
3 Dienste, Programme und Skripte

In diesem Kapitel werden die wichtigsten Teile des Samba-Paketes (Dienste, Programme und Skripte) kurz vorgestellt. Dabei erfahren Sie auch, mit welchen Parametern Sie die Programme aufrufen können. Weitere Informationen finden Sie auf den entsprechenden Manual-Seiten oder im Verzeichnis *doc* der jeweiligen Samba-Installation, also bei SuSE beispielsweise */usr/doc/packages/samba/doc*.

3.1 Die Dienste *smbd* und *nmbd*

3.1.1 *smbd*

Das Programm *smbd* stellt den Server dar, der die gemeinsame Nutzung von Dateisystemen bzw. Dateien und Druckern mit Windows-Clients erlaubt und auch verwaltet. Dabei wird das SMB-Protokoll verwendet, das mit dem LanManager-Protokoll kompatibel ist. Der Server sollte als Dämon im Hintergrund laufen, wofür beim Aufruf die Option *-D* erforderlich ist. Allerdings geschieht dies in der Regel automatisch, wenn Sie den Start des Servers über das Administrationsprogramm wie beschrieben eingerichtet haben. Deshalb wird Ihnen auch in der Liste, die Sie mit dem Kommando *ps aux* aufrufen, der Befehl */usr/sbin/smbd -D* angezeigt.

Zum Betrieb des Servers ist die Konfigurationsdatei *smb.conf* notwendig, die alle Angaben über die bereitzustellenden Dienste und Clients enthält. Dabei wird für jeden Client eine Kopie des Servers gestartet, wenn dieser eine Anfrage an den Server tätigt. Diese Kopie bearbeitet daraufhin solange alle Anfragen und Verbindungen des Clients, bis sie geschlossen werden. Erst danach wird auch die Kopie beendet.

Die Konfigurationsdatei wird von *smbd* automatisch neu geladen, wenn sie verändert worden ist. Dies wirkt sich aber nicht auf bereits bestehende Verbindungen aus. Sollen auch diese von den veränderten Einstellungen profitieren, müssen sie geschlossen und danach erneut aufgebaut werden, oder der Server muss neu gestartet werden – was aber letztendlich auf dasselbe hinausläuft.

Der Server schreibt standardmäßig eine Logdatei mit dem Namen *log.smb* im Verzeichnis */var/log* (zumindest bei der SuSE-Distribution, bei anderen müssen

Sie gegebenenfalls nach dieser Datei suchen). Dort finden Sie beispielsweise weitere Informationen über die aufgebauten Verbindungen und wann der Server jeweils gestartet wurde. Da in dieser Datei auch sicherheitsrelevante Daten enthalten sein könnten, sollte dieses Verzeichnis nur der root-Benutzer bzw. der Administrator lesen können. Gegebenenfalls müssen Sie die Zugriffsrechte entsprechend anpassen, wenn Sicherheitsaspekte auf Ihrem Rechner eine Rolle spielen ...

3.1.2 nmbd

Das Programm *nmbd* stellt einen Server dar, der NetBIOS-Namens-Service Anfragen versteht und beantworten kann, der also Clients hilft, Server zu lokalisieren. Empfängt der Server eine Namesanfrage und ist darin der eigene Name enthalten, wird die IP-Adresse des Rechners, auf dem dieser läuft, zurückgegeben. Dabei ist der eigene Name derjenige, der normalerweise auch für die anderen Netzwerkdienste verwendet wird (das lässt sich bei Bedarf ändern). In der Regel wird dieser Name erstmals bei der Installation des Systems festgelegt.

Zusätzlich kann das Programm auch als WINS-Server (Windows Internet Name Server) verwendet werden; in diesem Fall wird eine Datenbank mit Namen erzeugt. Auf diese kann bei Bedarf – also bei Anfragen durch Clients – zurück gegriffen werden.

Der Server sollte als Dämon im Hintergrund laufen. Dafür ist beim Aufruf die Option *-D* erforderlich. Dies passiert in der Regel automatisch, wenn Sie den Start des Servers über das Administrationsprogramm wie beschrieben eingerichtet haben. Deshalb wird Ihnen auch in der Liste, die Sie mit dem Kommando *ps aux* aufrufen, der Befehl */usr/sbin/nmbd -D* angezeigt.

Neben der Konfigurationsdatei *smb.conf* kann auch die Datei *lmhosts* verwendet werden, die Zuordnungen zwischen NetBIOS-Namen und IP-Adressen enthält. Allerdings wird diese Datei nicht zur Auflösung von Namensanfragen von außen verwendet, sie ist nur für den lokalen Gebrauch bestimmt.

Wie beim Dienst *smbd* wird auch für den Dienst *nmbd* eine Logdatei geschrieben, die den Namen *log.nmb* hat. Sie finden sie beispielsweise bei der SuSE-Distribution im Verzeichnis */var/log*, bei anderen Distributionen müssen Sie gegebenenfalls nach dieser Datei suchen. Auch hier gilt, dass die Datei sicherheitsrelevante Informationen wie die Daten einer aufgebauten Verbindung enthält. Deshalb müssen Sie sie durch passendes Setzen der Zugriffsrechte besonders schützen, sodass nur der Administrator auf sie zugreifen kann.

3.2 Programme und Skripte

3.2.1 nmblookup

Mit Hilfe des Programms *nmblookup* können NetBIOS-Namen abgefragt und in IP-Adressen umgesetzt werden, wozu NetBIOS über TCP/IP Anfragen verwendet werden. Diese Anfragen können dann direkt in einen speziellen IP Rundsendungsbereich oder an einen speziellen Rechner gesendet werden. Sie finden das Programm – zumindest bei der SuSE-Distribution – im Verzeichnis */usr/bin*.

Beim Aufruf muss immer der NetBIOS-Name spezifiziert werden, dessen IP-Adresse gefragt ist. Allerdings kann dies manchmal auch eine IP-Adresse sein, abhängig von den Parametern, die im Folgenden beschrieben werden. Wird ein NetBIOS-Name angegeben, so kann sein Typ durch Anhängen eines Doppelkreuzes, gefolgt von einem zweistelligen Hexadezimalwert, spezifiziert werden. Der NetBIOS-Name kann auch aus einem Stern ('*') bestehen, wobei dann alle registrierten Namen innerhalb eines Rundsendungsbereiches zurückgegeben werden.

Parameter	Beschreibung
-A	Der beim Befehl anzugebende Name wird als IP-Adresse interpretiert. Außerdem sollen auch noch die NetBIOS-Namen zurück gegeben werden, die von einem Host registriert worden sind.
-B Adresse	Die Anfrage soll an die angegebene Rundsendeadresse geschickt werden. Ohne die Angabe dieser Option wird die Anfrage an die Rundsendeadresse der primären Netzwerkschnittstelle gesendet. Diese Adresse wird entweder automatisch bestimmt oder sie ist definiert in der Konfigurationsdatei <i>smb.conf</i> über den Parameter <i>interfaces</i> .
-M	Sucht nach einem Master Browser, wobei nach einem NetBIOS-Namen mit einem Typ von 0x1d gesucht wird. Besteht der Name aus einem Minuszeichen ('-'), wird nach dem speziellen Namen <i>__MSBROWSE__</i> gesucht.
-r	UDP-Datagramme (User Datagram Protocol) sollen über den Port 137 gesendet und empfangen werden. Der Grund für diese Option ist ein Fehler in Windows 95, bei der der Quellport des anfragenden Paketes ignoriert wird. Dafür wird dann aber auf dem UDP-Port 137 geantwortet. Unglücklicherweise sind bei einem Linux-System Administrator-

(Fortsetzung nächste Seite)

Parameter	Beschreibung
	Rechte erforderlich, um diesen Port verwenden zu können und zusätzlich setzt auch der Dämon <i>nmbd</i> diesen Port ein.
-R	Es soll eine rekursive Suche durchgeführt werden. Diese Option sollte normalerweise verwendet werden, wenn eine Anfrage bei einem WINS-Server erfolgt und seine Namen bestimmt werden sollen. Wird diese Option nicht verwendet, wird der normale NetBIOS-Code auf dem Rechner verwendet.
-S	Nachdem die Anfrage eine IP-Adresse zurückgeliefert hat, sollen auch noch die NetBIOS-Namen zurückgegeben werden, die von einem Host registriert worden sind.
-T	Für jede gefundene Adresse wird noch in einer DNS-Tabelle nach dem entsprechenden DNS-Namen gesucht. Dieser Name wird noch vor dem Paar „ <i>IP-Adresse NetBIOS-Name</i> “ ausgegeben.
-U Adresse	Es erfolgt eine „Unicast“-Anfrage an die angegebene Adresse oder einen entsprechenden Rechner. Diese Option wird nur im Zusammenhang mit der Option <i>-R</i> verwendet, um einen WINS-Server abzufragen.

Soll beispielsweise eine Anfrage an einen WINS-Server erfolgen, so müssten Sie eingeben:

```
nmblookup -U Server -R Name
```

Der String *Server* beschreibt den Namen des WINS-Servers und *Name* den Namen des Rechners, nach dem gesucht werden soll.

3.2.2 smbclient

Das Programm *smbclient* ist ein ftp-artiger Unix-Client, um auf SMB-Server zugreifen zu können. Dies gilt auch, wenn der Server als Dämon auf dem eigenen Rechner läuft. Somit lässt sich mit dem Programm also testen, ob der Server korrekt funktioniert.

Beim Einsatz des Programms ist die Angabe diverser Parameter notwendig bzw. möglich. Zwingend erforderlich ist die Angabe eines Servicenamens, der spezifiziert, welcher Dienst auf dem Server verwendet werden soll. Dieser Name ist folgendermaßen aufgebaut:

```
//SMBServer/Service
```

wobei *SMBServer* den Namen des jeweiligen Servers und *Service* den entsprechenden Dienst beschreibt. Dabei muss der Name des Servers nicht notwendigerweise der Name des Rechners sein, aber in der Regel sollten sie identisch sein.

Zusätzlich können Sie auch noch ein Passwort angeben, das erforderlich ist, um auf den entsprechenden Dienst auf dem jeweiligen Server zuzugreifen. Dabei gibt es kein Standardpasswort, sodass immer eines angegeben werden muss, entweder als Parameter oder auf Nachfrage hin. Die letztere Möglichkeit ist übrigens ausgeschaltet, sofern ein Passwort als Parameter spezifiziert wird.

Insgesamt sollten Sie aber bei der Angabe von Passwörtern in Skripten oder als Kommandozeilenparameter vorsichtig sein, da diese von jedem gelesen werden können. Geben Sie hingegen ein Passwort erst bei Nachfrage ein, werden die einzelnen Buchstaben nicht im Klartext angezeigt, sodass sie nicht so ohne weiteres lesbar sind.

Beachten Sie außerdem, dass das Passwort bei einigen Servern, wie beispielsweise OS/2 oder Windows für Workgroups, aus Großbuchstaben bestehen sollte. Geben Sie hier Kleinbuchstaben ein oder werden Klein- und Großbuchstaben gemischt, könnte der Server die entsprechende Anfrage zurückweisen.

Neben anderen Parametern kann noch die Angabe von *-L* gefolgt von einem Servernamen wichtig sein. Mit diesem Parameter können Sie sich anzeigen lassen, welche Dienste und Verzeichnisse verfügbar sind. Werden Sie dabei nach einem Passwort gefragt, drücken Sie einfach die Eingabetaste, da hierfür normalerweise ein Passwort nicht erforderlich ist.

Wurde die Verbindung zu einem SMB-Server erfolgreich hergestellt, lässt sich *smbclient* in etwa so bedienen wie das Programm *ftp*. Zu den verfügbaren Operationen gehören beispielsweise das Laden von Dateien des Servers auf den eigenen Rechner, die Bestimmung von Informationen über ein Verzeichnis (Inhaltsverzeichnis) oder auch das Kopieren von lokalen Dateien zum Server hin (Abbildung 3.1).

Dazu wird ein Prompt angezeigt, der so aussieht:

```
smb: \>
```

Das Zeichen `\` beschreibt dabei das aktuelle Arbeitsverzeichnis auf dem Server. Wechseln Sie das Verzeichnis, sehen Sie hier auch eine andere Ausgabe, beispielsweise:

```
smb: \.x\fm\>
```

Hier wäre also das aktuelle Arbeitsverzeichnis auf dem Server *.x\fm*.

Sehen Sie den Prompt, ist der Client bereit, weitere Aktionen auszuführen, die über einzelne Kommandos eingegeben werden müssen. Jedes Kommando besteht

aus einem Wort gefolgt von einem oder mehreren Parametern. Bei den Kommandos müssen Sie nicht auf die Groß- oder Kleinschreibung achten, bei den Parametern hingegen könnte es wichtig sein.

```

Terminal <2>
Datei  Einstellungen  Hilfe
+uitrc.xterm          H    10687  Fri Nov 17 19:22:03 1995
+xcoralrc            H     7913  Thu Nov 30 17:48:22 1995
+xfm                 DH      0    Thu Oct 21 22:59:26 1999
+xinitrc             AH    1870   Thu Dec 10 13:42:34 1998
+xserverrc.secure    H     795   Mon Sep  7 16:48:09 1998
+xsession            AH    1714   Thu Dec 10 13:42:45 1998
+xtalkrc             H     119   Tue Oct 28 13:39:42 1997
+zsh                  H      10    Fri Nov 17 19:22:03 1995
suse01.gif           13332  Sat Oct 23 18:19:23 1999
suse02.gif           12190  Sat Oct 23 18:20:59 1999
suse03.gif           13793  Sat Oct 23 18:23:44 1999
suse04.gif           15811  Sat Oct 23 18:27:51 1999
suse05.gif           13790  Sat Oct 23 18:29:34 1999
suse06.gif           11019  Sat Oct 23 18:30:49 1999
suse07.gif           15049  Sat Oct 23 18:32:35 1999
suse08.gif           14811  Sat Oct 23 18:34:46 1999
suse09.gif            8366  Sat Oct 23 18:36:19 1999
suse10.gif           18446  Sat Oct 23 18:37:55 1999
suse11.gif           14627  Sat Oct 23 18:38:47 1999
suse12.gif           15656  Sat Oct 23 18:40:15 1999
suse13.gif           11266  Sat Oct 23 18:41:49 1999

64143 blocks of size 8192, 8065 blocks available
smb: \>

```

Abbildung 3.1: Die Ausgabe des Prompts in einem Fenster

Insgesamt sind folgende Kommandos verfügbar, wobei die Parameter jeweils durch ein Leerzeichen getrennt anzugeben sind. Außerdem sind die Parameter in den eckigen Klammern optional, die in den Kleiner- und Größerzeichen müssen hingegen spezifiziert werden.

Kommando	Beschreibung
? [Kommando]	Bei Angabe des Kommandos wird eine kurze Information dazu ausgegeben. Wird kein Kommando spezifiziert, erfolgt die Ausgabe einer Liste mit den verfügbaren Kommandos.
! [Shell-Kommando]	Wird das Kommando angegeben, erfolgt der Start einer Shell, in der dieses dann lokal ausgeführt wird. Fehlt die Angabe des Shell-Kommandos, wird eine lokale Shell gestartet.

(Fortsetzung nächste Seite)

Kommando	Beschreibung
blocksize <Blockgröße>	Setzt die Blockgröße für eine tar-Operation, wobei der Wert <i>Blockgröße</i> größer als 0 sein muss. Dadurch werden die Daten in (<i>Blockgröße</i> *TBLOCK) Bytes geschrieben, wobei TBLOCK normalerweise den Wert 512 besitzt.
cd [Verzeichnis]	Bei Angabe des Verzeichnisses wird auf dem Server in dieses gewechselt, sofern es existiert und der Client die Rechte zum Zugriff darauf besitzt. Andernfalls wird eine entsprechende Fehlermeldung ausgegeben. Erfolgt keine Angabe des Verzeichnisses, wird das aktuelle Arbeitsverzeichnis ausgegeben.
del <Löschmaske>	Alle Dateien, die der Löschmaske entsprechen, sollen auf dem Server gelöscht werden, wobei der Server vom Client dazu aufgefordert wird. Der Löschvorgang findet jeweils im aktuellen Arbeitsverzeichnis statt.
dir <Dateimaske>	Es wird eine Liste der Dateien auf dem Server bestimmt, die zu der angegebenen Dateimaske passen. Diese Liste wird anschließend angezeigt. Sie enthält neben dem Namen auch die Größe und das Datum sowie die Zeit, zu der die Datei oder das Verzeichnis erzeugt wurden. Ein Verzeichnis wird übrigens nur durch den Buchstaben <i>D</i> angezeigt.
exit	Die Verbindung zum Server wird abgebrochen und das Programm beendet.
get <Dateiname> [LokalerName]	Die Datei mit dem Namen <i>Dateiname</i> wird vom Server auf den Rechner kopiert, auf dem der Client läuft. Gegebenenfalls wird die Datei auf dem lokalen Rechner noch in <i>LokalerName</i> umbenannt. Alle Kopieroperationen werden in binärer Form, also Bit für Bit, ausgeführt.
help [Kommando]	Bei Angabe des Kommandos wird eine kurze Information hierzu ausgegeben. Wird kein

(Fortsetzung nächste Seite)

Kommando	Beschreibung
	Kommando spezifiziert, erfolgt die Ausgabe einer Liste mit den verfügbaren Kommandos.
lcd [Verzeichnis]	Bei Angabe des Verzeichnisses wird auf dem lokalen Rechner (auf dem der Client läuft) dorthin gewechselt, sofern es existiert und der Client die Rechte zum Zugriff darauf besitzt. Ist das nicht der Fall, wird eine entsprechende Fehlermeldung ausgegeben. Erfolgt keine Angabe des Verzeichnisses, wird das aktuelle Arbeitsverzeichnis ausgegeben.
lowercase	Die Dateinamen bei den Kommandos <i>get</i> und <i>mget</i> werden in Kleinbuchstaben umgewandelt. Diese Option kann sinnvoll sein, um Dateien von einem Unix-System zu kopieren.
ls <Dateimaske>	Es wird eine Liste der Dateien auf dem Server bestimmt, die zu der angegebenen Dateimaske passen. Diese Liste wird anschließend angezeigt. Sie enthält neben dem Namen auch die Größe und das Datum sowie die Zeit, zu der die Datei oder das Verzeichnis erzeugt wurden. Ein Verzeichnis wird übrigens nur durch den Buchstaben <i>D</i> angezeigt.
mask <Maske>	Hiermit kann ein Benutzer eine Maske setzen, die bei rekursiven Operationen der Kommandos <i>mget</i> und <i>mput</i> Verwendung findet. Mit Hilfe dieser Maske können bestimmte Dateien in ausgewählten Unterverzeichnissen selektiert werden. Wird diese Maske nicht gesetzt, werden standardmäßig alle Dateien in den Unterverzeichnissen ausgewählt.
md <Verzeichnisname>	Ein neues Verzeichnis mit dem spezifizierten Namen wird auf dem Namen angelegt. Dabei werden hierzu die Rechte des Benutzers verwendet.

(Fortsetzung nächste Seite)

Kommando	Beschreibung
mget <Maske>	<p>Alle Dateien, die zu der angegebenen Maske passen, werden vom Server auf den lokalen Rechner kopiert. Dabei müssen Sie beachten, dass die Masken im rekursiven und nichtrekursiven Betrieb unterschiedlich interpretiert werden.</p> <p>Alle Kopieroperationen werden in binärer Form, also Bit für Bit, ausgeführt.</p>
mkdir <Verzeichnisname>	<p>Ein neues Verzeichnis mit dem spezifizierten Namen wird auf dem Namen angelegt. Dabei werden hierzu die Rechte des Benutzers verwendet.</p>
mput <Maske>	<p>Alle Dateien, die der Maske entsprechen, werden vom aktuellen Arbeitsverzeichnis des lokalen Rechners in das Arbeitsverzeichnis des Servers kopiert. Dabei müssen Sie beachten, dass die Masken im rekursiven und nichtrekursiven Betrieb unterschiedlich interpretiert werden.</p> <p>Alle Kopieroperationen werden in binärer Form, also Bit für Bit, ausgeführt.</p>
print <Dateiname>	<p>Die angegebene Datei auf dem lokalen Rechner wird über einen Druckdienst auf dem Server ausgegeben.</p>
printmode <Grafik> printmode <Text>	<p>Der Druckmodus wird gesetzt, sodass entweder binäre Daten bzw. Grafiken oder Text verarbeitet werden können. Alle weiteren Druckoperationen verwenden den aktuell gesetzten Druckmodus.</p>
prompt	<p>Während der Ausführung der Kommandos <i>mget</i> und <i>mput</i> wird gefragt, ob die entsprechende Kopieroperation ausgeführt werden soll. Wollen Sie die Nachfrage permanent unterdrücken, so können Sie diesen Befehl einsetzen. Dabei wird dann jeweils zwischen den beiden Modi umgeschaltet, sodass Sie sich</p>

(Fortsetzung nächste Seite)

Kommando	Beschreibung
	durch doppeltes Ausführen des Befehls wieder im Fragemodus befinden.
put <LokalerName> [Dateiname]	Die Datei mit dem Namen <i>LokalerName</i> wird vom lokalen Rechner auf den Server kopiert. Gegebenenfalls wird sie dort noch in <i>Dateiname</i> umbenannt. Alle Kopieroperationen werden in binärer Form, also Bit für Bit, ausgeführt.
pwd	Auch die Eingabe dieses Kommandos erzeugt eine Ausgabe, allerdings sehen Sie nur, auf welchem Rechner Sie sich eingeloggt haben ...
queue	Gibt Informationen über die Drucker-Queue aus. Diese beinhalten die Job-ID, den Namen, die Größe und den aktuellen Status.
quit	Die Verbindung zum Server wird abgebrochen und das Programm beendet.
rd <Verzeichnis>	Das angegebene Verzeichnis auf dem Server wird gelöscht. Dabei wird vorausgesetzt, dass die entsprechenden Rechte zum Löschen vorhanden sind.
recurse	Die Verzeichnis-Rekursion für die Kommandos <i>mget</i> und <i>mput</i> wird mit diesem Kommando umgeschaltet. Ist sie angeschaltet, werden alle Unterverzeichnisse des Quellenverzeichnisses bearbeitet und die Dateien entsprechend den angegebenen Masken kopiert. Im anderen Fall werden nur die Dateien des aktuellen Arbeitsverzeichnisses, die der angegebenen Maske entsprechen, mit den Kommandos kopiert. Die mit dem Kommando <i>mask</i> gesetzte Maske wird in diesem Fall ignoriert.
rm <Maske>	Alle Dateien, die der Maske entsprechen, werden aus dem aktuellen Verzeichnis des Servers gelöscht.

(Fortsetzung nächste Seite)

Kommando	Beschreibung
rmdir <Verzeichnis>	Das angegebene Verzeichnis auf dem Server wird gelöscht. Dabei wird vorausgesetzt, dass die entsprechenden Rechte zum Löschen vorhanden sind.
setmode <Dateiname> <+rsha> setmode <Dateiname> <-rsha>	Setzt (+) bzw. löscht (-) die Attribute bzw. die Zugriffsrechte der Datei mit dem Namen <i>Dateiname</i> . Dabei bezeichnet der Buchstabe <i>r</i> das Attribut 'nur lesbar', bei <i>s</i> handelt es sich um das Systembit, bei <i>h</i> um das Attribut, dass die Datei nicht sichtbar ist (unter Linux jede Datei, die mit einem Punkt beginnt) und der Buchstabe <i>a</i> kennzeichnet das Archiv-Bit.
tar <c>[IXbgNa] tar <x>[IXbgNa]	Führt eine tar-Operation aus, wobei im ersten Fall ein Archiv erzeugt (<i>create</i>) und im zweiten ein Archiv entpackt (<i>extract</i>) wird. Dabei darf zwischen dem ersten und den weiteren optionalen Parametern kein Leerzeichen enthalten sein. Über den Parameter <i>I</i> können bestimmte Dateien und Verzeichnisse zusätzlich eingebunden werden, wobei der Parameter <i>X</i> genau das Gegenteil bewirkt, nämlich den Ausschluss bestimmter Dateien und Verzeichnisse. Mit dem Parameter <i>b</i> kann die Blockgröße gesetzt werden und durch Angabe von <i>g</i> werden nur die Dateien bearbeitet, bei denen das Archiv-Bit gesetzt ist. Sollen nur Dateien archiviert werden, die neuer als ein bestimmtes Datum sind, müssen Sie dazu den Parameter <i>N</i> angeben. Schließlich kann auch noch beim Archivieren das Archiv-Bit gesetzt werden, wozu die Angabe des Parameters <i>a</i> erforderlich ist.
tarmode tarmode <full> tarmode <inc> tarmode <noreset> tarmode <reset>	Werden keine Parameter angegeben, gibt das Programm die aktuellen Modi aus, wobei auch mehrere Modi gesetzt sein können. Verändert das Verhalten von tar im Umgang mit Archiv-Bits. Im Modus <i>full</i> werden alle Dateien archiviert, unabhängig vom Status

(Fortsetzung nächste Seite)

Kommando	Beschreibung
	<p>Ihres Archiv-Bits, wobei dieser Modus der Standardmodus ist. Im Modus <i>inc</i> hingegen werden nur die Dateien archiviert, die ein gesetztes Archiv-Bit besitzen. Im Modus <i>no-reset</i> wird das Archiv-Bit bei allen gesicherten Dateien nicht zurückgesetzt. Im Modus <i>reset</i> wird schließlich das Archiv-Bit bei allen gesicherten Dateien zurückgesetzt.</p> <p>Insgesamt können auch mehrere Modi gesetzt werden, wozu allerdings der Befehl mehrfach verwendet werden muss.</p>

Geben Sie übrigens einen Befehl ein, für den weitere Parameter erforderlich sind, und haben Sie diese nicht spezifiziert, wird Ihnen eine kurze Hilfe mit der Befehlssyntax angezeigt. Dasselbe gilt übrigens auch, wenn Sie einmal Parameter falsch angeben. Somit müssen Sie sich also nicht unbedingt alle Parameter merken, sondern nur die Befehle ...

Abschließend möchte ich Ihnen noch kurz ein Beispiel für den Einsatz des Programms zeigen. Allerdings funktioniert es nur dann korrekt, wenn bereits eine entsprechende Konfigurationsdatei *smb.conf* existiert. Unter der Annahme, dass der (lokale) Rechner *DeepThought* heißt und sich auf ihm ein Benutzerkonto *ws* befindet, könnte das Kommando folgendermaßen aussehen:

```
smbclient \\\DeepThought\ws
```

Dabei sind die doppelten Backslashes erforderlich, weil durch ein einzelnes Zeichen ein Befehl eingeleitet würde. Bei der Interpretation des Kommandos wird somit aus dem doppelten Backslash ein einfacher, sodass das Kommando *smbclient \DeepThought\ws* ausgeführt wird.

Danach werden Sie nach dem entsprechenden Passwort gefragt. Haben Sie dieses eingegeben und ist es korrekt, landen Sie an einem Prompt (*smb: \>*), an dem Sie jetzt die Befehle eingeben können, beispielsweise um Dateien zu kopieren.

Alternativ könnten Sie aber auch das nachfolgende Kommando angeben, um auf ein Verzeichnis zuzugreifen:

```
smbclient //DeepThought/home
```

Dabei wird wieder vorausgesetzt, dass der (lokale) Rechner *DeepThought* heißt und das Verzeichnis *home* in der Datei *smb.conf* entsprechend gesetzt worden ist.

Zusätzlich müssen Sie noch das entsprechende Passwort angeben, wonach wieder ein Prompt angezeigt wird.

Wollen Sie sich die Dienste anschauen, die auf dem Server *DeepThought* verfügbar sind, sollten Sie das nachfolgende Kommando angeben:

```
smbclient -L DeepThought
```

Nach dem Drücken der Eingabetaste für das Passwort sehen Sie eine Liste der Verzeichnisse, auf die Sie analog dem vorigen Beispiel zugreifen könnten.

3.2.3 smbmount

Das Programm *smbmount* ist eine spezielle Version von *smbclient*, die nur dazu verwendet wird, SMB-Verzeichnisse in das lokale Dateisystem einzubinden. Sie können dann wie gewohnt auf die Dateien zugreifen, als ob sie sich auf der lokalen Platte befinden würden. Das Programm versteht ansonsten fast alle Optionen, die auch *smbclient* kennt.

Wollen Sie also das Verzeichnis *home* eines Rechners mit dem Namen *DeepThought* einbinden, müssen Sie das dieses Kommando angeben:

```
smbmount //DeepThought/home /mnt
```

Dabei wird vorausgesetzt, dass das Verzeichnis *home* in der Datei *smb.conf* entsprechend gesetzt worden ist. Zusätzlich müssen Sie noch das entsprechende Passwort angeben und Sie landen wieder am normalen Linux-Prompt. Sie können im obigen Beispiel dann in das lokale Verzeichnis */mnt* wechseln, um auf die Dateien zuzugreifen.

3.2.4 smbpasswd

Hiermit können Sie das SMB-Passwort eines Benutzers verändern. Dabei hat das Programm *smbpasswd* mehrere unterschiedliche Funktionen, abhängig davon, ob es von einem normalen oder einem root-Benutzer gestartet wurde. Im ersten Fall (Start des Programms durch einen gewöhnlichen Benutzer) kann der Benutzer nur seine eigenen Passwörter verändern; dies ist aber nur auf den Rechnern möglich, zu denen er eine Zugangsberechtigung besitzt und die SMB-Passwörter speichern.

Wird das Programm ohne die Angabe von Parametern gestartet, versucht es, das aktuelle SMB-Passwort des Benutzers auf dem lokalen Rechner zu ändern (dies entspricht in etwa der Arbeitsweise des Linux-Programms *passwd*). Allerdings

arbeitet das Programm in einem Client-Server-Modus und kommuniziert mit einem lokal laufenden Server *smbd*, sodass hier ein Unterschied zu *passwd* besteht. Aus diesem Grund muss der Dämon *smbd* auf dem Rechner laufen, auf dem das Programm ausgeführt wird, da sonst ein Fehler auftritt. Auf einem Linux-Rechner werden die verschlüsselten SMB-Passwörter in einer Datei mit dem Namen *smbpasswd* gespeichert, die Sie zumindest bei der SuSE-Distribution im Verzeichnis */etc* finden.

Startet ein normaler Benutzer das Programm ohne Parameter, wird er zuerst nach seinem alten Passwort gefragt. Danach darf er das neue Passwort zweimal angeben, um Tippfehler auszuschließen. Bei der Eingabe werden übrigens keine Buchstaben im Eingabefeld angezeigt, sodass niemand mitlesen kann. Wurde zuvor noch kein Passwort gesetzt, d.h. existiert im entsprechenden Feld in der Passwort-Datei der Eintrag *NO PASSWORD*, muss bei der Frage nach dem alten Passwort nur die Eingabetaste gedrückt werden.

Das Programm kann von einem normalen Benutzer ansonsten auch verwendet werden, um die SMB-Passwörter auf entfernten Rechner zu verändern, beispielsweise auf einem Windows NT-Rechner, der als primärer Domain Controller dient.

Wird das Programm hingegen von einem root-Benutzer gestartet, können neue Benutzer hinzugefügt und auch gelöscht werden. Zusätzlich ist es möglich, die Attribute des entsprechenden Benutzers zu ändern. In diesem Fall wird auf die lokale Datei *smbpasswd* direkt zugegriffen, sodass auch Änderungen vorgenommen werden können, wenn der Dämon *smbd* nicht läuft. Außerdem können Sie die folgenden Parameter beim Aufruf mit angeben, wobei zum Schluss immer ein Benutzername stehen muss:

Parameter	Beschreibung
-a	Der angegebene Benutzername soll in die lokale Datei <i>smbpasswd</i> aufgenommen werden. Danach müssen das alte und das neue Passwort angegeben werden. Existiert der Eintrag in der Datei noch nicht, muss für das alte Passwort nur die Eingabetaste gedrückt werden. Im anderen Fall wird die Option ignoriert und der Befehl wirkt wie eine normale Veränderung des Passwortes. Dabei ist zu beachten, dass der Benutzer, der hinzugefügt werden soll, in der Passwort-Datei des Systems (normalerweise <i>/etc/passwd</i>) bereits existieren muss, da sonst ein Fehler auftritt.

(Fortsetzung nächste Seite)

Parameter	Beschreibung
-d	<p>Der angegebene Benutzername wird in der lokalen Datei <i>smbpasswd</i> deaktiviert. Dies erfolgt durch Schreiben eines D-Flags in das entsprechende Feld der Zeile. Danach kann der angegebene Benutzer unter seinem Namen keine Verbindung mehr mit dem SMB-Server aufbauen.</p> <p>Liegt die Datei noch in einem alten Format vor, existiert kein Platz für das Schreiben der Information, sodass der Benutzer deaktiviert wird durch das Überschreiben des Passwort-Feldes mit dem Buchstaben 'x' (oder 'X').</p>
-e	<p>Der angegebene Benutzername wird aktiviert in der lokalen Passwortdatei <i>smbpasswd</i>, wenn dieser vorher deaktiviert war. Wurde er vorher nicht deaktiviert, hat die Option keine Wirkung. Danach kann der Benutzer eine Verbindung unter seinem Namen mit dem SMB-Server aufbauen.</p> <p>Liegen die Einträge der Datei im alten Format vor, müssen Sie ein Passwort für den Benutzer angeben, bei einer Datei im neuen Format wird nur das D-Flag entfernt. Die Angabe eines Passwortes ist in diesem Fall nicht notwendig.</p>
-j Domain	<p>Mit Hilfe dieser Option wird ein Samba-Server in einer Windows NT-Domain hinzugefügt. Dabei kann der Samba-Server auch Benutzerkonten wie ein NT-Server authentifizieren. Dazu muss der Administrator der Domain das Programm „<i>Server Manager for Domains</i>“ starten, um den primären NetBIOS-Namen des Samba-Servers der Domain hinzuzufügen.</p> <p>Danach muss das Programm <i>smbpasswd</i> mit diesem Parameter aufgerufen werden, um eine Verbindung herstellen zu können. Das Programm wird jetzt beim primären Domain Controller nach dieser Domain nachschauen, um das Rechnerkonto-Passwort für den Aufbau einer sicheren Domain-Kommunikation zu ändern. Dieses Passwort wird dann in einer Datei gespeichert, die nur vom root-Benutzer gelesen werden kann und die den Namen <i>Domain.Rechner.mac</i> hat. Dabei bezeichnen der String <i>Domain</i> die Domain, in der die Verbindung aufgebaut wird, und <i>Rechner</i> den NetBIOS-Namen der Maschine, auf dem der Server läuft.</p> <p>Nachdem diese Operation ausgeführt worden ist, wird die Konfigurationsdatei <i>smb.conf</i> aktualisiert und der Parameter <i>security = domain</i> aufgenommen. Danach werden alle weiteren Login-Versu-</p>

(Fortsetzung nächste Seite)

Parameter	Beschreibung
	<p>che auf dem Samba-Server gegenüber dem primären Windows NT Domain Controller entsprechend authentifiziert.</p> <p>Allerdings müssen Sie dabei beachten, dass alle Benutzer, die auf den Samba Server zugreifen wollen, ein gültiges Linux-Benutzerkonto haben müssen.</p>
-m	<p>Das Benutzerkonto wird in ein Rechner-Konto umgewandelt. Momentan wird es genutzt, wenn Samba als primärer Domain Controller eingesetzt wird.</p>
-n	<p>Die Passwortangabe für den angegebenen Benutzer wird deaktiviert, d.h. das Passwort ist „leer“. Es erfolgt durch das Schreiben des Strings <i>NO PASSWORD</i> als ersten Teil des Passwortfeldes in der lokalen Datei <i>smbpasswd</i>.</p> <p>Damit ein Benutzer in diesem Fall eine Verbindung mit einem SMB-Server herstellen kann, muss ein Administrator in der Datei <i>smb.conf</i> in der Sektion <i>[global]</i> folgende Zeile eintragen:</p> <pre style="text-align: center;">null passwords = True</pre>
-r Rechnername	<p>Mit Hilfe dieser Option kann festgelegt werden, auf welchem Rechner die Passwort-Datei geändert werden soll. Ohne Angabe wird normalerweise die lokale Datei verändert. Als <i>Rechnername</i> muss der NetBIOS-Name des Servers angegeben werden. Dieser Name wird in eine IP-Adresse umgewandelt unter Verwendung des Standard-Auflösungsverfahrens.</p> <p>Der Benutzername, dessen Passwort verändert werden soll, ist der des momentan eingeloggten Unix-Benutzers.</p> <p>Wenn das Passwort einer Windows NT Domain verändert wird, muss der angegebene Rechner der primäre Domain Controller sein.</p> <p>Da die Rechner, auf denen Windows 95/98 läuft, normalerweise keine reale Passwort-Datenbank besitzen, ist es nicht möglich, dort die Passwörter zu ändern, sodass es nicht sinnvoll ist, einen solchen Rechner anzugeben.</p>
-R Reihenfolge	<p>Mit Hilfe dieser Option kann die Reihenfolge verändert werden, in der beispielsweise in Dateien gesucht wird, um einen NetBIOS-Namen in eine IP-Adresse umzusetzen.</p>

(Fortsetzung nächste Seite)

Parameter	Beschreibung	
	Dabei sind folgende Optionen möglich:	
	Option	Beschreibung
	bcast	Es wird eine Rundsendung auf allen bekannten lokalen Netzwerkschnittstellen, die in der Datei <i>smb.conf</i> spezifiziert worden sind, ausgeführt. Allerdings ist diese Möglichkeit wenig verlässlich, da Sie stark abhängig ist von dem Ziel-Rechner, der mit dem Netzwerk verbunden ist.
	hosts	Es wird ein Standard-Hostnamen zu IP-Adressen Auflösung ausgeführt unter Verwendung der Datei <i>/etc/hosts</i> oder den NIS bzw. DNS-Dateien.
	lmhosts	Die IP-Adresse wird in der Datei <i>lmhosts</i> gesucht.
	wins	Die IP-Adresse wird über den Parameter <i>wins server</i> in der Datei <i>smb.conf</i> bestimmt. Wurde kein entsprechender Server angegeben, wird diese Methode ignoriert.
	<p>Wird diese Option nicht angegeben, wird die Methode verwendet, die beim Parameter <i>name resolve order</i> in der Konfigurationsdatei <i>smb.conf</i> spezifiziert worden ist.</p> <p>Standardmäßig wird übrigens in der Reihenfolge <i>lmhosts</i>, <i>hosts</i>, <i>wins</i> und <i>bcast</i> versucht, den NetBIOS-Namen in eine IP-Adresse zu verwandeln.</p>	
-U Benutzername	Dieser Parameter sollte nur in Zusammenhang mit der Option <i>-r</i> verwendet werden. Wird ein Passwort auf einem entfernten Rechner verändert, kann ein Benutzer den entsprechenden Namen auf dem Rechner angeben, auf dem das Passwort geändert wird. Damit können Benutzer, die sich auf unterschiedlichen Rechner unter verschiedenen Namen einloggen, auf diesen auch ihre eigenen Passwörter ändern.	

Der Name des Benutzers darf stets nur dann angegeben werden, wenn der root-Benutzer das Programm gestartet hat, da nur er die Rechte hat, Attribute direkt in der lokalen Passwortdatei *smbpasswd* zu ändern.

Da das Programm in einem Client-Server-Modus arbeitet und mit dem lokalen Dämon *smbd* kommuniziert, wenn es von einem normalen Benutzer gestartet wird, muss dieser Dämon natürlich vor dem Aufruf gestartet worden sein. Dabei besteht ein allgemeines Problem, wenn Einschränkungen für Rechner hinzugefügt werden sollen, die auf den Dämon *smbd* zugreifen, der auf dem lokalen Rechner läuft. Dazu müssen die Einträge *allow hosts* oder *deny hosts* der Konfigurationsdatei *smb.conf* angepasst werden. Dabei darf keineswegs versäumt werden, den lokalen Rechner *localhost* auch passend einzutragen, damit dieser auch weiterhin auf den Server *smbd* zugreifen kann.

Insgesamt ist der Einsatz des Kommandos *smbpasswd* nur sinnvoll, wenn verschlüsselte Paßwörter eingesetzt werden sollen.

3.2.5 smbprint

Mit Hilfe des Skriptes *smbprint* kann ein Unix-Rechner auf einem SMB-Server drucken. Zum eigentlichen Drucken wird das Programm *smbclient* aufgerufen. Allerdings ist das Skript nur als Beispiel gedacht, sodass Sie es gegebenenfalls anpassen müssen. Sie finden es im Verzeichnis */usr/doc/packages/samba/examples/* (zumindest bei SuSE). Es kann normalerweise nicht von einer Shell aus aufgerufen werden, es sei denn, das entsprechende Verzeichnis wurde in den Pfad mit aufgenommen.

3.2.6 smbrun

Das Programm *smbrun* bildet die Schnittstelle zwischen dem Dämon *smbd* und einem externen Programm. Es dient dazu, Shell-Kommandos für den Server *smbd* auszuführen. Damit können auf bestimmten Betriebssystemen auch externe Programme ausgeführt werden, wenn keine Administrator-Berechtigung vorhanden ist.

Kann das Programm vom Server nicht gefunden oder nicht ausgeführt werden, schreibt er entsprechende Meldungen in seine Log-Datei. Aus diesem Grund sollten Sie sich die Datei zumindest nach einer Erstinstallation genauer ansehen, um gegebenenfalls Fehler zu finden.

3.2.7 smbstatus

Mit Hilfe des Programms *smbstatus* können Sie sich anzeigen lassen, welche Verbindungen mit dem Samba-Server gerade bestehen. Dabei können Sie die Ausgabe mit folgenden Parametern einschränken bzw. erweitern.

Parameter	Beschreibung
-b	Zeigt eine kurze Liste der bestehenden Verbindungen an.
-d	Gibt eine ausführliche Liste aus, wobei auch die verwendete Konfigurationsdatei mit ausgegeben wird.
-L	Zeigt nur Sperren an. Eine Sperre wird beispielsweise eingerichtet, wenn auf eine Datei zugegriffen wird.
-p	Gibt eine Liste der IDs aus, die die smbd-Prozesse verwenden.
-S	Die Liste enthält nur die Verzeichnisse, auf die momentan gerade zugegriffen wird.

Geben Sie nur den Befehl ein, sehen Sie Angaben über eine bestehende Verbindung und gegebenenfalls, auf welche Dateien der Zugriff gesperrt worden ist. Zusätzlich wird Ihnen angezeigt, wieviel Speicher verwendet worden ist (Abbildung 3.2).

```

Terminal <6>
Datei  Einstellungen  Hilfe
bash-2.03# smbstatus
Samba version 2.0.5a
Service      uid      gid      pid      machine
-----
ws           root    root     609     deepthought (192.168.0.1) Thu Nov  4 22:5
3:04 1999
No locked files
Share mode memory usage (bytes):
 1048464(99%) free + 56(0%) used + 56(0%) overhead = 1048576(100%) total
bash-2.03#

```

Abbildung 3.2: Die Ausgabe der Statusinformationen in einem Fenster

Besteht keine Verbindung, werden normalerweise nur eine leere Liste und der Speicherbedarf ausgegeben, wenn diese zuvor schon aufgebaut war. Bestand hingegen vorher noch keine Verbindung, sehen Sie nur die Fehlermeldung, dass die Datei `.../STATUS.LCK` nicht geöffnet werden kann. Das ist aber nicht unbedingt ein Fehler, sondern zeigt nur, dass keine Verbindung besteht.

Dieses Programm wird übrigens auch vom KDE-Kontrollzentrum verwendet, um den Samba-Status im Menü *Information* im Fenster anzuzeigen.

3.2.8 **smbtar**

Mit Hilfe des Skriptes *smbtar* können SMB-Verzeichnisse direkt auf Linux-Bandlaufwerke gesichert werden. Dabei müssen Sie natürlich angeben, welche Dateien gesichert werden sollen. Außerdem ist es notwendig, dem Skript zu sagen, welcher Server verwendet werden soll, wozu der Parameter *-s* gefolgt vom jeweiligen Namen dient.

Neben dem Server ist auch die Angabe des Verzeichnisses möglich, das gesichert werden soll. Hierzu dient der Parameter *-x*, dem noch der Name getrennt von einem Leerzeichen folgen muss. Geben Sie diesen nicht an, wird standardmäßig das Verzeichnis *backup* verwendet. Existiert es nicht, sehen Sie eine entsprechende Fehlermeldung.

Außerdem können Sie mit dem Parameter *-t* festlegen, welches Bandlaufwerk verwendet werden soll. Dabei müssen Sie getrennt von einem Leerzeichen die Bezeichnung des Laufwerkes angeben. Standardmäßig wird ansonsten das Laufwerk verwendet, das über die Variable *TAPE* definiert wird. Ist sowohl diese Variable als auch der Parameter nicht angegeben worden, wird im lokalen Verzeichnis auf dem Linux-Rechner eine Datei mit dem Namen *tar.out* angelegt.

Sollen also alle Dateien des Verzeichnisses *home* auf dem Rechner mit dem Namen *Deepthought* gesichert werden, können Sie dieses Kommando angeben:

```
smbtar -s Deepthought -x home *
```

Dabei wird das Bandlaufwerk verwendet, das durch die Variable *TAPE* spezifiziert worden ist. Existiert diese Variable nicht, wird die Datei *tar.out* im aktuellen, lokalen Verzeichnis des Linux-Rechners geschrieben.

3.2.9 **smbumount**

Das Programm *smbumount* entfernt eine mit *smbmount* erzeugte Einbindung aus dem Dateisystem. Dabei müssen Sie als Parameter den Namen des entsprechenden Verzeichnisses angeben. Für das Entfernen ist kein *Password* erforderlich.

Wollen Sie also beispielsweise das Verzeichnis */mnt* wieder freigeben, in das ein entferntes Verzeichnis mittels *smbmount* eingebunden worden ist, müssten Sie folgenden Befehl eingeben:

```
smbumount /mnt
```

Dabei kann nur jener Benutzer dieses Verzeichnis wieder freigeben, der auch die Verbindung aufgebaut hat.

3.2.10 swat

Das Programm *swat* erlaubt die Konfiguration der Datei *smb.conf* über einen Browser, wie z.B. den Netscape-Navigator. Zusätzlich können über die entsprechende Seite im Browser auch die Hilfedateien des Samba-Paketes aufgerufen werden. Auch zu den einzelnen Optionen, die über *swat* (Samba Web Administration Tool) eingestellt werden können, lassen sich durch Anklicken Hilfeseiten aufrufen, sodass sich ein Administrator genau informieren kann, was passiert, wenn eine Option geändert wird.

Die Möglichkeit zum Einsatz von *swat* wird über den Dämon *inetd* freigegeben; dessen Konfigurationsdatei *inetd.conf* muss angepasst werden. Zusätzlich ist auch die Datei *services* zu überprüfen und gegebenenfalls zu ändern. Beide Dateien finden Sie im Verzeichnis */etc* (zumindest bei der SuSE-Distribution).

Dabei muss in der Datei */etc/inetd.conf* folgende Zeile eingetragen werden:

```
swat stream tcp nowait.400 root /usr/sbin/swat swat
```

In der Regel finden Sie sie vorbereitet in der Datei, nur ist sie auskommentiert, sodass Sie lediglich ein Doppelkreuz löschen müssen und gegebenenfalls ein weiteres Zeichen, damit das Wort *swat* auch ganz links beginnt. Dabei finden Sie das Programm selbst, bei der SuSE-Distribution, im Verzeichnis */usr/sbin*. Ist die Zeile in der Datei hingegen nicht enthalten, fügen Sie sie einfach hinzu. Abschließend speichern Sie sie ab.

Daneben ist auch noch die Datei */etc/services* zu ändern. Hier muss eine Zeile zu finden sein, die etwa so aussieht:

```
swat 901/tcp
```

Ist sie enthalten und nicht mit einem Doppelkreuz auskommentiert, müssen Sie nichts machen. Ansonsten entfernen Sie wieder das Doppelkreuz und gegebenenfalls noch weitere Zeichen oder Sie fügen die Zeile hinzu. Haben Sie an der Datei etwas geändert, müssen Sie sie noch speichern.

Verwenden Sie die NIS-/YP-Dienste, müssen vorrangig die NIS-Service-Dateien verändert werden.

Die Wahl der Portnummer 901 ist eigentlich nicht wichtig, nur sollte sie kleiner als 1024 sein und nicht von einem anderen Dienst benutzt werden. Bei einer Nummer, die größer als 1024 ist, präsentiert sich unter Umständen ein (merkwürdiges) Sicherheitsloch, abhängig von der Implementation des Dämons *inetd*. Aus

diesem Grund sollten Sie es bei der 901 belassen, so haben Sie auch die Gewissheit, dass kein anderer Dienst Ihnen dazwischenfunkt ...

Haben Sie die beiden Dateien verändert, muss ein HUP-Signal (hangup) an den Dämon gesendet werden. Dazu müssen Sie allerdings erst einmal herausbekommen, welche ID der entsprechende Prozeß des Dämons besitzt. Zu diesem Zweck geben Sie folgendes Kommando ein:

```
ps aux
```

In der angezeigten Liste sehen Sie die gestarteten Prozesse (letzte Spalte) und auch die zugehörigen IDs (zweite Spalte), sodass Sie jetzt nur noch eine Zeile finden müssen, die so ähnlich aussieht wie diese::

```
root 125 0.0 0.3 1440 228 ? S 16:58 0:00 /usr/sbin/inetd
```

Dabei kennzeichnet in diesem Beispiel die 125 die Prozeß-ID. Alle anderen Angaben in der Zeile sind übrigens in diesem Zusammenhang nicht wichtig, sie können erst einmal ignoriert werden.

Haben Sie die ID (PID) des inetd-Prozesses bestimmt, so geben Sie den folgenden Befehl ein:

```
kill -1 PID
```

Dabei bezeichnet die Buchstabenfolge *PID* natürlich den zuvor bestimmten Wert. Dadurch wird der Prozeß beendet und erneut gestartet, sodass die Änderungen wirksam werden.

Ist Ihnen das zu viel Arbeit, können Sie alternativ auch die „Holzhammer“-Methode wählen, indem Sie den Rechner herunterfahren und anschließend neu starten. Auch in diesem Fall sind nach dem Neustart die Änderungen wirksam ...

Nachdem Sie nun diverse Konfigurationsdateien geändert haben, können Sie das Werkzeug in einem Internet-Browser aufrufen. Dazu starten Sie ihn jetzt, sofern Sie ihn bereits installiert haben. Geben Sie in der Kommandozeile in einem Terminalfenster auf einer grafischen Oberfläche beispielsweise ein:

```
netscape &
```

Dadurch wird der Netscape Navigator als Hintergrundprozeß gestartet, sodass Sie auch weiter mit dem Terminal arbeiten können. Anschließend geben Sie in der Zeile, in der Sie sonst Ihre Internetseiten eintragen, ein:

```
localhost:901
```

Danach drücken Sie die Eingabetaste. Es erscheint eine Seite, in der Sie das *Password* und den Benutzernamen eingeben müssen (Abbildung 3.3).



Abbildung 3.3: Vor dem Gebrauch muss ein *Passwort* eingegeben werden

Im Feld *User ID* geben Sie den Namen des Benutzerkontos ein, wobei dieses normalerweise das *root*-Konto sein sollte, damit Sie vollen Zugriff auf die Datei haben. Darunter müssen Sie dann nur noch das zugehörige *Passwort* eingeben, woraufhin die Startseite von *swat* erscheint (Abbildung 3.4)

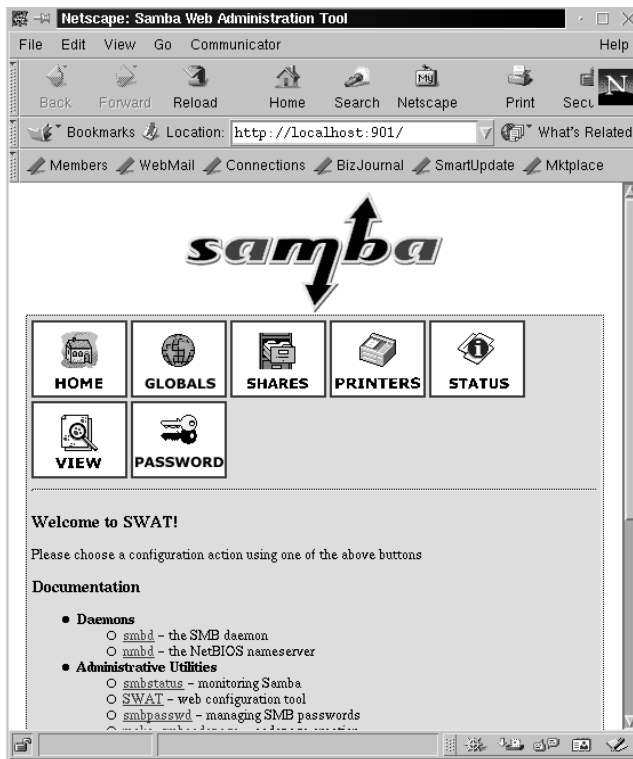


Abbildung 3.4: Die Startseite von *swat*

Dort sehen Sie im oberen Teil bunte Bildchen, die Sie durch Anklicken zu den zugehörigen Seiten bringen. Dabei haben Sie die Auswahl zwischen folgenden Möglichkeiten, wobei die Auswahlpunkte auf allen Seiten identisch sind.

Option	Beschreibung
HOME	<p>Hiermit gelangen Sie zur Startseite von <i>swat</i>. Dort finden Sie Verweise auf die Dokumentation zum Samba-Paket, sodass Sie gegebenenfalls schnell mal nachschauen können, wenn Sie etwas nicht wissen.</p>
GLOBALS	<p>Auf dieser Seite können Sie die Parameter einstellen, die in der Sektion <i>[global]</i> der Konfigurationsdatei <i>smb.conf</i> verwendet werden. Dabei werden Ihnen die einzelnen Parameter mit den zugehörigen Werten in Gruppen geordnet angezeigt. Dahinter finden Sie die entsprechenden Werte bzw. Strings.</p> <p>Normalerweise werden nach dem ersten Start nur die wesentlichen Einstellungen angezeigt. Das können Sie ändern, wenn Sie den Button <i>Advanced View</i> anklicken. Daraufhin erscheint zunächst eine Sicherheitswarnung – sofern Sie sie nicht abgeschaltet haben – und Sie müssen das Feld <i>Continue Submission</i> wählen, um weiterzukommen. Anschließend sehen Sie eine Liste von einstellbaren Optionen, die wesentlich länger ist als die zuvor. Wollen Sie übrigens wieder zurück, müssen Sie jetzt an gleicher Stelle das Feld <i>Basic View</i> anklicken.</p> <p>Neben den einzelnen Einstellmöglichkeiten sehen Sie in der jeweils ersten Spalte einer Parameterzeile das Wort <i>Help</i>. Klicken Sie es an, wird ein neues Fenster geöffnet, in dem dann die Beschreibung des Parameters angezeigt wird.</p> <p>Wollen Sie die Änderungen speichern, müssen Sie das Feld <i>Commit Changes</i> anklicken, das aber nur dann verfügbar ist, wenn Sie sich als <i>root</i> (oder als Benutzer mit entsprechenden Rechten) angemeldet haben ...</p>
SHARES	<p>Mit Hilfe dieser Seite können Sie die gemeinsam genutzten Verzeichnisse bearbeiten. Damit es funktioniert, müssen Sie als erstes im Feld rechts neben dem Button <i>Choose Share</i> einen entsprechenden Namen wählen. Hierzu klicken Sie das Feld an und halten die Maustaste gedrückt, wodurch eine Liste erscheint, in der Sie jetzt durch Bewegen der Maus einen Eintrag markieren können. Lassen Sie die Maustaste dann wieder los, erscheint der gewählte Name in diesem Feld.</p>

(Fortsetzung nächste Seite)

Option	Beschreibung
	<p>Anschließend müssen Sie den Button <i>Choose Share</i> anklicken, um sich die Optionen anzeigen zu lassen, wobei Sie gegebenenfalls wieder die Sicherheitswarnung sehen.</p> <p>Zusätzlich können Sie aber auch eine gemeinsam genutzte Ressource (share) löschen (<i>Delete Share</i>) oder auch erzeugen (<i>Create Share</i>). Wollen Sie die Änderungen speichern, müssen Sie das Feld <i>Commit Changes</i> auswählen. Allerdings sind die entsprechenden Optionen nur dann sichtbar, wenn Sie sich als <i>root</i> (oder als Benutzer mit entsprechenden Rechten) angemeldet haben ...</p> <p>Werden Ihnen die Optionen angezeigt, haben Sie wieder die Wahl zwischen einer Basis- (<i>Basic View</i>) und einer erweiterten Ansicht (<i>Advanced View</i>). Dabei ist die Basisansicht wieder diejenige, die zuerst angezeigt wird. Auch existiert wieder die Hilfeoption, sodass Sie sich schnell weitere Informationen zu einem Parameter anzeigen lassen können.</p>
PRINTERS	<p>Auch auf dieser Seite müssen Sie zuerst wieder einen Drucker auswählen, bevor Sie seine Optionen einstellen können. Dabei gilt es aber noch zu beachten, dass die Optionen der Drucker, deren Namen mit einem Stern beginnen, aus der Datei <i>printcap</i> geladen werden.</p> <p>Die weitere Vorgehensweise entspricht der zuvor beschriebenen, sodass Sie wieder zum Abschluss das Feld <i>Commit Changes</i> auswählen sollten, um die Änderungen zu speichern.</p>
STATUS	<p>Möchten Sie sich den Status der Verbindungen ausgeben lassen, die mit dem Samba-Server zustande gekommen sind, so wählen Sie diese Seite aus. Dort sehen Sie alle Einzelheiten über die Verbindung, so wie Sie es auch vom Programm <i>smbstatus</i> kennen.</p> <p>Zusätzlich haben Sie aber noch die Möglichkeit, dass auf dieser Seite die Anzeigen nach einer bestimmten Zeit automatisch aufgefrischt werden, wenn Sie das Feld <i>Auto Refresh</i> anklicken. Darunter können Sie übrigens vorher angeben, wie oft die Anzeige aktualisiert werden soll, wobei die Zeit in Sekunden gemessen wird. Tragen Sie dabei aber möglichst keinen zu kleinen Wert ein, da sonst der Bildschirm nur am Blinken ist ...</p> <p>Außerdem können Sie eine Verbindung gezielt abbrechen, indem Sie den Button in der Spalte <i>Kill</i> unter dem Oberbegriff <i>Active Connections</i> anklicken. Das ist besonders dann praktisch, wenn ein Client die</p>

(Fortsetzung nächste Seite)

Option	Beschreibung
	Verbindung einfach nicht trennen will, obwohl sie oberflächlich gar nicht mehr zu bestehen scheint ...
VIEW	<p>Haben Sie Ihre Optionen eingestellt und möchten Sie wissen, wie die zugehörige Konfigurationsdatei aussehen könnte (bzw. aussieht, wenn Sie die Änderungen gespeichert haben), sollten Sie diese Seite auswählen. Rufen Sie die Seite auf, landen Sie in der sogenannten normalen Ansicht. Dort sehen Sie nur die Optionen, die Sie eingestellt haben und die nicht den Standardwerten entsprechen. Möchten Sie sich hingegen die Werte aller Optionen anzeigen lassen, klicken Sie das Feld <i>Full View</i> an, woraufhin gegebenenfalls wieder die Sicherheitswarnung erscheint.</p> <p>Anschließend ist die Ausgabe wesentlich länger und Sie können sich dann auch die Werte der Optionen anschauen, bei denen Sie nichts verändert haben, wo also die Vorgabewerte gültig sind. Zurück in die normale Ansicht kommen Sie übrigens wieder durch Anklicken des Feldes <i>Normal View</i>.</p>
PASSWORD	Auf dieser Seite können Sie die Verwaltung von Paßwörtern vornehmen. Auch ist es hier möglich, Benutzer anzulegen oder zu löschen. Allerdings sollten Sie mit diesen Möglichkeiten etwas vorsichtig sein, wenn Sie sich von außen auf den Server eingeloggt haben, da die Paßwörter unverschlüsselt übertragen werden und gegebenenfalls also abgefangen werden könnten ...

Durch den Einsatz von *swat* wird die Konfigurationsdatei *smb.conf* neu geschrieben. Dabei werden die Einträge neu angeordnet und Kommentare entfernt (bis auf die von *swat* selbst erzeugten ...). Außerdem entfernt das Programm die Parameter *include* und *copy*, sodass Sie sehr vorsichtig sein sollten, wenn Sie die Datei von Hand angelegt haben. Wenn Sie dann auch noch die Optionen mit Kommentaren versehen haben, sollten Sie vor dem Einsatz von *swat* ein Backup der Datei anfertigen oder *swat* links liegen lassen ...

Der Aufruf des Programmes *swat* in einem Internet-Browser kann auch von außerhalb erfolgen, sofern die beiden Rechner eine IP-basierte Verbindung aufbauen können (was normalerweise immer der Fall sein sollte, wenn Sie das TCP/IP-Protokoll eingerichtet haben). Dazu starten Sie auf dem Rechner, auf dem nicht das Samba-Paket läuft, den Internet-Browser und geben in der entsprechenden Zeile den Internet-Namen des Samba-Server-Rechners gefolgt von einem Doppelpunkt und dem Wert 901 ein (sofern Sie ihn nicht geändert haben). Anschlie-

End drücken Sie die Eingabetaste, woraufhin Sie wieder aufgefordert werden, einen Benutzernamen und ein *Password* anzugeben. Ist alles korrekt, sollten Sie die swat-Startseite sehen (Abbildung 3.5).

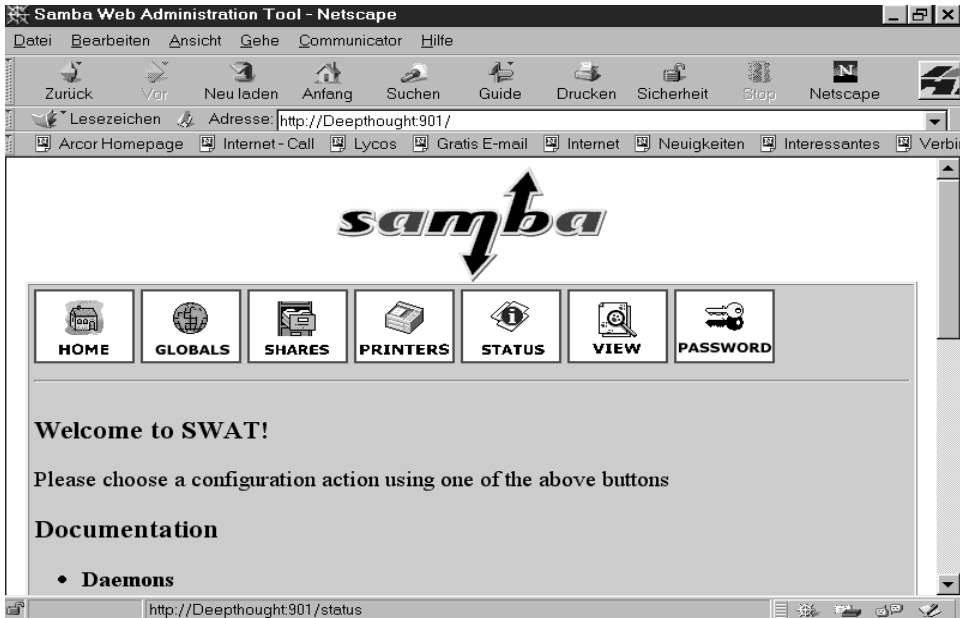


Abbildung 3.5: Die Startseite von swat auf einem Windows-Rechner

Beachten Sie dabei aber, dass unter Umständen die angegebenen Passwörter mitgelesen werden können, da sie unverschlüsselt übertragen werden. Gehen Sie mit dieser Möglichkeit also sehr vorsichtig um. Allerdings kann dies bei privaten Netzen sehr nützlich sein, da Sie jetzt Ihren Server einfach in die Ecke stellen können ...

Neben dieser (kurzen) Beschreibung der Möglichkeiten von *swat* wird sein Einsatz später noch genauer erläutert, wenn es darum geht, eigene Konfigurationsdateien zu erstellen. Dadurch wird die Administration sehr vereinfacht, weil Sie die Datei nicht immer erst in einem Editor bearbeiten müssen ...

3.2.11 testparm

Mit dem Programm *testparm* kann die Konfigurationsdatei auf interne Richtigkeit hin überprüft werden. Gibt dieses Programm bei der Untersuchung keine Feh-

ler aus, kann in der Regel auch der Server *smbd* die Konfigurationsdatei erfolgreich laden.

Allerdings ist das keine Garantie dafür, dass die in der Datei spezifizierten Services wie von Ihnen gewünscht funktionieren. Dies ist vergleichbar mit einem C++-Programm, bei dem der Compiler beim Übersetzen keine Fehler gemeldet hat. Das heißt aber noch lange nicht, dass das Programm auch wunschgemäß funktioniert ...

Als Parameter können optional der Name der Konfigurationsdatei und/oder der Name des Hostrechners und die IP-Adresse des Hostrechners angegeben werden. Dabei muss der Name der Konfigurationsdatei nur dann angegeben werden, wenn er nicht dem Standard entspricht oder die Datei sich in einem anderen Verzeichnis befindet.

Geben Sie einen Hostnamen an, ist auch die Angabe der IP-Adresse zwingend. In diesem Fall werden die Einträge *hosts allow* und *hosts deny* der Konfigurationsdatei untersucht, um zu bestimmen, ob der Rechner mit der Adresse Zugriff auf den Server hat (Abbildung 3.6).

Wird bei der Untersuchung ein Fehler in der Konfigurationsdatei gefunden, gibt das Programm einen Wert von 1 zurück, ohne Fehler hingegen ist der Rückgabewert 0. Damit lässt sich das Programm auch in Skripten verwenden, die bei einem Fehler abgebrochen werden können.

Befinden sich in der Konfigurationsdatei keine Fehler, werden nach dem Drücken der Eingabetaste alle Details der angegebenen Dienste ausgegeben. Geben Sie beim Aufruf den Parameter *-s* an, erfolgt keine Nachfrage und Sie sehen die Ausgabe der Dienste sofort. Dies kann unter Umständen ziemlich lang sein, da auch der Wert einiger Parameter mit angezeigt wird.



```
Terminal <2>
Datei  Einstellungen  Hilfe

guest ok = Yes

[homes]
comment = Heimatverzeichnis
read only = No
create mask = 0750
browseable = No

[printers]
comment = All Printers
path = /tmp
create mask = 0700
print ok = Yes
browseable = No
bash-2.03# testparm
Load smb config files from /etc/smb.conf
Processing section "[netlogon]"
Processing section "[test]"
Processing section "[home]"
Processing section "[homes]"
Processing section "[printers]"
Loaded services file OK.
Press enter to see a dump of your service definitions
```

Abbildung 3.6: Die Anfangs-Ausgabe des Programms bei einer fehlerfreien Datei

3.2.12 testprns

Mit diesem Programm läßt sich überprüfen, ob es für den Namen eines SMB-Druckers in der lokalen Datei *printcap* eine Entsprechung gibt. Existiert sie, wird der Name des Druckers als gültig (valid) betrachtet, und er sollte auch im SMB-Netz funktionieren.

Als Parameter müssen Sie den Namen des Druckers und können Sie den Namen und den Pfad der *printcap*-Datei angeben. Sie sollte normalerweise im Verzeichnis */etc* zu finden sein. Befindet sie sich dort nicht, geben Sie einen entsprechenden Namen und Pfad an.

Bei der Angabe des Druckernamens müssen Sie etwas aufpassen. Spezifizieren Sie hier auch den Namen des Rechners, wird der Drucker auf keinen Fall gültig sein. Deshalb reicht die Angabe des Namens des Druckers völlig aus ...