HOWTO zum File Transfer Protocol "FTP"

(C) 2005-2016 T.Birnthaler/H.Gottschalk <howtos(at)ostc.de>
OSTC Open Source Training and Consulting GmbH
http://www.ostc.de

$Id: ftp-HOWTO.txt,v 1.18 2019/11/26 19:37:07 tsbirn Exp $

Diese Dokument beschreibt das Protokoll FTP, die Konfiguration eines FTP-Server "ftpd" und die Verwendung des Kommandozeilen-FTP-Clients "ftp".

INHALTSVERZEICHNIS

1) Einführung
2) FTP-Client "ftp"
2.1) Optionen des FTP-Client "ftp"
2.2) Befehle im FTP-Client "ftp"
2.3) Umgebungsvariablen des FTP-Client "ftp"
3) FTP-Transfer automatisieren
3.1) Per Konfigurationsdatei "~/.netrc"
3.1.1) Beispiel 1
3.1.2) Beispiel 2
3.2) Per Here-Dokument in Shell-Skript
4) FTP-Server aktivieren
4.1) Internet Daemon "inetd"
4.2) Extended Internet Daemon "xinetd"
4.3) Test ob FTP-Server erreichbar ist
5) man-Pages
6) Sicherheit

---

1) Einführung

FTP (file transfer protocol) ist ein sehr alter Dienst zur Dateiverwaltung auf fremden Rechnern und zum Transfer von Dateien zwischen beliebigen Rechnern. Er ist unabhängig vom Betriebssystem und wird auch heute noch gerne benutzt, um Dateien zwischen Rechnern auszutauschen. Der Client heißt "ftp", der Server heißt "ftpd" (Daemon).

2) FTP-Client "ftp"

FTP führt eine Anmeldung auf einem fremden Rechner mit Benutzername + Passwort durch. Es stellt dann eine Art "Shell" (mit eigenem Befehlssatz) zur Verfügung, die während der Sitzung einen Prompt der Form

```
ftp>
```

anzeigt. Die Anmeldedaten beruhen auf den normalen Linux-Benutzerkonten des FTP-Servers. Alle in der Datei

```
/etc/ftpusers
```

auf dem FTP-Server aufgelistete Benutzer (pro Zeile ein Benutzername) werden vom FTP-Server abgewiesen. Hier sollte grundsätzlich der Benutzer "root" und alle Systembenutzer eingetragen werden.

Häufig kann man sich auch als Benutzer "guest" oder "anonymo(u)s" anmelden, das Passwort ist dann beliebig, sollte aber per Konvention die E-Mail-Adresse des Benutzers sein. Auf dem Server wird man in diesem Fall zum Benutzer "ftp" (dessen Heimat-Verz. sollte daher abgesichert sein).

Der allgemeine Aufruf des FTP-Clients "ftp" lautet

```
ftp [OPTIONEN] [HOST [PORT]]
```

und führt zu einer Verbindung zum FTP-Server HOST über den Port PORT (Std: 21).

2.1) Optionen des FTP-Client "ftp"

Folgende OPTIONEN können beim Aufruf angegeben werden:

```
+----+-------------+-------------------------------------------------------+
```

Opt	Name	Beschreibung
-p	passive	Passiven Modus für Datentransfer benutzen
-i	interactive	Keinen Prompt für interaktive Benutzung anzeigen
-n	noautologin	Kein automatisches Login gemäß "~/.netrc" durchführen
-e	edit	Kommandohistorie und -editierung abschalten
-g	globbing	Dateinnamen-Expansion abschalten
-v	verbose	Server-Antworten und Datentransfer-Statistik anzeigen
-d	debug	Debugging aktivieren

Hinweis: Die Option "-p" kann durch Aufruf von "pftp" statt "ftp" automatisch gesetzt werden.

2.2) Befehle im FTP-Client "ftp"

Die wichtigsten Befehle im FTP-Client "ftp" lauten (P = Port, PW = Passwort, LD = Local directory, RD = Remote directory, LF = Local file, RF = Remote file, NF = New file):

Befehl	Beschreibung
![CMD...]	Externes Kommando (oder Shell) auf Client aufrufen
open HOST [P]	Verbindung zum Rechner HOST Port P aufbauen (Std: 21)
user USER [PW]	Als Benutzer USER mit Passwort PW anmelden
close	Verbindung zum FTP-Server schließen (auch "disconnect")
bye/quit/exit	FTP-Client verlassen
help/?	Liste aller FTP-Befehle anzeigen
help/? CMD	Hilfe zu FTP-Befehl CMD anzeigen
form FORMAT	Transferformat FORMAT setzen (Std: file)
ascii/binary	CR/LF-Übersetzung an/aus (Std: an)
bell	Nach jeder übertragenen Datei Ton generieren (Std: an)
case	GROSS(Server) --> kleinschreib(Client) an/aus (Std: an)
cr	Carriage-Return aus Daten entfernen an/aus (Std: an)
glob	Dateinamen-Expansion *? für m*-Komandos an/aus (Std: an)
mode [MODE]	Transferformat setzen (Std: stream)
prompt	Interaktiven Prompt an/aus (Std: an)
verbose	Server-Antw. + Datentransfer-Stat. anzeigen (Std: aus)
cd RD	Auf Server in Verz. RD wechseln
cdup	Auf Server in Eltern-Verz. ".." wechseln
lcd [LD]	Auf Client in Verz. LD wechseln (Std: $HOME)
pwd	Aktuelles Verz. auf Server anzeigen
dir [RD] [LF]	Verz.inhalt RD (Std: akt. Verz.) des Server auflisten
mdir RD... LF	Verz.inhalte RD... des Server auflisten + in LF speichern
ls [RD] [LF]	Verz.inhalt RD des Servers auflisten (Std: akt. Verz.)
mls RD... LF	Verz.inhalt RD... des Servers auflisten + in LF speichern
nlist [RD][LF]	Verz.inhalt RD des Servers auflisten (Std: akt. Verz.)
get RF [LF]	Datei RF vom Server holen (auch "recv")
put LF [RF]	Datei LF auf Server ablegen (auch "send")
mget RF	Mehrere Dateien RF vom Server holen (*?-Notation)
mput LF	Mehrere Dateien LF auf Server ablegen (*?-Notation)
append LF [RF]	Datei LF an Datei RF auf Server anhängen
delete RF	Server-Datei RF löschen
mdelete RF	Mehrere Dateien RF auf Server löschen (*?-Notation)
rename RF NF	Datei RF auf Server umbenennen nach NF
mkdir RD	Verz. RD auf Server anlegen
rmdir RD	Verz. RD auf Server löschen
chmod MODE RF	Zugriffsrechte MODE für Datei RF auf Server setzen
umask [MASK]	umask MASK für Server definieren bzw. anzeigen
macdef MAKRO	Makro MAKRO definieren (Befehlliste bis Leerzeile)
$MAKRO ...	Makro MAKRO mit Argumenten ... ausführen

Eigenschaften von Makros (macdef):
* Limit: Max. 16 Makros + 4096 Zeichen in allen Makros
* $N = N. Aufrufargument
* $i = Schleife über alle Aufrufargumente

2.3) Umgebungsvariablen des FTP-Client "ftp"

Die für den FTP-Client "ftp" relevanten Umgebungsvariablen sind:

Variable	Beschreibung
HOME	Für "~/.netrc"
PAGER	Z.B. "more" oder "less"
SHELL	Shell für Shell-Escape
TERM	Terminaltyp für Prompt
FTPANONPASS FTPMODE FTPPROMPT FTPSERVER FTPSERVERPORT	active, auto (Std), gate, passive "ftp>"

3) FTP-Transfer automatisieren

Das Automatisieren eines FTP-Transfers ist durch die Konfigurations-Datei "~/.netrc" oder durch entsprechende Optionen + Here-Dokument beim Aufruf des FTP-Client möglich.

3.1) Per Konfigurationsdatei "~/.netrc"

Die Konfigurations-Datei "~/.netrc" steht im eigenen Benutzer-Verz. und sollte nur für den Benutzer lesbar sein. Sie kann folgende Befehle zur Steuerung eines automatischen ftp-Transfers enthalten:

Syntax	Beschreibung
machine HOST	Eintrag mit Befehlen für FTP-Server HOST
default	Eintrag für restliche Hosts außer aufgelistete
login USER	Anmeldename
password PASS	Anmeldepasswort
account PASS	Anmeldepasswort
macdef MAKRO	Makro MAKRO definieren (Befehlliste bis Leerzeile)
$MAKRO ...	Makro MAKRO mit Argumenten ... ausführen
macdef init	Automatisch beim Start des FTP-Client auszuführen

3.1.1) Beispiel 1

Datei "~/.netrc" hat folgenden Inhalt:

```
machine 192.168.66.99 | # NICHT ma_s_chine!
login guest           | # Benutzername
password guest        | # Passwort
macdef init           | # Automatisch beim Start des FTP-Client ausführen
get ftp.txt           | #   1. FTP-Befehl
bye                   | #   2. FTP-Befehl
                      | # LEERZEILE beendet Makrodefinition!
```

Die LEERZEILE ist notwendig, weil mehrere solcher Abschnitte für verschiedene Rechner existieren können, die dadurch getrennt werden. Die Datei ".netrc" muß im Heimat-Verz. stehen und darf nur für den Besitzer lesbar + schreibbar sein:

```
chmod go-rwx ~/.netrc        # oder
chmod 700    ~/.netrc
```

Ein automatischer FTP-Transfer erfolgt dann z.B. durch:

```
ftp 192.168.66.99
```

3.1.2) Beispiel 2

Datei "~/.netrc" hat folgenden Inhalt:

```
machine sun1      # Maschine "sun1"
login guest       #   Anmeldung
password guest    #   Anmeldung
macdef init       # Makro "init"
binary            #   1. FTP-Befehl
cd /etc           #   2. FTP-Befehl
get hosts         #   3. FTP-Befehl (Datei "hosts" holen)
get passwd        #   4. FTP-Befehl (Datei "passwd" holen)
quit              #   5. FTP-Befehl
                  # LEERZEILE schließt Makro "init" ab
machine sun3      # Maschine "sun3"
login dozent      #   Anmeldung
password test     #   Anmeldung
macdef init       # Makro "init"
binary            #   1. FTP-Befehl
cd /tmp           #   2. FTP-Befehl
put test.txt      #   3. FTP-Befehl (Datei "test.txt" ablegen)
quit              #   4. FTP-Befehl
                  # LEERZEILE schließt Makro "init" ab
```

Gibt man "ftp sun1" ein, dann werden die 2 Dateien "hosts" und "passwd" automatisch von "sun1" geholt.

Gibt man "ftp sun3" ein, dann wird die Datei "test.txt" automatisch in das Verz. "/tmp" von "sun3" kopiert.

Alle anderen FTP-Kommandos funktionieren wie bisher und müssen interaktiv durchgeführt werden.

3.2) Per Here-Dokument in Shell-Skript

Automatischer FTP-Zugriff per Here-Dokument in einem Shell-Skript (-v=verbose, -n=noautologin, <<=Here-Dokument, <<-=Einrücken per TAB erlaubt):

```
#/bin/sh
ftp -v -n ftp.doma.in <<-EOF        # Beginn Here-Dokument
    user anonymous tsbirn@ostc.de   #   1. FTP-Befehl
    binary                          #   2. FTP-Befehl
    cd /PATH/TO/DIR                 #   3. FTP-Befehl
    pwd                             #   4. FTP-Befehl
    get FILE                        #   5. FTP-Befehl
    bye                             #   6. FTP-Befehl
EOF                                 # Ende Here-Dokument
```

4) FTP-Server aktivieren

Damit der eigene Rechner per "ftp" erreichbar ist, muß darauf ein "FTP-Server" gestartet werden. Dieser läuft aber normalerweise nicht permanent, sondern wird nur bei Bedarf vom Dienst "inetd/xinetd" gestartet.

Für viele Dienste ist der Daemon "inetd" (Internet-Daemon) oder "xinetd" (eXtended Internet Daemon) als Stellvertreter aktiv, der auf eine Anforderung an einen Dienst wartet und ihn erst dann startet (z.B. ftp, telnet, finger, ...).

4.1) Internet Daemon "inetd"

Die Konfigurationsdatei des "inetd" heißt:

```
/etc/inetd.conf    # inetd
```

Sie enthält pro Dienst eine Zeile, inaktive Dienste sind darin mit einem "#" am Anfang auskommentiert. Die Zeile für den FTP-Daemon (es gibt 3 verschiedene) lautet:

```
# ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  wu.ftpd -a
```

```
# ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  proftpd
# ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  in.ftpd
```

Um z.B. den FTP-Server "in.ftpd" zu aktivieren, ist das Doppelkreuz vor der letzten Zeile zu entfernen:

```
# ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  wu.ftpd -a
# ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  proftpd
ftp  stream  tcp  nowait  root  /usr/sbin/tcpd  in.ftpd
```

Dem inetd-Daemon muß mitgeteilt werden, dass seine Konfigurationsdatei geändert wurde. Dies erfolgt (als "root") durch ("rc"=run control):

```
rcinetd reload                 # Falls der "inetd" schon läuft
/etc/init.d/inetd reload
/etc/rc.d/inetd reload
```

Ob der inetd-Daemon läuft, kann man feststellen durch:

```
rcinetd status
/etc/init.d/inetd status
/etc/rc.d/inetd status
```

Falls man ihn starten oder neu starten will:

```
rcinetd start
/etc/init.d/inetd start
/etc/rc.d/inetd start

rcinetd restart
/etc/init.d/inetd restart
/etc/rc.d/inetd restart
```

4.2) Extended Internet Daemon "xinetd"

Die Konfigurations-Dateien und -Verz. des "xinetd" heißen:

```
/etc/xinetd.conf   # Basisdefinitionen
/etc/xinetd.d/*    # Eine Datei pro verwaltetetem Dienst
```

"xinetd.conf" enthält Default-Einstellungen und liest alle Dateien aus dem Verz. "/etc/xinetd.d" ein:

Datei "xinetd.conf"

```
defaults
{
# Please note that you need a log_type line
# to be able to use log_on_success and
# log_on_failure. The default is the following:
# log_type = SYSLOG daemon info
}

includedir /etc/xinetd.d
```

Das Verz. "/etc/xinetd.d" enthält pro Dienst eine Datei --- benannt nach dem Dienstnamen --- mit den dafür gültigen Einstellungen:

Datei "daytime"

```
service daytime
{
    disable     = yes              # Inaktiv!
    type        = INTERNAL
    id          = daytime-stream
    socket_type = stream
    protocol    = tcp
    user        = root
    wait        = no
}
```

Inaktive Dienste sind darin mit "disable = yes" gekennzeichnet. Soll ein Dienst aktiviert werden, ist statt dessen "disable = no" einzutragen:

Datei "daytime"

```
service daytime
{
    disable     = no              # Aktiv!
    type        = INTERNAL
    id          = daytime-stream
    socket_type = stream
    protocol    = tcp
    user        = root
    wait        = no
}
```

Dem xinetd-Daemon muß mitgeteilt werden, dass seine Konfigurationsdatei geändert wurde. Dies erfolgt (als "root") durch ("rc"=run control):

```
rcxinetd reload                 # Falls der "xinetd" schon läuft
/etc/init.d/xinetd reload
/etc/rc.d/xinetd reload
```

Ob der xinetd-Daemon läuft, kann man feststellen durch:

```
rcxinetd status
/etc/init.d/xinetd status
/etc/rc.d/xinetd status
```

Falls man ihn starten oder neu starten will:

```
rcxinetd start
/etc/init.d/xinetd start
/etc/rc.d/xinetd start

rcxinetd restart
/etc/init.d/xinetd restart
/etc/rc.d/xinetd restart
```

4.3) Test ob FTP-Server erreichbar ist

Anschließend können andere Rechner mit dem eigenen Rechner über FTP Verbindung aufnehmen.

```
ftp HOSTNAME
ftp HOSTIP
```

Zum Testen kann man erst mal eine FTP-Verbindung zu sich selbst aufnehmen:

```
ftp localhost   # Eigenener Rechnername
ftp 127.0.0.1   # Eigene interne IP
ftp OWNIP       # Eigene externe IP
```

Anstelle des Kommandos "ftp" kann auch ein beliebiger Browser mit folgender URL per FTP-Protokoll auf andere Rechner zugreifen:

```
ftp://HOSTNAME                # Ohne Anmeldedaten (werden abgefragt)
ftp://guest@HOSTNAME          # Mit Benutzername (Passwort wird abgefragt)
ftp://guest:geheim@HOSTNAME   # Mit Anmeldedaten
ftp://guest:geheim@HOSTIP     # Mit Anmeldedaten
```

5) man-Pages

man-Page	Inhalt
ftp(1)	FTP-Client
netrc(5)	FTP-Client Konfigurationsdatei (Automatismen)
ftpd(8)	FTP-Server
vsftpd(8)	FTP-Server (very small/secure ftpd)

pureftpd(8)	FTP-Server	
ftpusers(8)	FTP-Server Konfigurationsdatei (nicht erlaubte Benutzer)	

6) Sicherheit

Sinnvoll wäre, auf einem Rechner mit FTP-Server einen Benutzer "guest" anzulegen, der das (allgemein bekannte) Passwort "guest" bekommt, sich aber nicht per Shell anmelden kann (Shell-Eintrag: /bin/false). Dann kann man sich über diesen Account per FTP anmelden.

Nachdem sich anschließend jeder per "ftp" auf dem Rechner anmelden kann und alles ausspionieren kann (NICHT verändern!), sollte das Heimat-Verz. aller anderen Benutzer gegen "ls" und "cd" geschützt werden, indem die Rechte "Lesen" + "Ausführen" für die "Gruppe" und "alle anderen" entfernt werden durch:

```
chmod go-rwx /home/EIGENER_NAME   # g=group, o=others, NICHT u=user!
chmod 700    /home/EIGENER_NAME   # analog
```

Umgekehrt sollten ALLE Rechte für alle Benutzer für das Heimat-Verz. von "guest" gesetzt werden durch:

```
chmod a+rwx /home/guest   # a = ugo = user+group+others
chmod 777   /home/guest   # analog
```

Anschließend können ALLE Dateien dort ablegen und holen!

Die Datei "/etc/ftpusers" enthält eine Liste von Benutzern, die sich nicht per FTP am FTP-Server anmelden dürfen. Hier sollte der Benutzer "root" sowie alle Systembenutzer aufgelistet werden:

Datei "/etc/ftpusers"

```
root       # Minimum!
ftp        # Falls anonyme Anmeldung nicht erlaubt sein soll
anonymous  # Falls anonyme Anmeldung nicht erlaubt sein soll
...        # Weitere Systembenutzer verbieten (cups, www-data, ...)
```

ACHTUNG: Bitte VOR dem Ändern einer (globalen) Konfigurationsdatei als "root" eine SICHERHEITSKOPIE dieser Datei anlegen (z.B. per RCS). Ab und zu macht man Syntax-Fehler oder sonstige Fehler beim Editieren und dann funktioniert das Linux-System oder Teile davon nicht mehr richtig. Über das Rettungs-System kann dann das Original wieder zurückkopiert und das System wieder funktionsfähig gemacht werden.