7 Dateiarchive per FTP bereitstellen

Der klassische Internet-Dienst FTP (*F*ile *T*ransfer *P*rotocol) dient dazu, Dateien zwischen zwei Rechnern auszutauschen. Er ist für jede Hard- und Software-Plattform verfügbar, über die ein Internet-Zugang möglich ist.

Wie bei anderen Internet-Diensten gibt es hier Clients und Server.

FTP-Server dienen zum

- Bereitstellen von Dateien zum Fernladen (*Download*) durch Benutzer und zum
- Aufnehmen von Dateien zum Fernspeichern (*Upload*) durch Benutzer.

FTP-Clients nutzen diese Dienste anonym oder mit Benutzernamen und Passwort, wobei sie die Benutzerdaten im Klartext übertragen.

Die SuSE-Grundinstallation für den FTP-Server sieht nur einen anonymen Zugang zum Fernladen und Fernspeichern vor. Einen Zugang, bei dem sich jeder dem Linux-Server bekannte Nutzer per FTP mit seinem Home-Verzeichnis verbinden kann, muss man erst konfigurieren.

Wenn FTP-Benutzer sich frei im Verzeichnisbaum des Linux-Servers bewegen dürfen, entstehen Sicherheitsrisiken. Dieses Kapitel zeigt, wie Sie den FTP-Server sicherer gestalten können, sodass Sie Ihren Benutzern auch Fernzugriff auf ihre Heimat-Verzeichnisse erlauben können.

Lesen Sie hier bitte zuerst grundlegende Ideen zu FTP und zur sicheren FTP-Installation:

- Zugänge für anonyme Benutzer,
- Zugänge für normale Benutzer und
- Zugänge für spezielle Benutzer.

Sie lernen Grundlagen von `vsftpd` und von Konzepten kennen, wie man anonymen Benutzern und auch registrierten Systembenutzern den Zugriff auf einen kleinen Ast des Dateibaumes beschränkt, ihnen aber dennoch grundlegende Dateibefehle zur Verfügung stellt.

Ferner können Sie sich mit Sicherheitskonzepten für lesenden FTP-Zugriff (*Download*) und schreibenden FTP-Zugriff (*Upload*) in ein besonderes Upload-Verzeichnis vertraut machen. Wenn man anonymen Benutzern den Upload erlaubt, sollte man die von ihnen abgelegten Dateien erst nach einer gründlichen Kontrolle durch Systemadministratoren auch zum Download bereitstellen, um Risiken durch Viren und unerwünschte Inhalte zu begrenzen.

7.1 Wann benötigen Sie einen eigenen FTP-Server?

In einem reinen Windows-Netz tauschen Anwender Daten am einfachsten über die Netzwerkumgebung aus. Auf freigegebene Ordner kann man per SMB-Protokoll über das Netz zugreifen. Mit dem Programm *Samba* (siehe Kapitel 9) kann man Linux-Server so ausrüsten, dass sie sich in dieses System integrieren.

Sind im Intranet verschiedenartige Betriebssysteme vorhanden, oder sollen Dateien auch über das Internet angeboten werden, so empfiehlt sich ein eigener FTP-Server.

7.2 So arbeitet ein FTP-Server

FTP arbeitet mit je einem Verbindungskanal zum Steuern der Übertragung und für die Übertragung selbst:

- Auf dem Kommandokanal wartet der FTP-Server auf Befehle.
- Die eigentlichen Daten versendet oder empfängt der FTP-Server dann über einen gesonderten Datenkanal.

Als Kommandos erwartet der Server Befehle, die üblichen Unix- oder DOS-Kommandos entsprechen. Darunter sind Befehle zum Bewegen im Verzeichnisbaum und für die Datenübertragung. Im Abschnitt 5.6 dieses Buches listet die Tabelle 5.2 die wichtigsten FTP-Befehle aus Client-Sicht auf. Die wichtigsten Kommandos sind:

Befehl	*Erläuterung*
`ls, dir`	Anzeige des Inhaltsverzeichnisses
`cd <Zielverzeichnis>`	Verzeichniswechsel auf dem Server
`lcd <Zielverzeichnis>`	Verzeichniswechsel auf dem Client
`ascii, asc`	ASCII-Übertragungsmodus einschalten
`binary`	Binären Übertragungsmodus einschalten
`get <Datei>`	Angegebene Datei vom Server laden
`mget <Datei(en)>`	Mehrere Dateien vom Server holen, Wildcards `*` und `?` erlaubt
`put <Datei>`	Datei zum Server übertragen
`put <Datei(en)>`	Mehrere Dateien zum Server übertragen, Wildcards `*` und `?` erlaubt
`quit`	Programm beenden

Tabelle 7.1: FTP-Befehle und Erläuterungen

Die meisten Benutzer haben nur noch wenig direkt mit diesen Kommandos zu tun, da es für alle Betriebssysteme sehr komfortable FTP-Clients (z. B. *WS_FTP für Windows*) gibt, die sich wie der Windows-Dateimanager bedienen lassen (siehe Abschnitt 5.6). Im Hintergrund senden diese Client-Programme die FTP-Standardbefehle an den FTP-Server.

Beachten sollte man immer den Typ der Dateien:

- Binärdateien wie kompilierte Programme und Anwendungsdateien in proprietären Formaten wie `doc` und `xls` kann man unverändert kopieren. Dazu muss der binäre Übertragungsmodus eingeschaltet sein. .
- Reine Textdateien bestehen aus Text, der durch Zeilenschaltungen gegliedert ist, bei DOS/Windows der Zeichenfolge `#13#10`, bei Linux der Zeichenfolge `#10` und beim Mac der `#13`. Das Ende von Zeilen erkennt das sendende System an den jeweiligen Endmarkierungen, das Zielsystem ergänzt die eigenen Endmarkierungen. Damit Textdateien unabhängig vom Zeilenendezeichen auf allen Rechnern verwendet werden können, überträgt FTP sie im so genannten ASCII-Modus zeilenweise.

Daher gibt es auch zwei Möglichkeiten, Fehler zu machen:

- Würde man eine Textdatei binär zwischen verschiedenen Systemen kopieren, so könnten diese Zeilenschaltungen auf dem Zielsystem nicht mehr funktionieren: Ein Mac-Text z. B. besteht auf einem Linux-System nur aus einer einzigen Zeile. Besonders problematisch ist das beim Übertragen von Programmquelltext, der dann auf dem Zielsystem nicht kompiliert werden könnte. Im ASCII-Modus setzt FTP die Zeilenschaltungen richtig um.
- Würde man eine Programm- oder Anwendungsdatei wie eine ASCII-Textdatei kopieren, würden zufällig enthaltene Zeilenschaltungs-Zeichen verändert und damit die ganze Datei voraussichtlich unlesbar.

7.3 FTP-Server einrichten und verwalten

In der Unix-Welt gibt es viele verschiedene Programme für FTP-Server mit unterschiedlichen Konfigurationsmöglichkeiten und Sicherheitslevels. SuSE liefert mit den aktuellen Versionen der Distribution nicht mehr wie bisher den *wu.ftp* mit, sondern schlägt die Installation des *vsftpd* vor, der als besonders sicher und gut konfigurierbar gilt.

Bei SuSE Linux Professional 9.2 finden Sie ihn im Paket *vsftpd* in der Selektion *Netzwerk/Server* bzw. in der entsprechenden rpm-Datei auf der *CD4*.

Nach der Installation ist der *vsftpd* noch nicht sofort einsatzbereit. Sie müssen ihn erst im YaST-Kontrollzentrum unter *Netzwerkdienste • Netzwerkdienste (inetd)* durch Konfigurieren des *xinetd* (siehe Kapitel 4) einrichten:

Dazu müssen Sie hier den xinetd ggf. aktivieren, den Leuchtbalken auf die FTP-Zeile bringen und auf *Status wechseln* klicken. YaST ändert die Einstellungen, sobald Sie auf *Weiter* klicken.

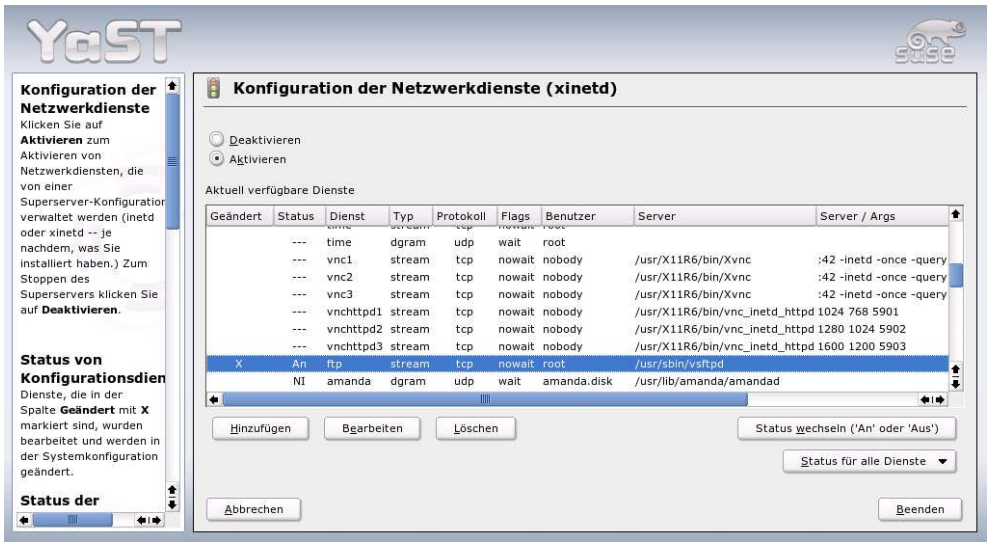

Abbildung 7.1: Eigene Konfigurationsdatei einbinden

Damit ist der FTP-Server grundsätzlich einsatzbereit.

Der FTP-Server benötigt nur eine globale Konfigurationsdatei, die Datei `/etc/vsftpd.conf`, die von SuSE sehr restriktiv voreingestellt ist.

Sie erlaubt keine individuelle Anmeldung von Benutzern, die einen Account auf dem System besitzen, wohl aber einen anonymen Zugriff mit den folgenden Daten:

Feld	*Inhalt*	*Erläuterung*
`Benutzername`	anonymous oder ftp	Wie oft habe ich mich da schon vertippt?
`Passwort`	beliebig	Üblich ist es hier, die eigene E-Mail-Adresse anzugeben; manche Systeme überprüfen die Gültigkeit.

Tabelle 7.2: Anonymer Zugriff von Benutzern

Diesen anonymen Zugriff nutzen Web-Browser, um Dateien zu beziehen und übermitteln beim Zugriff auf FTP-Adressen automatisch Benutzernamen und Passwort für den anonymen Zugriff.

Der FTP-Server stellt anonymen Benutzern eine sog. Changed-Root-Umgebung (*chroot*) zur Verfügung, um das restliche Dateisystem vor Zugriffen zu schützen.

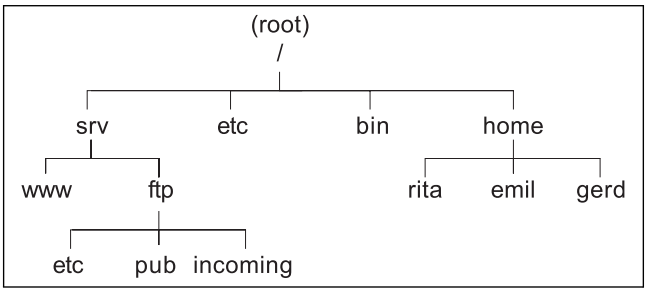

Abbildung 7.2: Changed-Root

Changed-Root-Umgebungen geben Benutzern nur einen Zugriff auf einen Teil des Dateisystems.

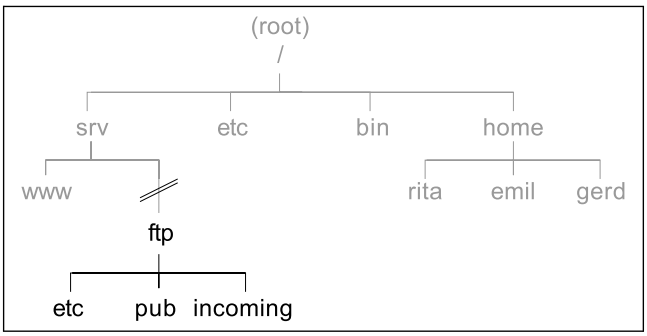

Abbildung 7.3: Dateisystem aus Sicht eines anonymen FTP-Nutzers

Sie geben Benutzern ein verändertes Wurzelverzeichnis (*Changed-Root*). Hier in der Installation ist das der Pfad `/srv/ftp`, das Home-Verzeichnis des Benutzers FTP. Für anonyme Benutzer ist das die Wurzel des Verzeichnisbaums, den Sie sehen können. Dieses System kann Sicherheitsrisiken vermindern.

Damit der User *ftp* die Dateien lesen kann, müssen die Eigentumsverhältnisse und die Dateirechte passend eingestellt sein. Mit

```
mkdir  /srv/ftp/incoming
chmod  733 /srv/ftp/incoming
mkdir  /srv/ftp/pub
chmod  755  /srv/ftp/pub
```

ist man da auf der sicheren Seite. Nun kann man die Dateien im Verzeichnis `pub` lesen, dort aber nicht schreiben. Im Verzeichnis `incoming` darf man generell schreiben, aber nicht lesen. Das unterbindet den ungefilterten Austausch von Dateien und dient wieder der Sicherheit.

7.4 Zugriffssteuerung mit vsftp

Das feine Prinzip der Changed-Root-Umgebung hat SuSE auch für eingetragene Benutzer voreingestellt. Wenn Sie Ihren System-Benutzern erlauben wollen, auf ihr Home-Verzeichnis zuzugreifen, müssen Sie die Konfigurationsdatei der *vsftpd* bearbeiten.

Die Konfigurationsdatei ist sehr gut lesbar und übersichtlich gegliedert.

- Im ersten Teil finden Sie wichtige Grundeinstellungen (*General Settings*),
- im zweiten Teil Einstellungen für die lokalen Benutzer (*Local FTP user Settings*),
- danach Einstellungen für die anonymen Nutzer (*Anonymous FTP user Settings*),
- der vierte Teil legt das Logging fest (*Log Settings*) und
- am Ende folgen ein paar weitere Einstellungen, die in keine der bisherigen Rubriken passen.

Die ersten Details müssen Sie im zweiten Teil ändern:

`/etc/vsftpd.conf` (Auszug ab Zeile 55)

```
# Local FTP user Settings
#
# Uncomment this to allow local users to log in.
#
local_enable=YES
#
```

Die hervorgehobene Zeile beginnt normalerweise mit dem Lattenzaun #, der die Einstellung deaktiviert. Entfernen Sie dieses erste Zeichen, um den lokalen Benutzern den Zugang zu ermöglichen.

Nach dieser minimalen Änderung der Konfigurationsdatei können sich die Benutzer Ihres Linux-Systems bereits per FTP anmelden; sie sehen dann den Inhalt ihres Home-Verzeichnisses.

Ein Problem besteht darin, dass die Benutzer ihr Home-Verzeichnis verlassen dürfen. Sie können sogar jedes Verzeichnis des gesamten Verzeichnisbaumes erreichen. Sinnvoller ist es, hier wieder eine Changed-Root-Umgebung einzurichten und die Benutzer im eigenen Home-Verzeichnis festzuhalten.

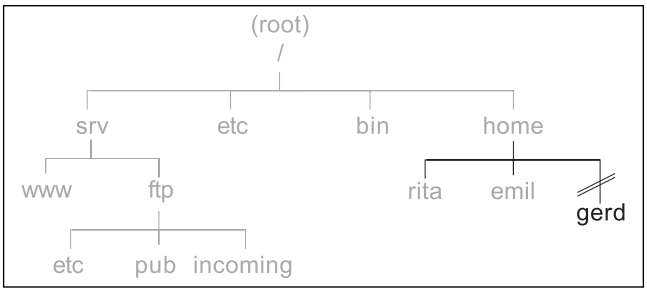

Abbildung 7.4: Dateisystem aus Sicht des autorisierten FTP-Nutzers *gerd*

Dieses Ziel erreichen Sie mit einer kleinen Änderung im zweiten Abschnitt der `vsftpd.conf`.

`/etc/vsftpd.conf` (Auszug ab Zeile 55)

```
# Local FTP user Settings
#
# Uncomment this to allow local users to log in.
#
local_enable=YES
#
# Default umask for local users is 077. You may wish to change
# this to 022, if your users expect that
# (022 is used by most other ftpd's)
#
#local_umask=022
#
# Uncomment to put local users in a chroot() jail in their
# home directory after login.
#
chroot_local_user=YES
```

Die hervorgehobene Zeile ist normalerweise durch Voranstellen des #-Zeichens deaktiviert. Entfernen Sie einfach dieses erste Zeichen, um sie einzuschalten.

Wenn sich Benutzer jetzt per FTP am System anmelden, sehen sie weiterhin ihr Home-Verzeichnis, können dessen Baum aber nicht mehr verlassen, da es für sie jetzt das Root-Verzeichnis ist.

Diese Einstellung gilt für alle Benutzer. Ausgewählten Benutzern können Sie abweichend von der Regel erlauben, auf das gesamte Dateisystem zuzugreifen. Tragen Sie diese in eine Liste ein, schalten Sie diese frei und verraten Sie dem FTP-Programm den Dateinamen der Liste.

`/etc/vsftpd.conf` (Auszug ab Zeile 55)

```
# Local FTP user Settings
#
# Uncomment this to allow local users to log in.
#
local_enable=YES
#
# Default umask for local users is 077. You may wish to change
# this to 022, if your users expect that
# (022 is used by most other ftpd's)
#
#local_umask=022
#
# Uncomment to put local users in a chroot() jail in their
# home directory after login.
#
chroot_local_user=YES
#
# You may specify an explicit list of local users to chroot()
# to their home directory. If chroot_local_user is YES, then
# this list becomes a list of users to NOT chroot().
#
chroot_list_enable=YES
#
# (default follows)
#
chroot_list_file=/etc/vsftpd.chroot_list
```

Die Benutzernamen dieser bevorzugten Benutzer tragen Sie in die Datei `/etc/vsftpd.chroot_list` ein und aktivieren die beiden hervorgehobenen Zeilen in der Konfigurationsdatei, die zuvor durch das #-Zeichen deaktiviert waren.

Damit kennt FTP dann drei Benutzer-Gruppen:

- Anonyme, hier ist das Verzeichnis `/srv/ftp` als Root-Verzeichnis eingestellt.
- System-Benutzer, die nicht in der Datei `vsftpd.chroot_list` aufgeführt sind; bei diesen ist das jeweilige Home-Verzeichnis als Root-Verzeichnis eingestellt.
- Nur System-Benutzer, die in der Datei `vsftpd.chroot_list` aufgeführt sind, behalten vollen Zugriff auf das Dateisystem.

7.5 Weitere Einstellungen für den vsftpd

In der bisher erstellten Konfiguration darf keine der Benutzergruppen auf den FTP-Server schreiben. Das ist auf alle Fälle eine sehr sichere Einstellung. Ohne großen Aufwand können Sie Ihren Benutzern auch schreibenden Zugriff erlauben.

Dazu müssen Sie im ersten Abschnitt der Konfigurationsdatei eine Zeile ändern:

`/etc/vsftpd.conf` (Auszug ab Zeile 9)

```
# General Settings
#
# Uncomment this to enable any form of FTP write command.
#
write_enable=YES
```

Wenn Sie hier wieder die hervorgehobene Zeile aktivieren, indem Sie das ursprünglich vorangestellte #-Zeichen entfernen, geben Sie allen Benutzer des Systems auch Schreibrechte.

Anonymen Benutzern sollten Sie keine Schreibrechte erteilen, auch wenn dies über Einstellungen im dritten Abschnitt der Konfigurationsdatei recht einfach möglich ist.

`/etc/vsftpd.conf` (Auszug ab Zeile 97)

```
#
# Uncomment this to allow the anonymous FTP user to upload
# files. This only has an effect if the above global write
# enable is activated. Also, you will obviously need to create
# a directory writable by the FTP user.
#
anon_upload_enable=YES
```

Zum Abschluss der Konfiguration können Sie noch einen kleinen Schönheitsfehler ausbügeln. Im Inhaltsverzeichnis per FTP tauchen statt der Benutzernamen und Gruppen die zugehörigen Nummern auf, da ftp innerhalb der Changed-Root-Umgebung natürlich nicht auf die Passwort-Dateien zugreifen kann.

```
ftp> dir
227 Entering Passive Mode (192,168,1,2,167,71)
150 Here comes the directory listing.
-rw-r--r--  1 500  100  131885 Jul 25 11:12 Bildschirmphoto1.png
-rw-r--r--  1 500  100  129616 Jul 22 11:46 Bildschirmphoto2.png
-rw-r--r--  1 500  100  109702 Jul 22 11:48 Bildschirmphoto3.png
drwx------  3 500  100     240 Jul 24 20:08 Desktop
drwxr-xr-x  5 500  100     168 Jul 21 15:05 Documents
drwxr-xr-x  2 500  100      80 Jul 21 15:05 public_html
226 Directory send OK.
```

Es gibt nun mehrere Schleichwege, dies zu umgehen:

- Sie könnten innerhalb ihrer Changed-Root-Umgebung ein Verzeichnis `/etc` anlegen und dort eine `passwd`-Datei ablegen. Das ist aber nicht besonders elegant, da Sie bei jeder Änderung der Nutzerdatei daran denken müssen, auch diese Dateien zu aktualisieren.
- Durch Verstecken der Identitäten per Konfigurationsdatei zeigt der FTP-Server statt der Zahlen immer die Zeichenfolge `ftp` an.

`/etc/vsftpd.conf` (Auszug ab Zeile 49)

```
#
# If  enabled,  all  user  and  group  information in
# directory listings will be displayed as "ftp".
#
hide_ids=YES
```

Die Anzeige sieht dann so aus:

```
ftp> dir
227 Entering Passive Mode (192,168,1,2,153,103)
150 Here comes the directory listing.
-rw-r--r--  1 ftp  ftp  131885 Jul 25 11:12 Bildschirmphoto1.png
-rw-r--r--  1 ftp  ftp  129616 Jul 22 11:46 Bildschirmphoto2.png
-rw-r--r--  1 ftp  ftp  109702 Jul 22 11:48 Bildschirmphoto3.png
drwx------  3 ftp  ftp     240 Jul 24 20:08 Desktop
drwxr-xr-x  5 ftp  ftp     168 Jul 21 15:05 Documents
drwxr-xr-x  2 ftp  ftp      80 Jul 21 15:05 public_html
226 Directory send OK.
```

Die Konfigurationsdatei bietet viele weitere nützliche Einstellungen, wie z. B. die Möglichkeit, Begrüßungsmeldungen zu definieren. Es lohnt sich, mit den Einstellungen zu experimentieren.

7.6 Zugriffe protokollieren und auswerten

Systemverwalter sollten Zugriffe auf allgemein zugängliche Dienste immer kontrollieren, insbesondere wenn sie auch anonyme Benutzer zulassen. Ansonsten besteht die Gefahr, dass sich die Speicher mit illegaler oder unerwünschter Software füllen.

In der bisherigen Konfiguration hält `vsftpd` wenig Informationen fest. In der Datei `/var/log/messages` protokolliert er die Zugriffe auf den Server:

```
Nov 23 17:15:37 boss vsftpd: Tue Nov 23 18:15:37 2004
  ↵ [pid 21099] CONNECT: Client "192.168.1.1"
Nov 23 18:15:50 boss vsftpd: Tue Nov 23 18:15:50 2004
  ↵ [pid 21098] [ftp] OK LOGIN: Client "192.168.1.1",
  ↵ anon password "debacher@linuxbu.ch"
```

Ebenfalls in der Datei `/var/log/vsftpd.log` speichert `vsftpd` folgendermaßen, welche Dateien Benutzer übertragen:

```
Nov 23 17:19:37 boss vsftpd: Tue Nov 23 18:19:37 2004
  ↵ [pid 21116] [ftp] OK DOWNLOAD: Client "192.168.1.1", "
  ↵ /pub/test.txt", 13 bytes, 8.78Kbyte/sec
```

Wollen Sie die Meldungen des FTP-Server nicht in der zentralen Datei `/var/log/messages` sehen, sondern in einer eigenen Datei, dann müssen Sie an den Abschnitt `Log Settings` der Konfigurationsdatei heran.

`/etc/vsftpd.conf` (Auszug ab Zeile 132)

```
# Log Settings
#
# Log to the syslog daemon instead of using an logfile.
#
# syslog_enable=YES
#
# Uncomment this to log all FTP requests and responses.
#
#log_ftp_protocol=YES
#
# Activate logging of uploads/downloads.
#
xferlog_enable=YES
#
# You may override where the log file goes if you like.
# The default is shown below.
#
#vsftpd_log_file=/var/log/vsftpd.log
```

Sie müssen hier in der ersten hervorgehobenen Zeile das Kommentarzeichen setzen und in der zweiten hervorgehobenen Zeile entfernen.

Damit `vsftpd` alle Details sehr ausführlich protokolliert, müssen Sie in der Konfigurationsdatei den Eintrag `log_ftp_protocol` ändern:

`/etc/vsftpd.conf` (Auszug ab Zeile 132)

```
# Log Settings
#
# Log to the syslog daemon instead of using an logfile.
#
# syslog_enable=YES
#
# Uncomment this to log all FTP requests and responses.
#
log_ftp_protocol=YES
```

So protokolliert der `vsftpd` alle Kommandos und jede Datei-Übertragung. Das Protokoll kann man recht einfach auswerten.

```
Tue Nov 23 17:30:46 2004 [pid 21194] FTP response:
  ↵ Client "192.168.1.1", "220 (vsFTPd 2.0.1)"
Tue Nov 23 17:30:53 2004 [pid 21194] FTP command:
  ↵ Client "192.168.1.1", "USER adams"
Tue Nov 23 17:30:53 2004 [pid 21194] [adams] FTP response:
  ↵ Client "192.168.1.1", "331 Please specify the password."
Tue Nov 23 17:30:57 2004 [pid 21194] [adams] FTP command:
  ↵ Client "192.168.1.1", "PASS <password>"
Tue Nov 23 18:30:57 2004 [pid 21193] [adams] FAIL LOGIN:
  ↵ Client "192.168.1.1"
Tue Nov 23 17:30:57 2004 [pid 21194] [adams] FTP response:
  ↵ Client "192.168.1.1", "530 Login incorrect."
```

Die Benutzeranmeldung ist gescheitert, Benutzername oder Passwort sind falsch.

Ja, es ist schön, seinen Autorennamen immer wieder zu sehen, aber besser in *Google* als in Beispielen im Buch ;-)

```
Tue Nov 23 17:33:06 2004 [pid 21206] FTP response:
  ↵ Client "192.168.1.1", "220 (vsFTPd 2.0.1)"
Tue Nov 23 17:33:07 2004 [pid 21206] FTP command:
  ↵ Client "192.168.1.1", "USER debacher"
Tue Nov 23 17:33:07 2004 [pid 21206] [debacher] FTP response:
  ↵ Client "192.168.1.1", "331 Please specify the password."
Tue Nov 23 17:33:13 2004 [pid 21206] [debacher] FTP command:
  ↵ Client "192.168.1.1", "PASS <password>"
Tue Nov 23 18:33:13 2004 [pid 21205] [debacher] OK LOGIN:
  ↵ Client "192.168.1.1"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "230 Login successful."
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "SYST"
```

```
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "215 UNIX Type: L8"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "FEAT"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "211-Features:"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " EPRT??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " EPSV??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " MDTM??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " PASV??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " REST STREAM??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " SIZE??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", " TVFS??"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "211 End"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "PWD"
Tue Nov 23 18:33:13 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "257 "/home/debacher""
```

Diese Benutzeranmeldung ist erfolgreich. Anschließend hat der FTP-Client das aktuelle Verzeichnis (`PWD`) abgefragt.

```
Tue Nov 23 18:35:23 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "CWD .."
Tue Nov 23 18:35:23 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "250 Directory successfully changed."
Tue Nov 23 18:35:23 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "PWD"
Tue Nov 23 18:35:23 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "257 "/home""
```

Ein Verzeichniswechsel (`CWD..`) auf die nächsthöhere Verzeichnisebene. Das Ergebnis zeigt deutlich, dass sich dieser Benutzer frei im Dateisystem bewegen darf, also keine Changed-Root-Umgebung besitzt.

Eine erfolgreiche Datenübertragung hinterlässt in der `Log-Datei` sehr unterschiedliche Einträge, im folgenden Beispiel zuerst die Umschaltung in den Binärmodus, dann die eigentliche Datenübertragung und die Empfangsbestätigung.

```
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "TYPE I"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "200 Switching to Binary mode."
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "SIZE xinetd.png"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "213 108387"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "EPSV"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "229 Entering Extended Passive Mode
  ↵ (|||40169|)"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "RETR xinetd.png"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "150 Opening BINARY mode data
  ↵ connection for xinetd.png (108387 bytes)."
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "226 File send OK."
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] OK DOWNLOAD:
  ↵ Client "192.168.1.1", "/home/debacher/xinetd.png",
  ↵ 108387 bytes, 8901.41Kbyte/sec
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP command:
  ↵ Client "192.168.1.1", "MDTM xinetd.png"
Tue Nov 23 18:36:57 2004 [pid 21207] [debacher] FTP response:
  ↵ Client "192.168.1.1", "213 20041123161443"
```

Insgesamt entstehen bei dieser Einstellung sehr viele Daten.

Blicken Sie regelmäßig in die Protokolldateien, und achten Sie vor allem auf die Häufung von Login-Fehlern, die von Hack-Versuchen herrühren könnten. Achten Sie auch darauf, was Benutzer mit vollem Dateizugriff auf Ihrem System treiben. Spätestens Zugriffe dieser Benutzer auf Systemdateien sollten Sie dazu veranlassen, diese auf eine Changed-Root-Umgebung zu beschränken.

Wer einen Upload-Ordner für anonyme Benutzer anbietet, sollte dort abgelegte Dateien regelmäßig prüfen und gegebenenfalls zum Download anbieten.

7.7 Statistische Auswertung mit Webalizer

Das Programm *Webalizer* haben Sie bereits in Kapitel 6 kennen gelernt. Es dient, wie auch der Name schon sagt, ursprünglich zur Auswertung von Log-Dateien von Web-Servern.

Die aktuelle Version des Programms kann auch Informationen aus der Datei `/var/log/vsftpd.log` auszuwerten. Auf gut besuchten Servern ist das sicherlich eine große Hilfe.

Leider müssen Sie zur Auswertung der vsftpd-Log-Datei das Format auf das Standard-Log-Format für Web-Server umstellen, da Webalizer die Datei sonst nicht sinnvoll auswerten kann. Die entsprechenden Einträge erscheinen dann in der Datei `/var/log/xferlog`.

`/etc/vsftpd.conf` (Auszug ab Zeile 132)

```
# Log Settings
#
# Log to the syslog daemon instead of using an logfile.
#
# syslog_enable=YES
#
# Uncomment this to log all FTP requests and responses.
#
log_ftp_protocol=YES
#
# Activate logging of uploads/downloads.
#
xferlog_enable=YES
#
# You may override where the log file goes if you like.
# The default is shown below.
#
#vsftpd_log_file=/var/log/vsftpd.log
#
# If you want, you can hsave your log file in standard ftpd
# xferlog format. Note: This disables the normal logging unless
# you enable dual_log_enable below.
#
xferlog_std_format=YES
#
# You may override where the log file goes if you like.
# The default is shown below.
#
#xferlog_file=/var/log/xferlog
#
```

```
# Enable this to have booth logfiles. Standard xferlog and
# vsftpd's own style log.
#
#dual_log_enable=YES
```

Soll der vsftpd die Übertragungen zusätzlich in seinem eigenen Format protokollieren, so müssen Sie hier noch die letzte Zeile aktivieren.

Die folgende Beschreibung geht davon aus, dass Sie die FTP-Statistik zusätzlich zu einer eventuell vorhandenen Web-Statistik pflegen möchten.

Sie müssen zuerst ein Verzeichnis einrichten, in dem Webalizer die FTP-Statistik ablegen kann. Eine Möglichkeit wäre `/srv/www/htdocs/ftpalizer`:

```
mkdir  /srv/www/htdocs/ftpalizer
```

Nun müssen Sie eine zweite Konfigurationsdatei erzeugen, die für die Analyse der FTP-Log-Datei angepasst ist. Sie können dazu einfach die vorhandene Datei kopieren, z. B. als `ftpalizer.conf`:

```
cp  /etc/webalizer.conf  /etc/ftpalizer.conf
```

Damit der Webalizer auch mit der Datei `xferlog` richtig umgehen kann, müssen Sie die Konfigurationsdatei anpassen. Am wichtigsten ist dabei die Einstellung, die dem Webalizer mitteilt, dass es sich um eine Log-Datei des FTP-Servers und nicht um eine des Web-Servers handelt.

`/etc/ftpalizer.conf` (Auszug ab Zeile 23)

```
# LogFile defines the web server log file to use.
# If not specified here or on on
# the command line, input will default to STDIN.

LogFile        /var/log/xferlog

# LogType defines the log type being processed.
# Normally, the Webalizer
# expects a CLF or Combined web server log as input.
# Using this option, you can process ftp logs as well
# (xferlog as produced by wu-ftp and others).
# Values can be 'web' or 'ftp', with 'web' the default.

LogType ftp

# OutputDir is where you want to put the output files.
# This should be a full path name, however relative ones
# might work as well.
```

```
# If no output directory is specified, the current directory
# will be used.

OutputDir       /srv/www/htdocs/ftpalizer
```

Nun können Sie den Webalizer starten und ihm die eben erstellte Konfigurationsdatei konkret über den Parameter `-c` angeben.

```
webalizer  -c  /etc/ftpalizer.conf
```

Auch diesen Programmaufruf sollten Sie in die Crontab von *root* mit aufnehmen, um damit die Auswertung tagesaktuell zu pflegen.

Auch wenn das Zusammenspiel zwischen dem vsftpd und Webalizer nicht ganz optimal ist, so ist die Entscheidung von SuSE, auf den vsftpd umzustellen, auf alle Fälle sinnvoll. Die Konfiguration des vsftpd ist deutlich übersichtlicher und stringenter als die seines Vorgängers. Auch ist die Zahl der Sicherheitswarnungen (siehe Kapitel 15) deutlich geringer. Den Autoren sind keine aktuellen Warnungen für den vsftpd bekannt. Es macht sich einfach bemerkbar, wenn Sicherheit schon beim Grund-Design eines Programms die zentrale Rolle spielt.