Verbindung aufgenommen wird. Deshalb sollte diese Route nur bei Einsatz von EGP verwendet werden.

Ist keine RIP-Klausel eingetragen, wird RIP nicht ausgeführt.

HELLO {yes | no | supplier | pointopoint | quiet | gateway #}

Damit wird `gated` mitgeteilt, wie das Routing-Protokoll HELLO auszuführen ist. Die Argumente gleichen bis auf kleine Abweichungen den RIP-Argumenten. Nach dem Schlüsselwort HELLO ist nur eins der oben genannten HELLO-Argumente zulässig. Ist mehr als eins angegeben, wird nur das erste erkannt. Es folgt eine Liste der in der HELLO-Klausel möglichen Argumente:

**yes**

Das HELLO-Protokoll ausführen. Alle ankommenden HELLO-Pakete verarbeiten und alle 15 Sekunden HELLO-Informationen nur dann verteilen, wenn mindestens zwei Netzwerkschnittstellen vorhanden sind.

**no**

Das HELLO-Protokoll nicht ausführen. HELLO nicht ausführen.

**supplier**

Das HELLO-Protokoll ausführen. Alle ankommenden HELLO-Pakete verarbeiten und alle 15 Sekunden unabhängig von der Zahl der vorhandenen Netzwerkschnittstellen die HELLO-Informationen verteilen.

**pointopoint**

Das HELLO-Protokoll ausführen. Alle ankommenden HELLO-Pakete verarbeiten und alle 15 Sekunden unabhängig von der Zahl der vorhandenen Netzwerkschnittstellen die HELLO-Informationen verteilen. Bei Angabe dieses Arguments wird die HELLO-Information nicht in einem Nachrichtenpaket gesendet. Die HELLO-Information wird direkt an die Gateways gesendet, die bei der unten beschriebenen Option `sourcehellogateways` aufgelistet sind.

**quiet**

Alle ankommenden HELLO-Pakete verarbeiten, aber unabhängig von der Zahl der vorhandenen Netzwerkschnittstellen keine HELLO-Information verteilen.

**gateway #**

Alle ankommenden HELLO-Pakete verarbeiten, alle 15 Sekunden die HELLO-Information verteilen und die Standardroute (0.0.0.0) mit einer Zeitverzögerung von # bekanntmachen. Die Zeitverzögerung sollte in Millisekunden angegeben werden. Die Standardroute wird nur bekanntgemacht, wenn wenigstens mit einem EGP-Nachbar eine Verbindung aufgenommen wird. Deshalb sollte diese Route nur bei Einsatz von EGP verwendet werden.

Ist keine HELLO-Klausel eingetragen, wird HELLO nicht ausgeführt.

EGP {yes | no}

Mit dieser Klausel kann die Verarbeitung von EGP durch `gated` aktiviert bzw. deaktiviert werden.

**no**

Keine EGP-Verarbeitung ausführen.

**yes**

Alle EGP-Operationen durchführen.

**Hinweis**

Die EGP-Verarbeitung erfolgt standardmäßig. Wenn keine EGP-Klausel eingetragen ist, werden alle EGP-Operationen ausgeführt.

**autonomoussystem #**

Wenn das EGP-Protokoll ausgeführt wird, ist mit dieser Klausel die Nummer (#) des autonomen Systems anzugeben. Ohne Angabe beendet sich `gated` und zeigt einen nicht behebbaren Fehler an.

**egpmaxacquire #**

Wenn das EGP-Protokoll ausgeführt wird, gibt diese Klausel die Anzahl der EGP-Knoten an, mit denen `gated` das EGP abwickelt. Diese Zahl muß größer Null und kleiner gleich

der Zahl der angegebenen EGP-Nachbarn sein; sonst beendet sich `gated`.

```
egpneighbor gateway
 metricin wert
 egpmetricout egp_wert
 ASin as_ein
 ASout as_aus
 AS as
 nogendefault
 acceptdefault
 defaultout egp_wert
 validate
 intf schnittstelle
 sourcenet netz
 gateway gateway
```

Wenn das EGP-Protokoll ausgeführt wird, wird in dieser Klausel der Rechner genannt, mit dem `gated` das EGP abwickelt. `gateway` kann ein symbolischer in `/etc/hosts` stehender Name oder ein in Punktschreibweise (a.b.c.d) dargestellter IP-Hostname sein. Die Punktschreibweise empfiehlt sich, damit Verwechslungen vermieden werden. Die EGP-Nachbarn werden in der Reihenfolge angesprochen, in der sie in der Konfigurationsdatei gelistet sind.

Mit der Option `metricin` wird die interne Zeitverzögerung angegeben, die als metrischer Wert für alle Routen verwendet werden soll, die von diesem Nachbarn gelernt wurden. Die Zeitverzögerung sollte als Wert von 0 bis 30000 angegeben werden. Wird weder diese Option noch die `validate`-Option verwendet, gilt als interner metrischer Wert die EGP-Distanz multipliziert mit 100.

Mit der Option `egpmetricout` wird die EGP-Distanz angegeben, die für alle Netze, die diesem Nachbarn bekanntgemacht wurden, verwendet wird. Als EGP-Distanz sollte ein Wert von 0 bis 255 eingetragen werden. Ist diese Option nicht eingetragen, wird die interne Zeitverzögerung für jede Route in eine EGP-Distanz umgerechnet, indem sie durch 100 dividiert wird; dabei werden Distanzen größer 255 auf 255 gesetzt.

Mit der Option `ASin` wird die Nummer des autonomen Systems dieses Nachbarn geprüft. Stimmt die Nummer des autonomen Systems, die in den Nachbarakquisitionspaketen angegeben wurde, nicht mit `ASin` überein, wird eine Fehlermeldung ausgegeben und der Verbindungsaufbau verweigert. Ist diese Option nicht eingetragen, werden die Nummern der autonomen Systeme nicht geprüft.

Mit der Option `ASout` wird die Nummer des autonomen Systems in EGP-Paketen angegeben, die diesem Nachbarn gesendet werden. Ist sie nicht angegeben, wird das autonome System aus der Klausel `autonomoussystem` verwendet. In der Regel sollte diese Klausel nicht verwendet werden, da sie für den Sonderfall einer Verbindung zwischen ARPAnet und NSFnet reserviert ist.

Mit der Option `AS` wird die Nummer des autonomen Systems angegeben, das den Routen zugeordnet wird, die von diesem Nachbarn gelernt wurden. Ist sie nicht angegeben, wird das autonome System in den EGP-Paketen, die von diesem Nachbarn empfangen

wurden, verwendet. In der Regel sollte diese Klausel nicht verwendet werden, da sie für den Sonderfall einer Verbindung zwischen ARPAnet und NSFnet reserviert ist.

Mit der Option `nogendefault` wird angegeben, daß dieser Nachbar für die interne Generierung einer Standard-Route bei Verwendung von `RIP gateway` oder `HELLO gateway` nicht berücksichtigt werden soll. Ist sie nicht eingetragen, wird bei aktiven Partnerverbindungen ("peer-to-peer") mit diesem Nachbarn die interne Standard-Route generiert.

Mit der Option `acceptdefault` wird angegeben, daß die Standardroute (Netz 0.0.0.0) als gültig angesehen werden soll, wenn sie von diesem Nachbarn empfangen wird. Ist diese Option nicht eingetragen, führt der Empfang der Standardroute dazu, daß eine Warnmeldung ausgegeben wird und die Route ignoriert wird.

Mit der Option `defaultout` wird angegeben, daß die intern generierte Standard-Route an diesen EGP-Nachbarn in der bezeichneten Distanz weitergeleitet werden darf. Die Distanz sollte als EGP-Distanz mit einem Wert von 0 bis 255 eingetragen werden. Eine Standardroute, die von einem anderen Gateway gelernt wurde, wird an einen EGP-Nachbar nicht weitergeleitet. In der Regel wird über EGP keine Standardroute weitergeleitet. Die Option `acceptdefault` sollte bei Verwendung der Option `defaultout` nicht eingetragen werden. Der Wert, der in der Option `egpmetricout` angegeben wurde, wird ignoriert, da für die Standardroute stets der metrische Wert, der in der Option `defaultout` angegeben wurde, verwendet wird.

Mit der Option `validate` wird angegeben, daß alle Netze, die von diesem EGP-Nachbarn empfangen werden, in der Klausel `validAS` eingetragen sein müssen. Diese Klausel gibt ebenfalls das autonome System dieses Nachbarn an. Netze ohne Klausel `validAS` werden ignoriert, und eine Warnmeldung wird ausgegeben.

Mit der Option `intf` wird die Schnittstelle angegeben, über die die EGP-Pakete an diesen Nachbarn gesendet werden. Diese Option wird nur benötigt, wenn es mit diesem EGP-Nachbarn kein gemeinsames Netz/Teilnetz gibt. Derzeit wird diese Option lediglich für Testzwecke eingesetzt, und der Eintrag garantiert nicht ohne weiteres einen korrekten Betrieb bei einer Verbindung mit einem EGP-Nachbarn ohne gemeinsames Netz/Teilnetz eine Verbindung.

Mit der Option `sourcenet` wird das Ursprungsnetzwerk angegeben, das in den EGP-Poll-Paketen enthalten sein muß, die an diesen Nachbarn gesendet werden. Ist diese Option nicht angegeben, wird das Netz (nicht das Teilnetz) der Schnittstelle für die Kommunikation mit diesem Nachbarn verwendet. Derzeit wird diese Option lediglich für Testzwecke eingesetzt, und der Eintrag bedeutet nicht von vornherein einen korrekten Betrieb.

Mit der Option `gateway` wird das Gateway angegeben, das verwendet werden soll, wenn Routen, die von einem EGP-Nachbar in einem anderen Netz gelernt wurden, installiert werden. In der Regel werden diese Routen ignoriert. Derzeit wird diese Option lediglich für Testzwecke eingesetzt, und sie wird wohl nicht in allen Fällen einwandfrei funktionieren.

## Optionen in der Konfigurationsdatei zur Abwicklung von Routing-Informationen

Mit den folgenden Konfigurationsdateioptionen wird `gated` mitgeteilt, wie ankommende und abgehende Routing-Informationen zu behandeln sind.

```
trustedrippgateways gateway [gateway] [gateway]
trustedhellogateways gateway [gateway] [gateway]
```

Werden diese Klauseln angegeben, akzeptiert `gated` nur RIP- oder HELLO-Informationen von den entsprechenden RIP- oder HELLO-Gateways. `gateway` kann ein symbolischer in `/etc/hosts` stehender Name oder ein in Punktschreibweise (a.b.c.d) dargestellter IP-Hostname sein. Wiederum empfiehlt sich die Punktschreibweise, damit Verwechslungen vermieden werden.

### Hinweis

Die Verteilung von Routing-Informationen wird durch diese Klausel nicht eingeschränkt.

```
sourceripgateways gateway [gateway] [gateway]
sourcehellogateways gateway [gateway] [gateway]
```

`gated` sendet RIP- oder HELLO-Informationen direkt an die angegebenen Gateways. Wurde in den oben erwähnten RIP- oder HELLO- Klauseln die Option `pointopoint` eingetragen, sendet `gated` die RIP- oder HELLO-Information nur an die angegebenen Gateways. `gated` sendet KEINE Informationen über die Broadcast-Adresse. Wurde in diesen Klauseln die Option `pointopoint` nicht eingetragen und verteilt `gated` RIP- oder HELLO-Informationen, sendet `gated` die Information an die angegebenen Gateways und verteilt sie als Broadcast-Information über die Broadcast-Adresse.

```
noripoutinterface schn_adr [schn_adr] [schn_adr]
nohellooutinterface schn_adr [schn_adr] [schn_adr]
noripfrominterface schn_adr [schn_adr] [schn_adr]
nohellofrominterface schn_adr [schn_adr] [schn_adr]
```

Die genannten Klauseln aktivieren bzw. deaktivieren die Protokolle jeweils pro Schnittstelle (`schn_adr`). `no{rip|hello}frominterface` bedeutet, daß RIP- oder HELLO-Informationen, die von einem anderen Gateway an die aufgeführten Schnittstellen gelangen, nicht akzeptiert werden. `no{rip|hello}outinterface` bedeutet, daß RIP- oder HELLO-Informationen nicht von den aufgeführten Schnittstellen abgesendet werden. `intfaddr` sollte in Punktschreibweise eingetragen sein.

```
passiveinterfaces schn_adr [schn_adr] [schn_adr]
```

Zur dynamischen Bestimmung, ob eine Schnittstelle ordnungsgemäß arbeitet, deaktiviert `gated` eine Schnittstelle nach einer bestimmten Zeitspanne, wenn an dieser Schnittstelle keine RIP-, HELLO- oder EGP-Pakete empfangen werden. PSN-Schnittstellen senden ein

RIP- oder HELLO-Paket an sich selbst, um festzustellen, ob die Schnittstelle ordnungsgemäß arbeitet, da die Verzögerung zwischen den EGP-Paketen länger als Deaktivierungszeitspanne der Schnittstelle sein kann. Die Routen von Schnittstellen, die nach dieser Zeitspanne deaktiviert wurden, werden automatisch wieder installiert, wenn Routing-Informationen wieder über diese Schnittstelle empfangen werden. Die genannte Klausel hindert `gated` daran, die aufgeführten Schnittstellen nach einer Zeitspanne zu deaktivieren, da sie als stets funktionstüchtig angesehen werden. Falls `gated` keine RIP- oder HELLO-Information verteilt, werden sämtliche Schnittstellen nicht gealtert, und die Option `passiveinterfaces` gilt automatisch für alle Schnittstellen.

`interfacemetric schn_adr metr_wert`

Mit dieser Klausel kann ein schnittstellenspezifischer metrischer Wert für die angegebene Schnittstelle definiert werden. Bei Systemen, die die schnittstellenspezifischen metrischen Werte unterstützen, wird dieser Klausel Priorität vor dem metrischen Wert des Kernels gegeben. Bei Systemen, die die schnittstellenspezifischen metrischen Werte nicht unterstützen, kann bei dieser Klausel ein Ersatzwert angegeben werden. Der schnittstellenspezifische metrische Wert wird dem tatsächlichen metrischen Wert hinzuaddiert, der in Routing-Information für jede Route enthalten ist, die von der aufgeführten Schnittstelle ankommt. Der schnittstellenspezifische metrische Wert wird ebenfalls dem tatsächlichen metrischen Wert jeder Information hinzuaddiert, die über die aufgeführte Schnittstelle abgesendet wird. Der metrische Wert der direkt angeschlossenen Schnittstellen wird ebenfalls auf den schnittstellenspezifischen metrischen Wert gesetzt; Routing-Information, die über direkt angeschlossene Netzen verteilt wird, basiert ebenfalls auf dem angegebenen schnittstellenspezifischen metrischen Wert. Diese Klausel wird für jede Schnittstelle benötigt, für die ein schnittstellenspezifischer metrischer Wert gewünscht wird.

`reconstmetric schn_adr metr_wert`

Es handelt sich hier um einen ersten Versuch, `gated` mit Anhaltspunkten für Reserve-Routing zu versorgen. Wenn die oben genannte Klausel verwendet wird, wird der metrische Wert der Routen aus jeder RIP-Information, die über die aufgeführte Schnittstelle empfangen wird, auf den mit `metric#` festgesetzten Wert gesetzt. Die Veränderung des metrischen Werts sollte umsichtig vorgenommen werden, da sie zu Routen-Loops führen kann.

### Vorsicht

Sie müssen hier mit äußerster Sorgfalt vorgehen! Jede Route, dessen metrischer Wert gleich unendlich ist, wird nicht wiederhergestellt, sondern der Wert bleibt unendlich.

```
fixedmetric sch_n_adr proto {riplhello} metr_wert
```

Es handelt sich hier um einen weiteren Versuch, `gated` mit Anhaltspunkten für Reserve-Routing zu versorgen. Wenn die oben genannte Klausel verwendet wird, erhalten alle Routing-Informationen, die von der aufgeführten Schnittstelle abgehen, den in `metric#` festgesetzten Wert. Bei RIP ist der metrische Wert als Anzahl der RIP-Knoten von 0 bis unendlich festzusetzen. Bei HELLO ist der metrische Wert als HELLO-Verzögerung in Millisekunden von 0 bis unendlich festzusetzen. Jede Route, die einen metrischen Wert unendlich hat, behält diesen Wert. Gehen Sie bei festen metrischen Werten mit äußerster Sorgfalt vor.

```
donotlisten netz intf adr [adr] ... proto {riplhello}
donotlistenhost host intf adr [adr] ... proto {riplhello}
```

Erläuterung dieser Klausel: das Schlüsselwort `donotlisten`, dann eine Netzwerknummer (Punktschreibweise empfohlen), dann das Schlüsselwort `intf`. Es folgt eine Liste der Schnittstellen in Punktschreibweise, dann das Schlüsselwort `proto`, und zuletzt `rip` oder `hello`.

Dies bedeutet, daß jede Information mit Bezug auf `net`, die über die angegebenen Protokolle UND von den aufgeführten Schnittstellen eingeht, ignoriert wird. Das Schlüsselwort `all` kann an das Schlüsselwort `intf` angefügt werden, um alle Schnittstellen in der Maschine anzugeben.

### Beispiele

```
donotlisten 10.0.0.0 intf 128.84.253.200 proto rip
```

Erläuterung: Jede RIP-Information mit Bezug auf Netz 10.0.0.0, die über die Schnittstelle 128.84.253.200 eintrifft, wird ignoriert. Es wird eine Klausel für jedes Netz benötigt, auf dem diese Einschränkung gelten soll.

```
donotlisten 26.0.0.0 intf all proto rip hello
```

Erläuterung: Jede RIP- oder HELLO-Information mit Bezug auf Netz 26.0.0.0, die über eine beliebige Schnittstelle eintrifft, wird ignoriert.

Die Beschreibung von `donotlistenhost` entspricht größtenteils der Beschreibung von `donotlisten`. Der Unterschied liegt darin, daß statt der Netzwerkadresse eine Host-Adresse angegeben wird. Die oben genannten Einschränkungen gelten für die angegebene Host-Route, die vom aufgeführten Routing-Protokoll erkannt wurde.

```
listen netz gateway adr [adr] ... proto {riplhello}
listenhost host gateway adr [adr] ... proto {riplhello}
```

Erläuterung dieser Klausel: das Schlüsselwort `listen`, dann eine Netzwerknummer (Punktschreibweise empfohlen), dann das Schlüsselwort `gateway`. Es folgt eine Liste der Gateways in Punktschreibweise, dann das Schlüsselwort `proto`, und zuletzt `rip` oder `hello`.

Akzeptiert werden in diesem Fall nur Information mit Bezug auf das Netz `netz`, und auch nur dann, wenn sie von den aufgeführten `gateways` über die angegebenen Protokolle kommen.

```
listen 128.84.0.0 gateway 128.84.253.3 proto hello
```

Erläuterung: Jede HELLO-Information mit Bezug auf das Netz `128.84`, die über Gateway `128.84.253.3` eingeht, wird akzeptiert. Jede andere Information zu `128.84` von jedem anderem Gateway wird abgewiesen. Für jedes Netz, für das die Beschränkung gelten soll, ist eine Klausel erforderlich.

```
listenhost 26.0.0.15 gateway 128.84.253.3 proto rip
```

Erläuterung: Jede Information zu Host `26.0.0.15` muß über RIP und von Gateway `128.84.253.3` eingehen. Alle anderen Informationen zu diesem Host werden ignoriert.

```
announce netz intf adr [adr] ... proto typ [egpmetric #]
announcehost host intf adr ... proto typ [egpmetric #]
noannounce netz intf adr [adr] ... proto typ [egpmetric #]
noannouncehost host intf adr ... proto typ [egpmetric #]
```

Mit diesen Klauseln können für die angegebenen Netze und Hosts und für ein bestimmtes Protokoll Einschränkungen festgelegt werden. Die Klauseln `announce{host}` und `noannounce{host}` können bei derselben Schnittstelle nicht zusammen verwendet werden. Bei der Klausel `announce{host}` verteilt `gated` nur die Netze oder Hosts, die eine zugehörige Klausel `announce{host}` mit dem entsprechenden Protokoll haben. Bei der Klausel `noannounce{host}` verteilt `gated` alles, *ausgenommen* jene Netze oder Hosts, die eine zugehörige Klausel `noannounce{host}` haben. Dadurch läßt sich pro Schnittstelle festlegen, daß entweder nur die in der `announce`-Liste eingetragenen Routen verteilt werden oder daß alle Netze verteilt werden, mit Ausnahme der Netze, die in der `noannounce`-Liste stehen.

Die Argumente sind mit denen in der Klausel `donotlisten` identisch, ausgenommen `egp`, das im Feld `proto` angegeben werden kann.

`typ` kann `rip`, `hello`, `egp` oder eine beliebige Kombination davon sein. Ist im Feld `proto` `egp` angegeben, muß der metrische Wert von `egp` eingetragen werden. Es handelt sich um einen metrischen Wert, bei dem `gated` das aufgeführte Netz über EGP bekanntmacht.

**Hinweis**

Es handelt sich nicht um statische Routeneinträge. Diese Einschränkungen gelten nur, wenn das Netz oder der Host über eines der Routing-Protokolle gelernt wird. Ist ein eingeschränktes Netz plötzlich unerreichbar und meldet sich ab, wird es erst wieder verteilt, wenn es erneut gelernt wird.

Derzeit kann pro Netz oder Host nur eine Klausel `announce{host}` bzw. `noannounce{host}` angegeben werden. Es ist nicht möglich, ein Netz oder einen Host über HELLO durch eine Schnittstelle und über RIP durch eine andere bekanntzumachen.

**Beispiele**

```
announce 128.84 intf all proto rip hell egp egpmetric 0
announce 10.0.0.0 intf all proto rip
announce 0.0.0.0 intf 128.84.253.200 proto rip
announce 35.0.0.0 intf all proto rip egp egpmetric 3
```

Wenn nur diese vier `announce`-Klauseln in der Konfigurationsdatei enthalten sind, verteilt `gated` nur diese vier Netze. `gated` verteilt 128.84.0.0 über RIP und HELLO an alle Schnittstellen und verteilt es über EGP mit einem metrischen Wert von 0. Netz 10.0.0.0 wird über RIP an alle Schnittstellen verteilt. Netz 0.0.0.0 (Standard) wird über RIP nur an die Schnittstelle 128.84.253.200 verteilt. Netz 35.0.0.0 wird über RIP an alle Schnittstellen verteilt und über EGP mit einem metrischen Wert von 3. Es handelt sich um die einzigen Netze, die über einen Broadcast von diesem Gateway verteilt werden. Ist die erste Klausel `announce` angegeben, werden nur die Netze mit den Klauseln `announce` mit Broadcast verteilt. Dies schließt auch lokale Teilnetze ein. Steht `all` nach `intf` bei `announce{host}` oder `noannounce{host}`, wird diese Klausel global angewendet und die Option mit den Einschränkungen jeweils pro Schnittstelle geht verloren. Sind Einschränkungen der Routen-Verteilung nicht erwünscht, sollten `announce`-Klauseln nicht verwendet werden. Alle erkannten Informationen werden weitergeleitet.

**Hinweis**

Dies hat keine Auswirkungen auf Informationen, die `gated` akzeptiert. Jedes Netz, das keine Klausel `announce` hat, wird dennoch in den Routing-Tabellen des Kernels eingetragen, aber nicht über eines der Routing-Protokolle verteilt. Sollen Netze im Kernel nicht eingetragen werden, kann die Klausel `donotlisten` verwendet werden.

```
announce 128.84 intf 128.59.2.1 proto rip
noannounce 128.84 intf 128.59.1.1 proto rip
```

Erläuterung: An Schnittstelle 128.59.2.1 wird über RIP nur die Information zu 128.84.0.0 verteilt, aber an Schnittstelle 128.59.1 werden über RIP sämtliche Informationen verteilt, mit Ausnahme der Informationen zu 128.84.0.0.

```
noannounce 128.84 intf all proto rip hello egp egpmetric 0
noannounce 10.0.0.0 intf all proto hello
```

**Erläuterung:** Außer den beiden angegebenen Netzen, werden alle Netze verteilt. Insbesondere werden überhaupt keine Informationen zum Netz 128.84.0.0 gesendet: alle Schnittstellen und alle Protokolle sind explizit ausgeschlossen. Die Information von 128.84.0.0 wird nicht versendet. Netz 10.0.0.0 wird an keine Schnittstelle über HELLO verteilt. Dies bedeutet auch, daß das Netz 10.0.0.0 über RIP jeder Schnittstelle bekanntgemacht wird. Dieses Netz wird auch über einen Broadcast über EGP verteilt, wobei der entsprechende metrische Wert in der Klausel `defaultegpmetric` verwendet wird.

`defaultegpmetric wert`

Dieser metrischer EGP-Wert soll als Standardwert gelten, wenn es keine Routen-Einschränkungen gibt. Wenn keine Routen-Einschränkungen bestehen, werden in der Regel alle Netze, die über HELLO oder RIP gelernt wurden, mit dem hier angegebenen standardmäßigen metrischen EGP-Wert über EGP verteilt. Wird diese Klausel nicht verwendet, wird der standardmäßige metrische EGP-Wert auf 255 gesetzt, wodurch eine über EGP verteilte Route dieser Art ignoriert wird. Gibt es keine Routing-Einschränkungen, wird jedes Netz mit einer direkten Schnittstelle über EGP und einem metrischen Wert von 0 verteilt.

#### *Hinweis*

Dies schließt nicht die Teilnetze ein, sondern nur das Netz mit seinen Teilnetzen.

```
defaultgateway gateway proto [metr_wert metr_wert] {active|passive}
```

Dieses Standard-Gateway wird in den Routing-Tabellen des Kernels während der Initialisierung installiert und wird immer dann neu installiert, wenn Informationen über die Standardroute verloren gehen. Installiert wird diese Route mit der Zeitverzögerung, die dem metrischen RIP-Wert 15 entspricht, außer ein anderer metrischer Wert wurde in der Option für den metrischen Wert angegeben.

Bei Verwendung von `RIP gateway` oder `HELLO gateway` wird diese Standardroute gelöscht, wenn eine erfolgreiche Verbindung mit einem EGP-Nachbar aufgebaut wurde, der bei `nogendefault` nicht angegeben wurde.

Eine aktive Standardroute wird von einer anderen Standardroute, die über ein anderes Routing-Protokoll erkannt wurde, außer Kraft gesetzt. Eine `passive` Standardroute wird nur von einer Standardroute mit einem niedrigeren metrischen Wert außer Kraft gesetzt.

Eine aktive Standardroute wird bei Routingaktualisierungen nicht weitergeleitet, eine `passive` Standardroute wird weitergeleitet.

`gateway` sollte eine Adresse in Punktschreibweise sein. `metric` ist optional und sollte

im angegebenen Protokoll ein metrischer Wert zwischen 0 und unendlich sein. Ist kein Wert angegeben, wird ein metrischer RIP-Wert von 15 verwendet. `proto` sollte entweder `rip`, `egp` oder `hello` sein. Das Feld `proto` initialisiert das Protokoll, von dem die Route gelernt wurde. Obwohl das Feld in diesem Fall nicht benutzt wird, blieb es aus Gründen der Konsistenz erhalten.

```
net netz_adr gateway adr metric hop_zahl {rip|egp|hello}
host host_adr gateway adr metric hop_zahl {rip|egp|hello}
```

Die nächsten Klauseln installieren eine statische Route zum Netz `netz_adr` oder zum Host `host_adr` über den Gateway `adr` mit einem metrischen Wert `hop_zahl`, der über RIP, HELLO oder EGP gelernt wurde. In der Regel empfiehlt sich Punkt Schreibweise für die Adressen. Diese Route wird in der Routing-Tabelle des Kernels installiert und wird von RIP- oder HELLO-Bekanntmachungen anderer Gateways nicht beeinträchtigt. Das Protokoll, über das sie erkannt wurde, ist wichtig, wenn die Route über EGP bekanntgemacht werden soll. Ist das Protokoll `rip` oder `hello` und gibt es keine Routing-Einschränkungen, dann wird diese Route von EGP mit dem metrischen Wert von `defaultegpmetric` verteilt. Ist das Protokoll `egp` und gibt es keine Routing-Einschränkungen, dann wird diese Route von EGP mit dem metrischen Wert von `hopcnt` verteilt.

```
egpnetsreachable net [net] [net]
```

Diese Option wurde als "weiche Einschränkung" beibehalten. Sie kann nicht angegeben werden, wenn die Klauseln `announce` oder `noannounce` verwendet werden. Wenn keine Routen-Einschränkungen bestehen, werden in der Regel alle Routen, die über HELLO oder RIP gelernt wurden, über EGP verteilt. Die Klausel `egpnetsreachable` beschränkt die EGP-Verteilung auf die in der Klausel aufgeführten Netze. Der metrische Wert, der für die über HELLO und RIP erkannten Routen verwendet wird, ist der Wert in der Klausel `defaultegpmetric`. Ist in dieser Klausel kein Wert angegeben, wird der Wert auf 255 gesetzt. Bei der Klausel `egpnetsreachable` können einzelne eindeutige metrische EGP-Werte nicht für jedes Netz gesetzt werden.

`defaultegpmetric` wird für alle Netze verwendet, ausgenommen sind die direkt angeschlossenen Netze mit einem metrischen Wert von 0.

```
martianets net [net] [net] ...
```

Diese Klausel wird an die `gated`-Liste der `martian`-Netze angehängt. `martian`-Netze sind Netze, die ungültig sind, und sollten ignoriert werden. `gated` ignoriert diese Netze, wenn es etwas von diesen Netzen hört. Ist die `external` Protokollierung aktiviert, wird im Protokoll eine entsprechende Meldung ausgegeben. Tritt die Klausel `martian-nets` häufiger auf, akkumulieren die Klauseln.

Eine ursprüngliche Liste von `martian`-Netzen ist in `gated` in der `include`-Datei `rt_control.h` in codierter Form enthalten. In dieser Liste stehen 127.0.0.0, 128.0.0.0, 191.253.0.0, 192.0.0.0, 223.255.255.0 und 224.0.0.0.

### Optionen in der Konfigurationsdatei für Routen autonomer Systeme

In den internen Routing-Tabellen hält `gated` die Nummer des autonomen Systems, das Information zu einer bestimmten Route geliefert hat. Autonome Systeme werden nur verwendet, wenn ein externes Routing-Protokoll, in diesem Fall EGP, verwendet wird. Den Routen wird die Nummer des autonomen Systems des EGP-Nachbarn zugeordnet, von dem sie gelernt wurden. Routen, die durch die internen Routing-Protokolle RIP und HELLO gelernt wurden, wird die Nummer des autonomen Systems zugeordnet, wie sie in der Klausel `autonomoussystem` angegeben wurde.

In der Regel verteilt `gated` die Routen, die über externe Routing-Protokolle gelernt wurden, nicht an die internen Routing-Protokolle weiter. Dies ist historisch begründet, da die EGP-fähige Systeme mit ARPAnet-Kern keine angemessene Überprüfung der Routing-Informationen, die sie empfangen, vornehmen können. Einige der folgenden Klauseln ermöglichen über interne Protokolle die Verteilung von externen Routen. Aus diesem Grund muß mit äußerster Sorgfalt vorgegangen werden, wenn die Verteilung von externen Routen zugelassen wird. Sind Sie im Umgang mit diesen Klauseln nicht sicher, setzen Sie sie nur mit großer Vorsicht ein.

Mit den folgenden Klauseln ist eine begrenzte Kontrolle der Routen aufgrund der Nummer des autonomen Systems möglich.

```
validAS netz AS as metric wert
```

Die Klausel `validAS` wird dient zur Überprüfung von Netzen bestimmter autonomer Systeme. Wenn von einem Nachbarn, bei dem die Option `validate` in der zugehörigen Klausel `egpneighbor` angegeben wurde, eine EGP-Aktualisierung empfangen wird, wird nach einer Klausel `validAS` gesucht, die das Netz und die Nummer des autonomen Systems des EGP-Nachbarns angibt. Wird die entsprechende Klausel `validAS` gefunden, wird das Netz für eine Einbindung in die Routing-Tabelle mit dem angegebenen metrischen Wert vorgesehen. Wird die Klausel `validAS` nicht gefunden, wird eine Warnmeldung ausgegeben und das Netz wird ignoriert.

Ein Netz kann in mehreren Klauseln `validAS` so eingetragen sein, daß es mit mehreren unterschiedlichen autonomen Systemen verbunden ist.

```
announcetoAS as0 {restrict|norestrict} ASlist as1 as2 as3 ...
noannouncetoAS as0 {restrict|norestrict} ASlist as1 as2 as3 ...
```

`announcetoAS` und `noannouncetoAS` steuern den Austausch der Routing-Informationen zwischen verschiedenen autonomen Systemen. In der Regel macht `gated` die Routing-Informationen zwischen autonomen Systemen nicht global bekannt. Ausgenommen sind davon Routen, die vom `gated`-eigenen autonomen System über RIP und HELLO gelernt

wurden; sie werden über EGP verteilt. Diese Klauseln ermöglichen die Verteilung von Routing-Informationen, die über EGP von einem autonomen System gelernt wurden, über EGP an ein anderes autonomes System oder über RIP oder HELLO an das eigene autonome System.

Ist die Klausel `announcetoAS` angegeben, wird die Information, die über EGP von den autonomen Systemen `as1`, `as2`, `as3`, ... gelernt wurde, an das autonome System `as0` verteilt. Wird das autonome System von `gated`, wie in der Klausel `autonomoussystem` angegeben, als `as0` eingetragen, wird diese Information über RIP und HELLO verteilt. Die Routen-Informationen von autonomen Systemen, die in `ASlist` nicht eingetragen sind, werden nicht an das autonome System `as0` verteilt.

Ist die Klausel `noannouncetoAS` angegeben, werden die Informationen, die über EGP von allen autonomen Systemen außer den gelisteten (`as1` usw.) gelernt wurden, an das autonome System `as0` verteilt. Wird das autonome System von `gated` als `as0` eingetragen, wird diese Information über RIP und HELLO nicht verteilt.

Die Option `[no]restrict` steuert, wie die Klauseln `announce` und `noannounce` auf die Verteilung von Routen an verschiedene autonome Systeme angewendet werden sollen. Wenn `restrict` angegeben ist, finden die normalen Bekanntmachungsbeschränkungen Anwendung. Wenn `norestrict` angegeben ist, werden Bekanntmachungsbeschränkungen nicht berücksichtigt, und alle Routen vom autonomen Ursprungssystem werden zum autonomen Zielsystem verteilt.

Es kann jeweils pro autonomes Zielsystem nur eine Klausel `announcetoAS` oder `noannounceAS` angegeben werden.

### Hinweise zu Konfigurationsoptionen

`gated` speichert seine Prozeßnummer in der Datei `/etc/inet/gated.pid`.

Wird EGP verwendet, wenn die Standardroute (über RIP gateway oder HELLO gateway) verteilt wird, und sind alle EGP-Nachbarn verloren, wird die Standardroute erst bekanntgemacht, wenn wenigstens ein EGP-Nachbar wieder vorhanden ist.

Wenn Routing-Einschränkungen verwendet werden, protokolliert `gated` alle ungültigen Netze in `Syslog` mit dem Log Level `LOG_WARNING` und dem Merkmal `LOG_DAEMON`.

Aufgrund der Komplexität der derzeitigen Netzwerktopologie und den vielen Netzzugängen "durch die Hintertür" empfiehlt sich die Verwendung von Routing-Einschränkungen. Bei den derzeitigen Routing-Strategien können illegale oder ungültige Netze ohne weiteres in den ARPAnet-Core oder das NSFnet-Backbone eindringen. Derzeit müssen die Routing-Einschränkungen jedoch eingesetzt werden, um ordnungsgemäß mit Internet zu arbeiten.

### Der interne metrische Wert von `gated`

gated speichert sämtliche metrischen Werte intern als Zeitverzögerung in Millisekunden, um die Struktur der HELLO-Zeitverzögerungen zu erhalten. Die interne Verzögerung reicht von 0 bis 30000 Millisekunden, wobei 30000 unendlich bedeutet. Die metrischen Werte von anderen Protokollen werden bei Empfang bzw. Senden in bzw. aus Zeitverzögerungen umgerechnet. Die EGP-Distanzen können nicht mit den metrischen HELLO- bzw. RIP-Werten verglichen werden, werden aber intern als Zeitverzögerungen gespeichert, damit Vergleiche mit anderen metrischen EGP-Werten möglich sind. Der Faktor für die Umwandlung von EGP-Distanzen und Zeitverzögerungen ist 100. RIP-spezifische und schnittstellenspezifische metrischen Werte werden anhand der folgenden Umrechnungstabellen in bzw. aus internen Zeitverzögerungen umgerechnet.

| Zeitverzögerung | metrischer RIP-Wert | metrischer RIP-Wert | Zeitverzögerung |
|-----------------|---------------------|---------------------|-----------------|
| 0 - 0           | 0                   | 0                   | 0               |
| 1 - 100         | 1                   | 1                   | 100             |
| 101 - 148       | 2                   | 2                   | 148             |
| 149 - 219       | 3                   | 3                   | 219             |
| 220 - 325       | 4                   | 4                   | 325             |
| 326 - 481       | 5                   | 5                   | 481             |
| 482 - 713       | 6                   | 6                   | 713             |
| 714 - 1057      | 7                   | 7                   | 1057            |
| 1058 - 1567     | 8                   | 8                   | 1567            |
| 1568 - 2322     | 9                   | 9                   | 2322            |
| 2323 - 3440     | 10                  | 10                  | 3440            |
| 3441 - 5097     | 11                  | 11                  | 5097            |
| 5098 - 7552     | 12                  | 12                  | 7552            |
| 7553 - 11190    | 13                  | 13                  | 11190           |
| 11191 - 16579   | 14                  | 14                  | 16579           |
| 16580 - 24564   | 15                  | 15                  | 24564           |
| 24565 - 30000   | 16                  | 16                  | 30000           |

### Hinweise zu Besonderheiten der Implementierung

In der Konfigurationsdatei von gated müssen sämtliche Verweise auf POINT-TO-POINT-Schnittstellen über die ZIEL-Adresse erfolgen. Dies ist die einzige Änderung, die gegenüber früheren Versionen an der Syntax der Konfigurationsdatei vorgenommen wurde. Früher wurde die Ursprungsadresse des PTP-Links verwendet. Ansonsten müßten die Konfigurationsdateien kompatibel sein.

Sämtliche Protokolle haben eine Wartezeit von zwei Minuten. Wenn es in einer Routenaktualisierung heißt, daß die benutzte Route gelöscht wird, löscht gated sie erst nach zwei Minuten.

Änderungen, die an Schnittstellen vorgenommen werden können, werden von gated bemerkt, ohne daß der Prozeß neu gestartet werden muß. Falls *netmask*, *subnetmask*, die Broadcast-Adresse oder der metrische Schnittstellenwert geändert werden, sollte die Schnittstelle mit *ifconfig* als defekt markiert werden und anschließend mindestens 30 Sekunden später als betriebsbereit markiert werden. Bei Optionsänderungen ist es nicht erforderlich, die Schnittstellen herunter und dann wieder heraufzufahren.

RIP sorgt für die Bekanntmachung und Überwachung von Host-Routen. Dadurch soll

die Konsistenz bei PTP-Verbindungen erhöht werden. Diese Version unterstützt auch die RIP\_TRACE-Befehle.

Teilnetzschnittstellen werden unterstützt. Teilnetz-Informationen werden nur über Schnittstellen zu anderen Teilnetzen des selben Netzes verteilt. Besteht beispielsweise zwischen zwei Class-B-Netzen ein Gateway, werden die Teilnetz-Routen für jedes Class-B-Netz nicht in das andere Class-B-Netz verteilt. Es wird nur die Nummer des Class-B-Netzes verteilt.

gated hört die Routing-REDIRECTs von Host und Netz ab und trägt sie in den eigenen internen Tabellen ein, was mit der Aktion des des Kernels parallel läuft. Dadurch ist die Umleitungsroutine von gated mit der Umleitungsroutine des Berkeley-Kernels so weit als möglich identisch. Im Gegensatz zum Berkeley-Kernel schaltet gated die Routen, die über ein REDIRECT erkannt wurden, nach sechs Minuten ab. Die Route wird dann aus den Kernel-Routing-Tabellen gelöscht. Dadurch ist die Konsistenz der Routing-Tabellen besser gewährleistet. Jede Route, die über ein REDIRECT erkannt wird, wird über *keine* Routing-Protokolle bekanntgemacht.

Der EGP-Code von gated verifiziert, daß alle gesendeten und empfangenen Netze laut EGP-Spezifikation gültige Class-A-, Class-B- oder Class-C-Netze sind. Die Informationen über Netze, die diese Kriterien nicht erfüllen, werden nicht verteilt. Wenn ein EGP-Aktualisierungspaket Informationen über ein Netz enthält, das kein Class-A-, Class-B- oder Class-C-Netz ist, wird die Aktualisierung als fehlerhaft angesehen und wird ignoriert. Die Information über das spezifische Netz wird nur ignoriert, wenn gated mit EGP\_IGNORE\_BAD übersetzt wurde. Diese Option ist mit Sorgfalt anzuwenden.

## DATEIEN

|                         |                                                              |
|-------------------------|--------------------------------------------------------------|
| /etc/inet/gated.conf    | Konfigurationsdatei                                          |
| /etc/inet/gated.pid     | Prozeßnummer des laufenden gated                             |
| /usr/tmp/gated_dump     | Speicherabzugdatei                                           |
| /etc/inet/gated.version | gated-Version, Erstellungsdatum, Startdatum und Prozeßnummer |
| /usr/sbin/gated         | gated selbst                                                 |

## SIEHE AUCH

routed(1M)

- RFC827 EXTERIOR GATEWAY PROTOCOL (egp)
- RFC888 STUB EXTERIOR GATEWAY PROTOCOL
- RFC891 DCN Local/Network Protocols (HELLO)
- RFC904 Exterior Gateway Protocol Formal Specification
- RFC911 EGP GATEWAY UNDER BERKELEY UNIX 4.2

## smtpqer(1M)

### BEZEICHNUNG

smtpqer - Nachrichten zur Auslieferung durch SMTP in Warteschlange einreihen

### ÜBERSICHT

smtpqer [-nu] [-a *zieladr*] [-d *bereich*] [-H *hallohost*] [-N] *sender host empf ...*

### BESCHREIBUNG

Die Beschreibung von smtpqer im "Referenzhandbuch für den Systemverwalter" [11] wird um die Option -N ergänzt:

-N

unterdrückt DNS-MX-Anfragen (siehe auch "Netzwerke - Leitfaden für Benutzer und Verwalter" [3]).

## swap(1M)

Beim Kommando `swap` dürfen die Schalter `-l` und `-s` nicht gleichzeitig angegeben werden. Die dritte Zeile im Abschnitt *ÜBERSICHT* lautet daher:

```
/usr/sbin/swap -l
```

Beim Abschnitt *BESCHREIBUNG* gelten folgende Hinweise:

`-l`

Die Beschreibung der Spalte `dev` wird berichtigt: Bei normalen Dateien wird nicht Null ausgegeben, sondern es werden die major, minor device numbers des slices ausgegeben, auf dem die Datei liegt.

`-s`

Der Schalter `-s` darf nur unter einer privilegierten Kennung verwendet werden.

## traceroute(1M)

Dieses Kommando sollte nur für Diagnosezwecke benutzt werden.

### BEZEICHNUNG

traceroute - Ausgabe des Weges, den Pakete zu einem Zielrechner durchlaufen

### ÜBERSICHT

```
traceroute [-m max_ttl] [-n] [-p port] [-q nqueries] [-r] [-s src_addr] [-g addr]
 [-t tos] [-v] [-w waittime] host [packetsize]
```

### BESCHREIBUNG

Das Internet ist eine umfangreiche, komplexe Konstellation von Netzen, die über Gateways miteinander verbunden sind. Das Nachvollziehen der Route, die ein Paket zu den verschiedenen Hostrechnern nimmt (bzw. das Feststellen des Gateways, an dem Pakete verworfen werden) kann mit Schwierigkeiten verbunden sein. `traceroute` wertet das Feld *ttl* ("time-to-live") im IP-Protokollkopf aus und versucht, von jedem Gateway auf dem Weg zu einem bestimmten Hostrechner die ICMP-Meldung TIME\_EXCEEDED zu bekommen.

Der einzige obligatorische Parameter ist der Name bzw. die IP-Nummer des Ziel-Hostrechners. Die Länge des Test-Datagramms beträgt standardmäßig 38 Byte; allerdings kann auch mit einem höheren Wert gearbeitet werden, indem nach dem Namen des Ziel-Hostrechners eine Paketgröße (in Byte) angegeben wird.

Zusätzlich stehen die folgenden Optionen zur Verfügung:

-m *max\_ttl*

Maximale Lebensdauer (maximale Anzahl der Zwischenstationen (hops)) für abgehende Testpakete auf *max\_ttl* Zwischenstationen festlegen. Standardwert ist 30 Zwischenstationen (derselbe Standardwert wie bei TCP-Verbindungen).

-n

Die Adressen von Zwischenstationen werden nur im numerischen Format ausgegeben; die symbolische Adresse wird unterdrückt (so bleibt dem Namensserver eine Name/Adreß-Abbildung für jedes Gateway im Übertragungsweg erspart).

-p *port*

*port* ist die Nummer des UDP-Ports, den Testpakete als ersten Port verwenden sollen (Standard: 33434). `traceroute` hofft, daß an den UDP-Anschlüssen *port* bis *port + max\_ttl - 1* des Ziel-Hostrechners keine Netzwerk-Dienste warten (und das Nachvollziehen der Route durch die zurückgegebene ICMP-Meldung PORT\_UNREACHABLE beendet wird). Wenn innerhalb des Standard-Bereichs ein

Netzwerk-Dienst wartet, kann mit dieser Option ein nicht benutzter Bereich von Anschlüssen angegeben werden.

-r

Die Daten werden unter Umgehung der normalen Routing-Tabellen direkt an einen Hostrechner in einem angeschlossenen Netzwerk geschickt. Befindet sich der Hostrechner nicht in einem direkt angeschlossenen Netzwerk, wird ein Fehler zurückgegeben. Mit dieser Option kann an einen lokalen Hostrechner mit ping über eine Schnittstelle, durch die keine Route verläuft (z.B. da die Schnittstelle durch routed(1M) aufgegeben worden ist), ein Datagramm geschickt werden.

-s *src\_addr*

eine IP-Adresse *src\_addr* (die als IP-Nummer und nicht über einen Hostnamen angegeben werden muß) als Ausgangsadresse für abgehende Testpakete benutzen. Auf Hostrechnern mit mehreren IP-Adressen kann über diese Option angegeben werden, daß als Ausgangsadresse nicht die IP-Adresse der Schnittstelle benutzt werden soll, über die das Testpaket abgeschickt wird. Handelt es sich bei der IP-Adresse nicht um eine der Schnittstellenadressen des jeweiligen Rechners, wird ein Fehler zurückgegeben; es findet dann keine Übertragung statt.

-g *addr*

IP-Option LSRR (Loose Source Record Route) zusätzlich zu den TTL-Tests einschalten. Hiermit kann angefragt werden, wie der Benutzer an der IP-Adresse *addr* ein bestimmtes Ziel erreicht.

-t *tos*

Type-Of-Service (Serviceklasse) in Testpaketen auf *tos* setzen (Standard: 0). Bei dem Wert muß es sich um eine dezimale Ganzzahl im Bereich 0 bis 255 handeln. Über diese Option kann festgestellt werden, ob verschiedene Werte in *tos* zu unterschiedlichen Übertragungswegen führen. (Sofern Sie nicht mit 4.4bsd arbeiten, handelt es sich hierbei möglicherweise um eine rein theoretische Frage, da die meisten Netzwerk-Dienste wie telnet und ftp Ihnen keine Möglichkeit zur Steuerung von TOS bieten). TOS ist nicht in jeder Einstellung sinnvoll oder zulässig (siehe IP-Spezifikation). Sinnvolle Werte dürften sein: -t 16 (geringe Verzögerung) und -t 8 (hoher Durchsatz).

-v

Ausführliche Ausgabe. Es werden die empfangenen ICMP-Pakete mit Ausnahme von TIME\_EXCEEDED und UNREACHABLE aufgelistet.

-w *waittime*

Wartezeit für eine Antwort auf ein Testpaket auf *waittime* Sekunden setzen (Standard: 3 Sek.).

Dieses Programm versucht, die potentielle Route eines IP-Pakets zu irgendeinem Internet-Hostrechner festzustellen, indem UDP-Testpakete mit kleinem *ttl* (time to live, Lebensdauer) abgeschickt und dann auf die ICMP-Antwort *time exceeded* von einem Gate-

way gewartet wird. Die Testpakete werden üblicherweise mit dem *ttl*-Wert 1 abgeschickt; *ttl* wird dann so oft um 1 erhöht, bis die ICMP-Meldung *port unreachable* zurückkommt (diese Meldung zeigt die Ankunft des Pakets bei *host* an) oder ein Maximalwert erreicht wird (der Standardwert beträgt hier 30 Zwischenstationen und kann mit dem Schalter *-m* geändert werden). Bei jeder *ttl*-Einstellung werden drei Testpakete geschickt (dieser Wert kann mit dem Schalter *-q* geändert werden); in jeder Zeile ist der *ttl*-Wert, die Adresse des Gateway sowie die Zeitdauer enthalten, die jedes Testpaket unterwegs war. Kommen die Antworten auf die verschiedenen Testpakete von unterschiedlichen Gateways, wird die Adresse jedes antwortenden Systems ausgegeben. Wird nicht innerhalb von 3 Sekunden eine Antwort empfangen (dieser Wert kann mit dem Schalter *-w* geändert werden), wird für dieses Testpaket das Zeichen *\** ausgegeben.

Soll das UDP-Testpaket nicht vom Ziel-Hostrechner verarbeitet werden, so wird der Ziel-Anschluß auf einen wahrscheinlich nicht vorhandenen Wert gesetzt (sollte dieser Wert auf dem Ziel-Hostrechner dennoch vorhanden sein, kann er mit dem Schalter *-p* geändert werden).

Die Kommandozeile mit der zugehörigen Ausgabe kann beispielsweise folgendermaßen aussehen:

```
[yak 71]# traceroute nis.nsf.net.
```

```
traceroute to nis.nsf.net (35.1.1.48), 30 hops max, 56 byte packet
 1 helios.ee.lbl.gov (128.3.112.1) 19 ms 19 ms 0 ms
 2 lilac-dmc.Berkeley.EDU (128.32.216.1) 39 ms 39 ms 19 ms
 3 lilac-dmc:Berkeley.EDU (128.32.216.1) 39 ms 39 ms 19 ms
 4 ccngw-ner-cc.Berkeley.EDU (128.32.136.23) 39 ms 40 ms 39 ms
 5 ccn-nerif22.Berkeley.EDU (128.32.168.22) 39 ms 39 ms 39 ms
 6 128.32.197.4 (128.32.197.4) 40 ms 59 ms 59 ms
 7 131.119.2.5 (131.119.2.5) 59 ms 59 ms 59 ms
 8 129.140.70.13 (129.140.70.13) 99 ms 99 ms 80 ms
 9 129.140.71.6 (129.140.71.6) 139 ms 239 ms 319 ms
10 129.140.81.7 (129.140.81.7) 220 ms 199 ms 199 ms
11 nic.merit.edu (35.1.1.48) 239 ms 239 ms 239 ms
```

Beachten Sie, daß die Zeilen 2 und 3 identisch sind. Dies ist auf einen fehlerhaften Systemkern auf der zweiten Zwischenstation (*lbl-csam.arpa*) zurückzuführen, die das Paket mit dem *ttl*-Wert 0 weiterleitet (hierbei handelt es sich um einen Fehler in der 4.3BSD-Distribution).

## Ein interessanteres Beispiel:

```
[yak 72]% tracertool allspice.lcs.mit.edu.
```

```
tracertool to allspice.lcs.mit.edu (18.26.0.115), 30 hops max
 1 helios.ee.lbl.gov (128.3.112.1) 0 ms 0 ms 0 ms
 2 lilac-dmc.Berkeley.EDU (128.32.216.1) 19 ms 19 ms 19 ms
 3 lilac-dmc.Berkeley.EDU (128.32.216.1) 39 ms 19 ms 19 ms
 4 ccngw-ner-cc.Berkeley.EDU (128.32.136.23) 19 ms 39 ms 39 ms
 5 ccn-nerif22.Berkeley.EDU (128.32.168.22) 20 ms 39 ms 39 ms
 6 128.32.197.4 (128.32.197.4) 59 ms 119 ms 39 ms
 7 131.119.2.5 (131.119.2.5) 59 ms 59 ms 39 ms
 8 129.140.70.13 (129.140.70.13) 80 ms 79 ms 99 ms
 9 129.140.71.6 (129.140.71.6) 139 ms 139 ms 159 ms
10 129.140.81.7 (129.140.81.7) 199 ms 180 ms 300 ms
11 129.140.72.17 (129.140.72.17) 300 ms 239 ms 239 ms
12 * * *
13 128.121.54.72 (128.121.54.72) 259 ms 499 ms 279 ms
14 * * *
15 * * *
16 * * *
17 * * *
18 ALLSPICE.LCS.MIT.EDU (18.26.0.115) 339 ms 279 ms 279 ms
```

Beachten Sie, daß die Gateways, die um 12, 14, 15, 16 & 17 Zwischenstationen entfernt sind, die ICMP-Meldung *time exceeded* entweder überhaupt nicht abschicken oder sie mit einem zu kleinen *ttl*-Wert abschicken, so daß sie den lokalen Hostrechner nicht erreichen kann. Auf 14 - 17 läuft der MIT C Gateway-Code, der keine *time exceeded*-Meldungen abschickt. Die Gründe für das Problem mit Gateway 12 sind unbekannt.

Daß Gateway 12 im obigen Beispiel keine Meldung schickt, kann auf einen Fehler im Netzwerkcode von 4.[23]BSD (und seinen Derivaten) zurückzuführen sein: 4.x (x <= 3) schickt eine UNREACHABLE-Meldung mit dem *ttl*-Wert, der jeweils im ursprünglichen Datagramm verblieben ist. Da der verbleibende *ttl*-Wert für Gateways 0 ist, kann die ICMP-Meldung *time exceeded* mit Sicherheit nicht zu dem lokalen Hostrechner zurückkehren. Das Verhalten dieses Fehlers ist etwas interessanter, wenn er auf dem Ziel-System auftritt:

```

1 helios.ee.lbl.gov (128.3.112.1) 0 ms 0 ms 0 ms
2 lilac-dmc.Berkeley.EDU (128.32.216.1) 39 ms 19 ms 39 ms
3 lilac-dmc.Berkeley.EDU (128.32.216.1) 19 ms 39 ms 19 ms
4 ccngw-ner-cc.Berkeley.EDU (128.32.136.23) 39 ms 40 ms 19 ms
5 ccn-nerif35.Berkeley.EDU (128.32.168.35) 39 ms 39 ms 39 ms
6 csgw.Berkeley.EDU (128.32.133.254) 39 ms 59 ms 39 ms
7 * * *
8 * * *
9 * * *
10 * * *
11 * * *
12 * * *
13 rip.Berkeley.EDU (128.32.131.22) 59 ms ! 39 ms ! 39 ms !

```

Beachten Sie, daß es 12 *gateways* gibt (13 ist das endgültige Ziel), von denen genau die Hälfte *fehlt*. Dazu ist es folgendermaßen gekommen: *rip* benutzt den *tll*-Wert aus dem angekommenen Datagramm als *tll*-Wert in seiner ICMP-Antwort. Deshalb löst die Antwort auf ihrem Rückweg eine Zeitlimitüberschreitung aus (die aber nicht gemeldet wird, da für ICMP-Meldungen selbst keine ICMP-Meldungen abgeschickt werden). Daher muß ein *tll*-Wert eingesetzt werden, der mindestens das Doppelte der Länge des Übertragungswegs beträgt. Somit ist *rip* in Wirklichkeit lediglich um sieben Zwischenstationen entfernt. Wenn eine Antwort mit dem *tll*-Wert 1 zurückkehrt wird, ist dies ein Hinweis darauf, daß es dieses Problem gibt. *traceroute* gibt nach der Zeitangabe bei einem *tll*-Wert kleiner gleich 1 ein Ausrufezeichen (!) aus. Da von den Herstellern häufig eine Menge veralteter oder Nicht-Standard-Software ausgeliefert wird, sollte man sich auf das häufige Auftreten dieses Problems einstellen oder bei der Auswahl der Ziel-Hostrechner für die Testpakete sorgfältig sein.

Weitere Meldungen nach der Zeitangabe können sein: !H, !N, !P (betreffender Hostrechner, Netzwerk bzw. Protokoll nicht erreichbar), !S oder !F (Störung in Ausgangs-Route oder Fragmentierung notwendig - diese Hinweise sollten jedoch in keinem Fall erscheinen und weisen gegebenenfalls auf eine Störung im betreffenden Gateway hin). Wenn so gut wie alle Testpakete zum Ergebnis "unreachable" führen, startet *traceroute* keinen neuen Versuch und beendet sich.

```
traceroute -g 10.3.0.5 128.182.0.0
```

zeigt den Übertragungsweg zwischen Cambridge Mailbridge und PSC;

```
traceroute -g 192.5.146.4 -g 10.3.0.5 35.0.0.0
```

dagegen zeigt, wie Merit von Cambridge Mailbridge erreicht wird; dabei wird PSC zum Erreichen von Mailbridge benutzt.

Dieses Programm ist zum Einsatz beim Testen, Messen und Verwalten von Netzwerken bestimmt. Es sollte hauptsächlich zur manuellen Fehlertlokalisierung benutzt werden. Um eine Überlastung des Netzes zu verhindern, sollte *traceroute* möglichst nicht während des Normalbetriebs oder innerhalb von Kommandoskripts benutzt werden.

**SIEHE AUCH**

netstat(1), ping(1M).

## exa(7)

### BESCHREIBUNG

Der Treiber `exa` unterstützt das 8 mm Video-Bandlaufwerk EXABYTE EXB-8200 am SCSI-Bus (intern und extern).

Für den MX300 mit Onboard-SCSI-Hostadapter NCR 53C710 gilt: Der Treiber `exa` unterstützt außerdem den Streamer Tandberg TDC3660b im Hauptgehäuse.

### Struktur der Gerätenummer (minor device number)

Einheitennummer (unit number)

- Bits 0-2: SCSI-Targetnummer des Geräts.
- Bit 3: immer 1.
- Bit 10-13: Nummer des Host Adapters (MX300: 0 = interner Bus).

Bedeutung der weiteren Bits:

- Bit 4: Falls die letzte Operation ein Schreibvorgang war, wird beim Schließen eine Dateimarke geschrieben. Falls die letzte Operation ein Lesevorgang war, wird "skip 1/skip on filemark" an das Gerät gesandt.
- Bit 5: Das Band wird beim Schließen zurückgespult.
- Bit 6: Das Band wird beim Schließen entladen (ausgeworfen).
- Bit 7: Beim Öffnen wird der Raw-Modus ausgewählt. Der Raw-Modus ist satzorientiert (record-oriented). Datensätze müssen mit der gleichen Puffergröße gelesen werden, mit der sie geschrieben wurden. Die Abarbeitung eines Auftrags im Raw-Modus dauert länger, da es schwierig ist, das Band im "streaming mode" zu halten.
- Bit 8: Bei einem Tandberg Streamer wird nach dem Einlegen einer Kassette vor einem Schreib- oder Lesevorgang ein Umspulen (Retensioning) des Bandes durchgeführt. Anschließend wird auf dem Bandanfang positioniert.

### ioctl-Aufrufe

Verschiedene ioctl-Aufrufe werden unterstützt. Die für Magnetbänder üblichen Aufrufe sind in `sys/mtio.h` und `sys/tape.h` beschrieben. Spezielle ioctl-Aufrufe und Datenstrukturen sind in `sys/exaioct1.h` definiert.

## DATEIEN

Die Gerätedateien haben folgende Namen:

```
/dev/rmt/exaYtZ,
/dev/rmt/exaYtZn,
/dev/rmt/exaYtZv,
/dev/rmt/exaYtZvn
```

Dabei bezeichnet *Y* die Nummer des Controllers und *Z* die SCSI-Targetnummer (bei MX300 den ON BOARD-NCR-53C710 Controller). Eine 1 für *Y* bezeichnet den ersten ADP32 Controller (adp32.0), eine 2 den zweiten ADP32 Controller (adp32.1) usw.

Die Gerätedateien (Geräteknoten, device nodes) für Controller 1 und Target 0 werden bei der Installation angelegt. Falls Geräte an einem anderen Controller oder an einer anderen Target-Adresse betrieben werden, müssen die entsprechenden Gerätedateien mit `/etc/init.d/scsigen exa` erzeugt werden. Dabei müssen die Geräte eingeschaltet mit dem System verbunden sein.

Die Gerätedateien für den Streamer Tandberg TDC 3660 an Onboard-SCSI-HA NCR53710 sind:

```
/dev/rmt/c0s0
/dev/rmt/c0s0n - no rewind
/dev/rmt/c0s0r - Retension bei open
/dev/rmt/c0s0nr - Retension bei open, no rewind
```

Diese Gerätedateien werden nicht durch `scsigen` erzeugt.

### **/dev/exa-Knoten**

Der Geräteknoten `/dev/exa*` ist lediglich aus Kompatibilitätsgründen eingerichtet. Mit diesem Geräteknoten kann nur ein Exabyte-Gerät am 1. NCR-Hostadapter angesprochen werden (d.h. `/dev/rmt/exa1t*`). Es wird empfohlen nurmehr die `/dev/rmt/exa*-Knoten` zu verwenden.

## SIEHE AUCH

adp32(7), tape.h(4), mtio.h(4), exaioct1.h(4).

## FEHLERMELDUNGEN

Der Treiber gibt folgende Fehlermeldungen aus:

exaYtZ: no cartridge loaded.  
exaYtZ: data cartridge write protected.  
exaYtZ: new cartridge!.  
exaYtZ: No write after LEOT allowed.  
exaYtZ: No WriteFilemark after LEOT allowed.  
exaYtZ: No Erase after LEOT allowed.  
exaYtZ: No WriteVar after LEOT allowed.  
exaYtZ: LEOT (logical end of tape) encountered.  
exaYtZ: PEOT (physical end of tape) reached.  
exaYtZ: VOLUME OVERFLOW.  
exaYtZ: bad cartridge with unrecoverable medium error.  
exaYtZ: illegal tape position.

Dabei bezeichnet *Y* die Nummer des Controllers und *Z* die SCSI-Targetnummer.

### Hinweis

Findet ein Reset auf dem SCSI-Bus statt, an dem das Exabyte-Gerät betrieben wird, dann führt das Exabyte-Gerät ein Rewind durch und positioniert das Band auf den Bandanfang (BOT). Die bisherige Position des Bandes geht verloren. Weitere Aufträge werden am Bandanfang ausgeführt. Bereits geschriebene Daten können dadurch überschrieben werden. Ein Sicherungslauf, der gerade stattgefunden hat, muß z.B. wiederholt werden.

## shd(7)

### BESCHREIBUNG

Der Treiber `shd` unterstützt SCSI-Festplatten wie z.B. Micropolis am NCR Host Adapter (ADP32 Controller). Er unterstützt bis zu 7 Targetgeräte pro Host Adapter und bis zu 8 logische Untereinheiten (LUN, logical unit). Die tatsächliche Anzahl der verwendbaren Festplatten hängt vom `shd`-Eintrag in der Konfigurationsdatei `mdevice` und den verfügbaren Host Adapters ab. Der Treiber bestimmt die Geometrie der Festplatte dynamisch (siehe unten). Für die einzelnen Slices (Slice 1 bis 15) bietet der Treiber jeweils eine Raw-Schnittstelle und eine Block-Schnittstelle an. Auf die gesamte Partition kann über die Raw-Schnittstelle von Slice 0 zugegriffen werden.

Die Nummer des Geräts (minor device number), auf das zugegriffen wird, legt fest, wie das entsprechende Laufwerk behandelt wird. Die Gerätenummer (minor device number) ist so aufgebaut:

Bits 0 - 3 kennzeichnen die Slice (0-15).

Bits 4 - 6 sind für logische Untereinheiten (LUN) reserviert und werden nicht benutzt. Sie sind immer 0.

Bits 7 - 9 geben die SCSI-Targetnummer an einem SCSI-Bus an.

Bits 10 -12 wählen einen der Host Adapter aus.

Slice 0 steht für die komplette UNIX-Systempartition, wie sie in der `fdisk`-Tabelle definiert ist. Die weiteren Slices werden durch Angaben definiert, die im Inhaltsverzeichnis des Datenträgers (VTOC: volume table of contents) stehen. Slice 0 umfaßt die gesamte UNIX-Partition, insbesondere also alle anderen Slices (s1 - s15).

Die ganze Festplatte wird auf zwei Ebenen partitioniert: Erstens sind Plattenbereiche, die von einem anderen Betriebssystem benutzt werden sollen, in der `fdisk`-Tabelle beschrieben. Diese Tabelle liegt im ersten Block der Festplatte. Zweitens werden die UNIX-Bereiche der Festplatte weiter aufgeteilt. Dazu wird die Information aus dem VTOC verwendet, das in einem beliebigen Block enthalten sein kann. Zur Zeit befindet sich das VTOC im Sektor 29 der aktiven UNIX-Partition. Das VTOC enthält auch Informationen über andere Betriebssysteme, die in der `fdisk`-Tabelle beschrieben sind. Wenn die zu der Festplatte gehörige Gerätedatei geöffnet wird, liest der Treiber das VTOC und benutzt es, um seine Tabellen über logische Platten (logische Laufwerke) zu füllen. Für die Zuweisung verwendet der Treiber die Gerätenummer (minor device number).

Jeder Partition wird in der `fdisk`-Tabelle ein Typ zugeordnet (z.B. DOS, UNIX o.a.). Das UNIX-System kann eine Partition (Dateisystem) nur dann nutzen, wenn sie vom richtigen Typ ist. Eine DOS-Partition kann z.B. vom UNIX-System nicht benutzt werden außer im Raw-Modus ohne Dateisystem (raw, non-file system device).

Zylinder 0 ist auf jedem Laufwerk reserviert für Laufwerkscharakteristika und die `fdisk`-Partitionstabelle; beide liegen im physikalischen Festplattensektor 0 mit der Länge 512 Bytes. Weiter ist der Zylinder 0 reserviert für eine Volume- und Dateikennung, die beide im physikalischen Festplattensektor 1 mit der Länge 512 Bytes zu finden sind; und dann liegt im Zylinder 0 noch der Boot-Bereich für das Hochfahren des Systems. Die Startadresse des Boot-Bereichs wird durch ein Byte in der Dateikennung angezeigt: `flb.f_boe[4]` enthält den Wert in Kbytes (normalerweise 4 KB). Zur Zeit ist es nicht möglich, von einer `shd` SCSI-Festplatte an einem NCR Host Adapter zu booten.

Die `fdisk`-Tabelle beschreibt die Partitionierung der Festplatte wie bei DOS. Es sind höchstens `FD_NUMPART` (= 4) Partitionen möglich. Normalerweise existiert nur eine Partition für UNIX die aktive UNIX Partition - die die ganze Festplatte mit Ausnahme von Zylinder 0 belegt.

Defekte Sektoren werden vom SCSI-Controller ausgeblendet. Da diese Abbildung auf Hardware-Ebene stattfindet, bleibt die Geometrie der Festplatte unverändert. Für die Software besteht die Festplatte aus einer Folge intakter Blöcke.

Auf der aktiven UNIX Partition befinden sich im Sektor `PDINFO_SEC` (= Sektor 0 der Partition) die Datenstrukturen `pdinfo` und `vtoc`. `pdinfo` (physical disk info) beschreibt die physikalischen Parameter der Festplatte, während `vtoc` (volume table of contents) die einzelnen UNIX-Slices innerhalb der gesamten UNIX-Partition beschreibt. Die UNIX-Slice 0 ist die "Backup-Slice", sie enthält die komplette UNIX-Partition.

Um fehlerhafte Sektoren auf Ersatzsektoren abzubilden, steht auf der MX500 das Kommando `addbad` zur Verfügung.

### **ioctl-Aufrufe**

Die verfügbaren `ioctl`-Aufrufe und Datenstrukturen sind in `sys/shdioc1.h` und `sys/vtoc.h` definiert.

### **Partitionen**

In der `fdisk`-Tabelle können Partitionen an Zylindergrenzen beginnen, im VTOC können Slices jedoch auf Spurgrenzen beginnen. Dies wird für den Boot-Bereich der UNIX Systempartition benutzt, um die erste Spur (mit dem Boot-Code) nicht mit in die Slice aufzunehmen. Die `fdisk`-Tabelle erlaubt höchstens vier Partitionen auf einer Festplatte, mit dem VTOC kann die UNIX-Partition jedoch in bis zu 16 Slices aufgeteilt werden. Jede Slice wird durch die Gerätenummer identifiziert. Einer Slice wird ihre Gerätenummer zugeordnet, wenn zum ersten Mal auf die Platte zugegriffen wird. Die Zuordnung ist in der Datei `/etc/partitions` festgelegt und bleibt so lange unverändert, bis das Dienstprogramm `disksetup(1M)` erneut gestartet wird.

Versuche, ein Dateisystem zu öffnen, schlagen fehl (non-existent device), wenn dafür keine Slice existiert. Ebenso erfolglos sind Versuche, Slices einzuhängen (siehe mount(1M)), die kein UNIX-Dateisystem enthalten.

## DATEIEN

Die Gerätedateien haben folgende Namen:

externe Platten:

```
/dev/dsk/shdXtYsZ,
/dev/rdisk/shdXtYsZ
```

interne Platten:

```
/dev/dsk/c0dYsZ
/dev/rdisk/c0dYsZ
```

Dabei steht *X* für die Nummer des Controllers, *Y* für die SCSI-Targetnummer und *Z* für die Slicenummer. Eine 1 für *X* bezeichnet den ersten ADP32 Controller (adp32.0), eine 2 den zweiten ADP32 Controller (adp32.1) usw. *Z* bezeichnet die Slicenummer. Sie kann bei Geräten, die im Raw-Modus betrieben werden, Werte von 0 bis 15 annehmen. Bei Geräten im Block-Modus sind Slicenummern von 1 bis 15 m gleich. Die Gerätedateien (Geräteknoten, device nodes) werden beim Hochfahren des Systems erzeugt, nachdem ein neuer Kern generiert wurde (z.B. bei der Installation). Die jeweiligen Geräte müssen eingeschaltet und mit dem System verbunden sein, damit die Geräteknoten korrekt erzeugt werden können.

Beim MX300 mit Onboard-SCSI-Hostadapter NCR 53C710 sind die Gerätedateien `cxdysz` in `/dev/dsk` und `/dev/rdisk`. Diese Dateien werden nicht durch `scsigen` erzeugt, sondern schon beim Installieren des Systems angelegt.

*x* = Hostadapter-Nummer (immer 0)

*y* = Laufwerk-Nummer (bei Platten am NCR 53C710 identisch mit der Target-Nummer)

*z* = Slice-Nummer

Es werden bei vorhandenen internen SCSI-Platten über die Gerätedateien sowohl interne als auch externe SCSI-Platten angesprochen. Sind keine internen SCSI-Platten vorhanden, werden nur die externe Platten angesprochen.

Beim MX500 werden nur externe Platten angesprochen, und zwar über die Gerätedateien

```
/dev/dsk/shdXtYsZ und
/dev/rdisk/shdXtYsZ.
```





## Reguläre Ausdrücke

Reguläre Ausdrücke werden verwendet, um in einem Text nach Stellen zu suchen, die zu einem vorgegebenen Muster passen. Ein regulärer Ausdruck steht für eine Menge von Zeichenketten. Von jeder Zeichenkette in dieser Menge sagt man, daß sie zu dem regulären Ausdruck paßt. Ein oder mehrere reguläre Ausdrücke bilden ein Muster.

Ein regulärer Ausdruck besteht aus einer Folge von Zeichen. Bei diesen Zeichen unterscheidet man

- einfache Zeichen und
- Sonderzeichen.

Einfache Zeichen sind alle Zeichen im Zeichensatz außer dem Neue-Zeile-Zeichen und den Sonderzeichen. Einfache Zeichen in einem Muster stehen für sich selbst, z.B. passen zu dem Muster *abc* nur diejenigen Zeichenketten, in denen die Folge *abc* an irgendeiner Stelle enthalten ist.

Sonderzeichen stehen nicht für sich selbst, sondern haben eine besondere Bedeutung. Diese ist unten erläutert.

Man unterscheidet zwei Arten regulärer Ausdrücke:

- einfache reguläre Ausdrücke
- erweiterte reguläre Ausdrücke

Wie die verschiedenen Arten regulärer Ausdrücke gebildet werden, wird in den folgenden Abschnitten beschrieben.

## Reguläre Ausdrücke

---

Die folgende Tabelle gibt einen Überblick darüber, welche Kommandos reguläre Ausdrücke verarbeiten:

| Kommando | Art der regulären Ausdrücke     |
|----------|---------------------------------|
| awk      | erweiterte internationalisierte |
| bfs      | einfache                        |
| csplit   | einfache                        |
| ed       | einfache                        |
| egrep    | erweiterte                      |
| ex       | *)                              |
| expr     | einfache                        |
| extract  | einfache                        |
| grep     | einfache                        |
| lex      | erweiterte                      |
| n1       | einfache                        |
| pg       | einfache                        |
| sed      | einfache                        |
| vi       | *)                              |

Tabelle 9: Kommandos, die reguläre Ausdrücke verarbeiten

- \*) Die Kommandos *ex* und *vi* verarbeiten reguläre Ausdrücke, die von den einfachen regulären Ausdrücken in einigen Punkten abweichen. Dies ist bei *ex* und *vi* beschrieben.

## EINFACHE REGULÄRE AUSDRÜCKE

Einfache reguläre Ausdrücke werden wie folgt gebildet:

| Nr. | regulärer Ausdruck                           | Bedeutung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Beispiel                                                                                                                                                                                            | passende Zeichenketten                                                                                                                                                                                                                                                                                                                                                                       |
|-----|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | <code>c</code>                               | Das Zeichen <code>c</code> , wobei <code>c</code> kein Sonderzeichen sein darf.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <code>a</code>                                                                                                                                                                                      | <code>a</code>                                                                                                                                                                                                                                                                                                                                                                               |
| 2   | <code>\c</code>                              | Das Zeichen <code>c</code> , wobei <code>c</code> jedes Zeichen sein darf außer <code>( ) { }   2 3 4 5 6 7 8 9</code><br>Sinnvoll ist ein regulärer Ausdruck der Form <code>\c</code> , wenn <code>c</code> ein Sonderzeichen ist.<br><code>\c</code> steht dann für das Zeichen <code>c</code> . Sonderzeichen verlieren durch einen vorangestellten Gegenschrägstrich ihre Sonderbedeutung (Entwertung von Sonderzeichen).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <code>\a</code><br><br><code>\*</code>                                                                                                                                                              | <code>a</code><br><br><code>*</code>                                                                                                                                                                                                                                                                                                                                                         |
| 3   | <code>.</code>                               | Ein beliebiges Zeichen.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | <code>.</code>                                                                                                                                                                                      | <code>a, x, *, ...</code>                                                                                                                                                                                                                                                                                                                                                                    |
| 4   | <code>[s]</code><br><br><code>[c1-c2]</code> | Eines der Zeichen, die in der Zeichenkette <code>s</code> enthalten sind. Wenn eines der Zeichen die schließende eckige Klammer <code>]</code> sein soll, muß diese an erster Stelle stehen.<br>Wenn eines der Zeichen der Bindestrich <code>-</code> sein soll, muß dieser an erster oder an letzter Stelle stehen.<br>Wenn eines der Zeichen das Dach <code>^</code> sein soll, darf dieses an beliebiger Stelle stehen außer an erster.<br><br>Ein beliebiges Zeichen aus dem Bereich von <code>c1</code> bis <code>c2</code> , gemäß der ASCII-Sortierreihenfolge (Grenzen <code>c1</code> und <code>c2</code> eingeschlossen).<br><br><code>c1</code> muß in der ASCII-Sortierreihenfolge vor <code>c2</code> stehen. Ist dies nicht der Fall, dann steht <code>c1-c2</code> nicht für einen Bereich, sondern für die beiden Zeichen <code>c1</code> und <code>c2</code> .<br><br>Eine Kombination der beiden Formen ist möglich:<br><code>[s1c1-c2s2]</code> | <code>[mz]</code><br><br><code>[]a]</code><br><br><code>[-a]</code><br><code>[a-]</code><br><br><code>[a^]</code><br><br><code>[a-m]</code><br><br><code>[m-a]</code><br><br><code>[ado-qxz]</code> | <code>m, z</code><br><br><code>], a</code><br><br><code>-, a</code><br><code>-, a</code><br><br><code>a, ^</code><br><br><code>a, m</code> sowie jedes Zeichen, das in der ASCII-Sortierreihenfolge dazwischen steht<br><br><code>m, a</code><br><br><code>a, d, o, q, x, z</code> sowie jedes Zeichen, das in der ASCII-Sortierreihenfolge zwischen <code>o</code> und <code>q</code> steht |

## Reguläre Ausdrücke

| Nr. | regulärer Ausdruck | Bedeutung                                                                                                                                                                                                                    | Beispiel  | passende Zeichenketten                                                                                        |
|-----|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|---------------------------------------------------------------------------------------------------------------|
| 5   | [^s]               | Eines der Zeichen, die nicht in der Zeichenkette <i>s</i> enthalten sind.                                                                                                                                                    | [^xyz]    | jedes Zeichen außer x, y, z                                                                                   |
|     | [^c1-c2]           | Ein beliebiges Zeichen, das nicht im Bereich von <i>c1</i> bis <i>c2</i> liegt. Das für [c1-c2] Gesagte gilt analog.                                                                                                         | [^0-9]    | jedes Zeichen außer 0, 9 und allen Zeichen, die in der ASCII-Sortierreihenfolge zwischen 0 und 9 stehen       |
|     |                    | Eine Kombination der beiden Formen ist möglich:<br>[^\s1c1-c2s2]                                                                                                                                                             | [^a0-9b]  | jedes Zeichen außer a, b, 0, 9 und allen Zeichen, die in der ASCII-Sortierreihenfolge zwischen 0 und 9 stehen |
| 6   | r*                 | Null-, ein- oder mehrmals der reguläre Ausdruck <i>r</i> . <i>r</i> muß von der Form 1 - 5, 12, 15 oder 16 sein.                                                                                                             | a*        | nichts, a, aa, aaa, ...                                                                                       |
| 7   | r\{m,n\}           | Mindestens <i>m</i> - und höchstens <i>n</i> -mal der reguläre Ausdruck <i>r</i> . <i>r</i> muß von der Form 1 - 5, 12, 15 oder 16 sein.                                                                                     | a\{1,2\}  | a oder aa                                                                                                     |
|     | r\{m\}             | Genau <i>m</i> -mal der reguläre Ausdruck <i>r</i> . <i>r</i> muß von der Form 1 - 5, 12, 15 oder 16 sein.                                                                                                                   | a\{3\}    | aaa                                                                                                           |
|     | r\{m,\}            | Mindestens <i>m</i> -mal der reguläre Ausdruck <i>r</i> . <i>r</i> muß von der Form 1 - 5, 12, 15 oder 16 sein.                                                                                                              | a\{3,\}   | aaa, aaaa, aaaaa, ...                                                                                         |
| 8   | rx                 | (Verkettung) Aufeinanderfolge einer zum regulären Ausdruck <i>r</i> passenden Zeichenkette und einer zum regulären Ausdruck <i>x</i> passenden Zeichenkette. <i>r</i> und <i>x</i> dürfen beliebige reguläre Ausdrücke sein. | [ab].     | ax, a3, a*, bz, ...                                                                                           |
| 9   | ^r                 | Eine zum regulären Ausdruck <i>r</i> passende Zeichenkette am Zeilenanfang, d.h. direkt nach einem Neue-Zeile-Zeichen oder am Dateianfang. <i>r</i> darf ein beliebiger regulärer Ausdruck sein, außer von der Form 9.       | ^[aA]pfel | apfel oder Apfel am Zeilenanfang                                                                              |

| Nr. | regulärer Ausdruck | Bedeutung                                                                                                                                                                                                                                                                                                                 | Beispiel                                                                         | passende Zeichenketten                                 |
|-----|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|--------------------------------------------------------|
| 10  | <code>r\$</code>   | Eine zum regulären Ausdruck $r$ passende Zeichenkette am Zeilenende, d.h. direkt vor einem Neue-Zeile-Zeichen. $r$ darf ein beliebiger regulärer Ausdruck sein, außer von der Form 10.                                                                                                                                    | <code>[bB]irne\$</code>                                                          | birne oder Birne am Zeilenende                         |
| 11  | <code>\(r\)</code> | Zeichenketten, die zum regulären Ausdruck $r$ passen. $r$ darf ein beliebiger regulärer Ausdruck sein.<br>Nur sinnvoll zusammen mit Nr. 12                                                                                                                                                                                | <code>\([aA]pfel\)</code>                                                        | apfel, Apfel                                           |
| 12  | <code>\n</code>    | $n$ ist eine ganze Zahl von 1 bis 9. Wenn <code>\n</code> in einem zusammengesetzten regulären Ausdruck vorkommt, steht es für den regulären Ausdruck $x$ , wobei $x$ der $n$ -te in <code>\(</code> und <code>\)</code> eingeschlossene reguläre Ausdruck ist, der in dem zusammengesetzten regulären Ausdruck vorkommt. | <code>\(a(b)\)\2</code><br><code>\(ha\)\1i\1lo</code><br><code>\(ab\)x\1*</code> | abb<br>hallihallo<br>abx,<br>abxab,<br>abxabab,<br>... |

Tabelle 10: Abbildung einfacher regulärer Ausdrücke

### Sonderzeichen

| Sonderzeichen                                                                                          | Das linksstehende Zeichen ist ein Sonderzeichen, wenn ...                                                                                                                           |
|--------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| \                                                                                                      | - ihm kein Gegenschrägstrich \ vorangestellt ist.                                                                                                                                   |
| .<br>[                                                                                                 | - ihm kein Gegenschrägstrich \ vorangestellt ist und<br>- es nicht in eckigen Klammern [...] steht.                                                                                 |
| *                                                                                                      | - ihm kein Gegenschrägstrich \ vorangestellt ist,<br>- es nicht in eckigen Klammern [...] steht,<br>- es nicht das erste Zeichen eines Musters ist und<br>- es nicht nach \) steht. |
| \$                                                                                                     | - es das letzte Zeichen eines Musters ist.                                                                                                                                          |
| ^                                                                                                      | - es das erste Zeichen eines Musters ist oder<br>- es das erste Zeichen in eckigen Klammern [...] ist.                                                                              |
| -                                                                                                      | - es in eckigen Klammern steht, aber nicht an erster<br>oder letzter Stelle.                                                                                                        |
| Begrenzungs-<br>zeichen für<br>reguläre<br>Ausdrücke,<br>wie z.B.<br>/.../ oder<br>?...? bei <i>ed</i> | - ihm kein Gegenschrägstrich \ vorangestellt ist.                                                                                                                                   |

Tabelle 11: Sonderzeichen in regulären Ausdrücken

ERWEITERTE REGULÄRE AUSDRÜCKE

Erweiterte reguläre Ausdrücke umfassen die einfachen regulären Ausdrücke mit folgender Ausnahme:

Die bei einfachen regulären Ausdrücken verwendete Konstruktion  $\backslash(\dots\backslash)$  hat bei erweiterten regulären Ausdrücken *keine* Sonderbedeutung, z.B. steht der erweiterte reguläre Ausdruck  $\backslash(ab\backslash)$  für die Zeichenkette  $(ab)$ .

Darüberhinaus bieten erweiterte reguläre Ausdrücke die folgenden zusätzlichen Syntaxelemente zur Bildung von Mustern:

| Nr. | regulärer Ausdruck | Bedeutung                                                                                                 | Beispiel                | passende Zeichenketten                      |
|-----|--------------------|-----------------------------------------------------------------------------------------------------------|-------------------------|---------------------------------------------|
| 13  | $r^+$              | Ein- oder mehrmals der reguläre Ausdruck $r$ . $r$ muß von der Form 1 - 5, 15 oder 16 sein.               | $u^+$                   | $u, uu, uuu, \dots$                         |
| 14  | $r^?$              | Null- oder einmal der reguläre Ausdruck $r$ . $r$ muß von der Form 1 - 5, 15 oder 16 sein.                | $u^?$                   | nichts oder $u$                             |
| 15  | $(r)$              | Zeichenketten, die zu dem regulären Ausdruck $r$ passen. $r$ kann ein beliebiger regulärer Ausdruck sein. | $(ok(abc))$<br>$(au)^*$ | $okabc$<br>nichts oder<br>$au, auau, \dots$ |
| 16  | $(r1 r2)$          | Zeichenketten, die zu dem regulären Ausdruck $r1$ oder zu dem regulären Ausdruck $r2$ passen.             | $(ok ko)$               | $ok$ oder $ko$                              |

Tabelle 12: Bildung von Mustern in regulären Ausdrücken

INTERNATIONALISIERTE REGULÄRE AUSDRÜCKE

Einfache internationalisierte bzw. erweiterte internationalisierte reguläre Ausdrücke verwenden dieselben Elemente wie die oben beschriebenen einfachen bzw. erweiterten regulären Ausdrücke, bieten aber zusätzlich die folgenden Möglichkeiten, um Ausdrücke innerhalb eckiger Klammern [...] zu bilden:

| Ausdruck     | Bedeutung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| [[:klasse:]] | <p>(Ausdruck für Zeichenklasse) Ein Zeichen der Zeichenklasse <i>klasse</i>. Die Zeichenklassen werden in einer internationalisierten Umgebung durch die Umgebungsvariablen <i>LC_CTYPE</i> bzw. <i>LANG</i> definiert (siehe "SINIX V5.41, Kommandos, Band 1, 3 Internationale Umgebung - NLS").</p> <p><i>klasse</i> kann sein:</p> <p><b>alpha</b> alle Buchstaben<br/> <b>upper</b> alle Großbuchstaben<br/> <b>lower</b> alle Kleinbuchstaben<br/> <b>digit</b> alle Dezimalziffern (0 bis 9)<br/> <b>xdigit</b> alle Hexadezimalziffern (0 bis 9, a bis f und A bis F)<br/> <b>alnum</b> alle alphanumerischen Zeichen (Buchstaben und Ziffern)<br/> <b>space</b> alle Zeichen, die bei der Textdarstellung Zwischenraum produzieren (z.B. Leer- oder Tabulatorzeichen)<br/> <b>punct</b> alle Trennzeichen<br/> <b>print</b> alle abdruckbaren Zeichen (einschließlich der Zeichen in <i>space</i>)<br/> <b>graph</b> alle sichtbaren abdruckbaren Zeichen (ohne die Zeichen in <i>space</i>)<br/> <b>cntrl</b> alle Steuerzeichen</p> <p>Beispiele</p> <p>In der deutschen Umgebung <i>LANG=LC_CTYPE=De_DE.88591</i> passen zum regulären Ausdruck <code>[[:lower:]]</code> die Zeichen <i>ä, ö, ü, ß, a, ...</i>, z. Das Zeichen (<i>e</i> mit Akzent Gravis) gehört weder zur Klasse <i>lower</i> noch zur Klasse <i>alpha</i>, sondern ist nur ein abdruckbares Zeichen.</p> <p>In der französischen Umgebung <i>LANG=LC_CTYPE=Fr_FR.88591</i> gehört zu den Klassen <i>lower</i> und <i>alpha</i>, <i>ä</i> jedoch nicht.</p> |
| [=c=]        | <p>(Ausdruck für Äquivalenzklasse) Ein Zeichen bzw. eine Zeicheneinheit, die in der ASCII-Sortierreihenfolge die gleiche relative Ordnung wie <i>c</i> hat. Für <i>c</i> dürfen Sie kein Gleichheitszeichen = und keine schließende eckige Klammer ] angeben. Die Sortierreihenfolge wird in einer internationalisierten Umgebung durch die Umgebungsvariablen <i>LC_COLLATE</i> bzw. <i>LANG</i> definiert (siehe "SINIX V5.41, Kommandos, Band 1, 3 Internationale Umgebung - NLS").</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

| Ausdruck | Bedeutung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|          | <p><b>Beispiele</b></p> <p>In der deutschen Umgebung <code>LANG=LC_COLLATE=De_DE.88591</code> bilden die Zeichen <code>u</code> und <code>ü</code> eine Äquivalenzklasse. Folglich passen zum regulären Ausdruck <code>[[=u=]]</code> die Zeichen <code>u</code> und <code>ü</code>. Die regulären Ausdrücke <code>[[=u=]v]</code>, <code>[[=ü=]v]</code> und <code>[uüv]</code> sind in dieser Umgebung gleichbedeutend.</p> <p>In der französischen Umgebung <code>LANG=LC_COLLATE=Fr_FR.88591</code> passen zum regulären Ausdruck <code>[[=e=]]</code> das Zeichen <code>e</code> sowie alle akzentuierten <code>e</code>.</p>                                                                        |
| [.cc.]   | <p>(Zeicheneinheits-Symbol) Zeicheneinheiten, die aus mehreren Zeichen bestehen, müssen in dieser Form dargestellt werden, um sie von einfachen Zeichen zu unterscheiden. Beim Sortieren wird ein solcher Ausdruck als ein einziges Zeichen betrachtet. <code>cc</code> muß in der ASCII-Sortierreihenfolge als zulässige Zeicheneinheit definiert sein. Ist <code>cc</code> keine zulässige Zeicheneinheit, so ist <code>[.cc.]</code> ein ungültiger Ausdruck. Die Sortierreihenfolge wird in einer internationalisierten Umgebung durch die Umgebungsvariablen <code>LC_COLLATE</code> bzw. <code>LANG</code> definiert (siehe "SINIX V5.41, Kommandos, Band 1, 3 Internationale Umgebung - NLS").</p> |
|          | <p><b>Beispiel</b></p> <p>In der spanischen Umgebung <code>LANG=LC_COLLATE=Es_SP.88591</code> ist <code>ch</code> eine zulässige Zeicheneinheit: <code>ch</code> ist im Spanischen ein eigener Buchstabe und wird zwischen <code>c</code> und <code>d</code> sortiert. Dieser Buchstabe muß in der Form <code>[.ch.]</code> dargestellt werden, um ihn von der zweibuchstabigen Zeichenkette <code>ch</code> zu unterscheiden. Siehe auch Beispiele zu <code>c1-c2</code>.</p>                                                                                                                                                                                                                            |
| c1-c2    | <p>Ein Zeichen aus dem Bereich von <code>c1</code> bis <code>c2</code>, gemäß der aktuell gültigen Sortierreihenfolge (Grenzen <code>c1</code> und <code>c2</code> eingeschlossen). Im Unterschied zu nicht-internationalisierten regulären Ausdrücken können <code>c1</code> und <code>c2</code> auch Ausdrücke für Äquivalenzklassen <code>[=c=]</code> oder Zeicheneinheits-Symbole <code>[.cc.]</code> sein.</p>                                                                                                                                                                                                                                                                                      |
|          | <p><b>Beispiele</b></p> <p>In der deutschen Umgebung <code>LANG=LC_COLLATE=De_DE.88591</code> sind die Umlaute folgendermaßen in die Sortierreihenfolge eingereiht:<br/> <code>a, ä, b, c, ..., n, o, ö, p, ..., t, u, ü, v, ...</code><br/> Umlaut und zugehöriger Vokal bilden jeweils eine Äquivalenzklasse (siehe Beispiel zu <code>[=c=]</code>).</p>                                                                                                                                                                                                                                                                                                                                                |

## Reguläre Ausdrücke

| Ausdruck | Bedeutung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|          | <p>Zeichenbereiche werden deshalb wie folgt ausgewertet:</p> <p>regulärer Ausdruck    passende Zeichen</p> <p>[t-v]                    t, u, ü, v</p> <p>[t-u]                    t, u</p> <p>[t-ü]                    t, u, ü</p> <p>[t-[=u=]]                t, u, ü</p> <p>[^t-[=u=]]                beliebiges Zeichen außer t, u, ü</p> <p>In der französischen Umgebung <i>LANG=LC_COLLATE=Fr_FR.88591</i> passen zum regulären Ausdruck [a-f] die Zeichen a, b, c, , d, e, alle akzentuierten e, f.</p> <p>In der spanischen Umgebung <i>LANG=LC_COLLATE=Es_SP.88591</i> ist <i>ch</i> eine zulässige Zeicheneinheit: <i>ch</i> ist im Spanischen ein eigener Buchstabe und wird zwischen <i>c</i> und <i>d</i> sortiert (siehe [.cc.]). Zeichenbereiche werden deshalb wie folgt ausgewertet:</p> <p>regulärer Ausdruck    passende Zeichen</p> <p>[a-f]                    a, b, c, ch, d, e, f</p> <p>[b[.ch.]-eg]            b, ch, d, e, g</p> |

Tabelle 13: Internationalisierte reguläre Ausdrücke

### Sonderzeichen

|               |                                                                                     |
|---------------|-------------------------------------------------------------------------------------|
| Sonderzeichen | Das linksstehende Zeichen ist in eckigen Klammern [...] ein Sonderzeichen, wenn ... |
| :             | - es Teil eines Zeichenklassen-Ausdrucks [:klasse:] ist.                            |
| =             | - es Teil eines Äquivalenzklassen-Ausdrucks [=c=] ist.                              |
| .             | - es Teil eines Zeicheneinheits-Symbols [.cc.] ist.                                 |
| -             | - es in den eckigen Klammern nicht an erster oder letzter Stelle steht.             |
| ^             | - es der öffnenden eckigen Klammer [ unmittelbar folgt.                             |

Tabelle 14: Sonderzeichen in internationalisierten regulären Ausdrücken

**PRIORITÄT**

Die folgende Tabelle zeigt die Priorität der Operatoren in regulären Ausdrücken.

| Operator      | Priorität            |
|---------------|----------------------|
| [...] (...)   | höchste Priorität    |
| * ? + \{...\} | .                    |
| Verkettung    | .                    |
|               | niedrigste Priorität |

Tabelle 15: Priorität der Operatoren in regulären Ausdrücken

**Beispiele**

1. Einfache reguläre Ausdrücke

| Muster   | Bedeutung                                                | passende Zeichenketten |
|----------|----------------------------------------------------------|------------------------|
| ab.d     | a - b - ein beliebiges Zeichen - d                       | abcd, abXd, ab*d, ...  |
| ab.*d    | a - b - beliebige Zeichenkette (kann auch leer sein) - d | abd, abxd, abX*Yd, ... |
| ab[xyz]d | a - b - entweder x oder y oder z - d                     | abxd, abyd, abzd       |
| ab[^c]d  | a - b - beliebiges Zeichen ungleich c - d                | abbd, abXd, ab*d, ...  |
| ^abcd\$  | eine Zeile, die nur die Zeichenkette abcd enthält        |                        |

2. Erweiterte reguläre Ausdrücke

| Muster    | Bedeutung                                                          | passende Zeichenketten |
|-----------|--------------------------------------------------------------------|------------------------|
| ab.+d     | a - b - beliebige Zeichenkette aus einem oder mehreren Zeichen - d | abjd, abX*Yd, ...      |
| abc?d     | a - b - c oder nichts - d                                          | abd, abcd              |
| (abc xyz) | abc oder xyz                                                       | abc, xyz               |



---

## Literatur

- [ 1] MX500 (SINIX V5.40)  
MX300 (SINIX V5.41)  
**Leitfaden für Systemverwalter**  
Beschreibung
- Zielgruppe*  
Systemverwalter
- Inhalt*  
Einführung in die Systemverwaltung von SINIX-Systemen Anleitung zur Konfiguration und Wartung des SINIX-Systems
- Bestell-Nr.: U6406-J-Z145-1
- [ 2] SINIX V5.41  
**Bediensystem für Systemverwalter**  
Beschreibung
- Zielgruppe*  
Systemverwalter
- Inhalt*  
Das Bediensystem für Systemverwalter umfaßt unter anderem die Verwaltung der lokalen Endgeräte, der Benutzerkennungen, des Spools und der Post. Die Beschreibung wurde für SINIX V5.41 überarbeitet.
- Bestell-Nr.: U6391-J-Z145-2
- [ 3] SINIX V5.40  
**Netzwerke - Leitfaden für Benutzer und Verwalter**  
Benutzerhandbuch
- Zielgruppe*  
– Benutzer  
– Netzverwalter
- Inhalt*  
Beschreibung von TCP/IP, NFS und DFS.
- Bestell-Nr.: U6402-J-Z95-1

- [ 4] SINIX V5.40/5.41  
**Netzwerke - Leitfaden für Benutzer und Verwalter**  
MX300, MX500, RM400, RM600  
Ergänzung
- Zielgruppe*
- Netzverwalter
  - Systemverwalter
  - Nicht privilegierte Benutzer
- Inhalt*
- Ergänzungen zum Handbuch "Netzwerke: Leitfaden für Benutzer und Verwalter (Beschreibung)", wie z.B. Installation und Bedienung der LAN-Anbindung über das Bediensystem.
- Bestell-Nr.: U6599-J-Z815-2
- [ 5] SINIX V5.40  
**COLLAGE-Bediensystem**  
Benutzerhandbuch
- Zielgruppe*
- Anwender
- Inhalt*
- Ausführliche Beschreibung der Bestandteile des COLLAGE-Bediensystems, der Grundlagen der Bedienung sowie der Möglichkeiten zur Anpassung der Oberfläche an individuelle Anforderungen.
- Bestell-Nr.: U6392-J-Z95-1
- [ 6] **COLLAGE V4.0**  
**Bedienen und Verwalten**  
Benutzerhandbuch
- Zielgruppe*
- Dieses Handbuch richtet sich an Benutzer und Systemverwalter des Fenster-Verwaltungssystems COLLAGE.
- Inhalt*
- Das Benutzerhandbuch enthält eine Einführung zu COLLAGE für Anfänger. Die Abschnitte über die Verwaltung von COLLAGE enthalten wertvolle Informationen für den Systemverwalter.
- Bestell-Nr.: U3004-J-Z95-6

- [ 7] **CES (SINIX)**  
**Leitfaden und Werkzeuge für die Programmierung mit C**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an C-Programmierer, die unter dem Betriebssystem SINIX arbeiten.

*Inhalt*

Im Handbuch werden das C-Übersetzungssystem (Präprozessor, Binder, Include-Dateien, Bibliotheken) und Dienstprogramme zur Entwicklung, Verwaltung, Wartung und Generierung von C-Programmen beschrieben.

Bestell-Nr.: U6296-J-Z145-1

- [ 8] **CES (SINIX)**  
**cc-Kommando für MX300 und WX200**  
Beschreibung

*Zielgruppe*

C-Programmierer, die unter dem Betriebssystem SINIX auf einem MX300 oder einem WX200 arbeiten.

*Inhalt*

Das Taschenbuch enthält die vollständige Beschreibung des cc-Kommandos mit Optionen für CES unter SINIX auf MX300 und WX200.

Bestell-Nr.: U6301-J-Z145-1

- [ 9] **CES V5.40/V5.41**  
**Referenzhandbuch für Programmierer**  
Beschreibung

*Zielgruppe*

C-Programmierer, die unter SINIX V5.40 oder SINIX V5.41 arbeiten.

*Inhalt*

Beschreibung der Kommandos für die Programmentwicklung, der Systemaufrufe und Funktionen sowie C-spezifischer Dateiformate.

Bestell-Nr.: U6401-J-Z95-1

- [10] SINIX V5.41  
**Asynchronous I/O Reference Manual  
RM600, RM400  
Reference Manual**
- Zielgruppe*  
– Systemadministrator  
– Benutzer  
– Systemprogrammierer
- Inhalt*  
Beschreibt die Installation und Anwendung der Siemens Nixdorf AIO (Asynchronous I/O).
- Bestell-Nr.: U8255-J-Z815-1-7600
- [11] SINIX V5.40 (MX500)  
SINIX V5.41 (MX300/WX200)  
**Referenzhandbuch für Systemverwalter**
- Zielgruppe*  
Systemverwalter
- Inhalt*  
Beschreibt Kommandos und Anwendungsprogramme zur Systempflege sowie Dateiformate, spezielle Dateien zur Systemverwaltung und gibt Diagnosehinweise.
- Bestell-Nr.: U8888-J-Z145-1
- [12] SINIX V5.41  
**Kommandos Band 1, A - K  
Beschreibung**
- Zielgruppe*  
SINIX-Anwender mit Grundkenntnissen
- Inhalt*  
Nachschlagewerk für die Benutzerkommandos (A - K) des Betriebssystems SINIX V5.41.
- Bestell-Nr.: U8480-J-Z145-1

- 
- [13] SINIX V5.41  
**Kommandos Band 2, L - Z**  
Beschreibung
- Zielgruppe*  
SINIX-Anwender mit Grundkenntnissen
- Inhalt*  
Nachschlagewerk für die Benutzerkommandos (L - Z) des Betriebssystems SINIX V5.41.
- Bestell-Nr.: U8481-J-Z145-1
- [14] SINIX V5.41  
**Kommandos Band 3**  
Tabellen und Verzeichnisse  
Beschreibung
- Zielgruppe*  
SINIX-Anwender mit Grundkenntnissen
- Inhalt*  
Nachschlagewerk  
Band 3 enthält ein Gesamtinhaltsverzeichnis, eine Tabellensammlung (Reguläre Ausdrücke, Gerätedateien für Datenträger u.a.), ein Fachwort-, ein Literatur- und ein Stichwortverzeichnis.
- Bestell-Nr.: U8482-J-Z145-1
- [15] SINIX V5.41  
**Leitfaden für Benutzer**  
Benutzerhandbuch
- Zielgruppe*  
Benutzer
- Inhalt*  
Beschreibung der wesentlichen Elemente des SINIX Betriebssystems. Dazu gehört: Einführung in die Benutzung von SINIX, das Dateisystem, die Prozessverarbeitung, die Shell.
- Bestell-Nr.: U6404-J-Z145-2

- [16] DPTG-2 V1.0  
**Terminal-Multiplexer-Protokoll,  
Installationsanleitung**

*Zielgruppe*

Systemverwalter

*Inhalt*

Beschreibung der Installation der DPTG-2-Software und der Konfiguration der über DPTG-2 betriebenen Peripheriegeräte.

Bestell-Nr.: 08621.00.0.93

### **Bestellen von Handbüchern**

Die aufgeführten SINIX-Handbücher finden Sie mit Ihren Bestellnummern im *Druckschriftenverzeichnis* der Siemens Nixdorf Informationssysteme AG. Neu erschienene Titel finden Sie in den *Druckschriften-Neuerscheinungen*.

Beide Veröffentlichungen erhalten Sie regelmäßig, wenn Sie in den entsprechenden Verteiler aufgenommen sind. Wenden Sie sich bitte hierfür an Ihre zuständige Geschäftsstelle. Dort können Sie auch die Handbücher bestellen.

---

# Stichwörter

## A

- acctcom 63
- addbad(1M) 125
- Alpha-COLLAGE 40
  - in einem Shell-Fenster arbeiten 40
  - unterdrücken der Ausgabe 40
- ar 64
- asynchrones Schreiben über NFS 28
- Asynchronous Raw I/O 52
- Ausdrücke, reguläre 173
- awk 65

## B

- BA80-Terminals konfigurieren 21
- bc 66
- Benutzerkennungen sichern 15
- bind(1M) 126
- Block-Device-Geräte Dateien 17
- boot(1M) 128

## C

- cal 69
- CES
  - #pragma weak 45
  - ABI386-Konformität 42
  - Optionen zur Codegenerierung 43
  - Optionen zur Optimierung 43
  - Präprozessor-Anweisung 45
- Cluster-Bildschirm 10
- COLLAGE
  - Kopfzeile drucken 38
  - neuen Editor eintragen 39
- colltbl(1M) 135
- Community 30
- Community-Name 30

copy 70  
cpio 71  
crypt 72  
csh 73

### D

df 74  
diff 75  
diff3 75  
dirname 76  
Disketten, 2,88 MB 13  
DKIOCAIOVERS 53  
DKIOCASTAT 54  
DKIOCASTRT 53  
DOS-Partition 168  
Drucker, am TACLAN verwalten 11

### E

edvtoc(1M) 136  
Ethernet-Prozessor 24  
ETHERNETFLAGS 26  
etherstat 26  
exa(7) 165

### F

fdisk-Partitionstabelle 168  
Festplatte, nachträgliche Installation 16  
fchk 77  
Fortran77, Komponenten 46  
fsck 17

### G

gated(1M) 138  
Gateway 138  
Gerätedatei  
  dev/exa0 20  
  dev/exa8 20  
  dev/exa8r 20  
  dev/exar 20  
eigene einrichten 16  
Gerätedateien 20  
  blockorientiert 17  
  für Datenträger 17  
  zeichenorientiert 17

**Gerätetreiber-Datei**

exa(7) 165

shd(7) 168

gettxt 78

**I**

ifconfig(1M) 26

inet-Konfigurationsdatei 25, 26

INTERFACES 25, 26

internationalisierter regulärer Ausdruck 180

iocntl-Aufruf 165

**K****Kommando**

acctcom 63

addbad(1M) 125

ar 64

awk 65

bc 66

bind(1M) 126

boot(1M) 128

cal 69

colltbl(1M) 135

copy 70

cpio 71

crypt 72

csh 73

df 74

diff 75

diff3 75

dirname 76

edvtoc(1M) 136

flchk 77

gated(1M) 138

gettxt 78

ksh 79

login 80

lp 81

lpr 81

lpstat 81

ls 82

mail 83

mailx 83

### Kommando

- mt 84
- mv 85
- newgrp 86
- passwd 87
- pr 88
- ps 89
- rm 90
- sar 91
- secure 92
- sed 95
- sh 96
- shl 96
- smtper(1M) 157
- sort 97
- swap(1M) 158
- tail 98
- talk 99
- tar 100
- tcpdump 108
- time 101
- timex 101
- traceroute(1M) 159
- uniq 102
- vi 103
- wc 105
- wcheck 106
- who 107

Konfigurationsdatei, inet 25, 26

konfigurieren

- Konsole 8

- Post&Wait-Treiber 49

- SX-Board 9

Konsole konfigurieren 8

ksh 79

### L

login 80

loginsave 15

lp 81

lpr 81

lpstat 81

ls 82

**M****Magnetband-Kassetten-Laufwerk**

2,3 Gbyte 20

60/150 Mbyte 19

**Magnetbandgerät 1/2 Zoll**

FS1000 21

FS2000 21

mail 83

mailx 83

**Master-Prozeß** 49, 50

mmap() 51

mt 84

**Muster** 173

mv 85

**N****Nachrichten, in Warteschlangen einreihen** 157**Netzüberwachung** 29

newgrp 86

**NFS** 28**P****Partition** 168, 169**Partitionstabelle** 168

passwd 87

**PIF-Dateien** 39

pkginfo 19

**Plattenslices** 18**Post&Wait-Treiber** 47

Konfiguration 49

Schnittstellen 48

pr 88

**Protokoll**

Ausgabe 140

Optionen 138

Routing-Protokoll 138

ps 89

**Pseudo-TTYs** 17**R****regulärer Ausdruck** 173

einfacher 175

erweiterter 179

internationalisierter 180

Priorität 183

Sonderzeichen 178  
Überblick 174  
rm 90  
Routing-Dämon 138  
Routing-Protokoll 138  
Run-Level, ändern 19

**S**

sar 91  
SCSI  
-Band am NCR Host Adapter 165  
-Bus 168  
-Controller 169  
-Hostadapter 136  
-Targetnummer 166, 168  
secure 92  
sed 95  
sh 96  
shd(7) 168  
shl 96  
sichern, der Benutzerkennungen 15  
SIGHUP 139  
SIGINT 139  
SIGUSR1 139  
Slave 50  
Slice 136, 168  
slip, Konfigurierung 26  
SMTP 157  
smtpqer(1M) 157  
SNMP 29  
SNMP-Agent  
Funktion 29  
Installation 31  
Rekonfiguration 34  
Start 33  
Stop 33  
Softwareprodukt, Informationen ausgeben 19  
Sonderzeichen (reguläre Ausdrücke) 182  
sort 97  
swap(1M) 158  
SX-Board konfigurieren 9  
synchrones Schreiben über NFS 28

**T**

tail 98  
talk 99  
tar 100  
TCP-Portnummer 12  
tcpdump(1) 108  
time 101  
time-Variable 50  
timex 101  
traceroute(1M) 159  
Treiber, Post&Wait 47  
Treiberfunktion mmap() 51

**U**

uname 26  
uniq 102

**V**

Variable  
    INTERFACES 25, 26  
    time 50  
vi 103  
VTOC 136, 168

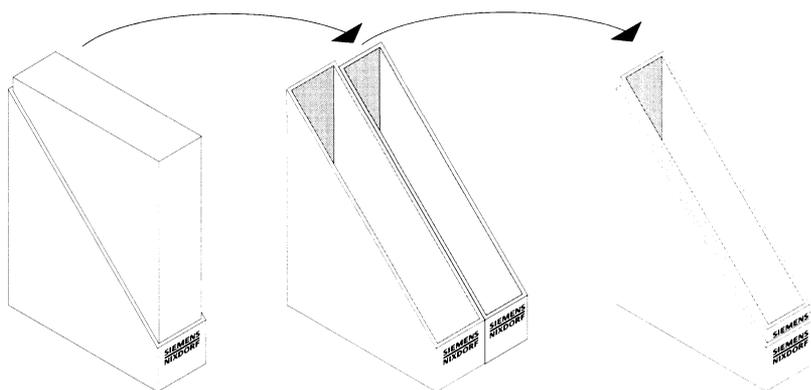
**W**

wc 105  
wcheck 106  
who 107



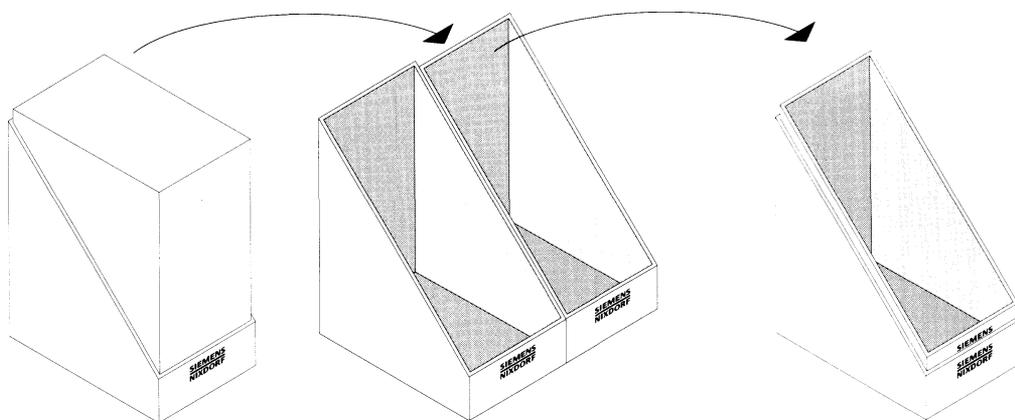
# Sammelboxen

Für Handbücher des vorliegenden Formates bieten wir zweiteilige Sammelboxen in zweierlei Größen an. Der Bestellvorgang entspricht dem für Handbücher.



Breite: ca. 5 cm

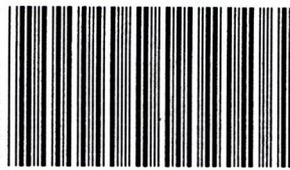
Bestellnummer: U3775-J-Z18-1



Breite: ca. 10 cm

Bestellnummer: U3776-J-Z18-1

960771 P/MCD



9Y504253

Herausgegeben von/Published by  
Siemens Nixdorf Informationssysteme AG  
Postfach 21 60, W-4790 Paderborn  
Postfach 83 09 51, W-8000 München 83

Bestell-Nr./Order No. **U20244-J-Z145-1**  
Printed in the Federal Republic of Germany  
3540 AG 12923. (4430)