vsftpd – der Very Secure FTP Daemon, eine Einführung

by Mario M. Knopf
<netzmeister/at/neo5k/dot/org>

About the author:

Mario beschäftigt sich leidenschaftlich gerne mit Linux, Netzwerken und sicherheitsrelevanten Themen. Nebenbei betreut er in seiner Freizeit die beiden Webpräsenzen neo5k.org und linuxwallpapers.de.

Abstract:

Dieser Artikel soll eine grundlegende Einführung zum "Very Secure FTP Daemon" darstellen. Beginnen möchte ich mit einer allgemeinen Beschreibung von FTP und vsftpd. Danach sehen wir uns die Installation, Konfiguration und Startmöglichkeiten des vsftp−Daemons genauer an. Abschließend wird noch ein kurzer Funktionstest durchgeführt werden.

Einleitung

Das File Transfer Protocol (FTP) dient zur plattformunabhängigen Dateiübertragung im Internet und basiert auf einer Server−Client−Architektur. RFC 959 [1] schreibt vor, daß FTP in zwei unterschiedliche Kanäle getrennt wird, wobei ein Kanal für die Daten (TCP−Port 20) und der andere zur Steuerung (TCP−Port 21) dient. Über den Steuerkanal tauschen die beiden Seiten (Server und Client) Kommandos aus, welche dann den Datentransfer einleiten.
Eine FTP−Verbindung verläuft in vier Schritten:

- Benutzerauthentifizierung
- Aufbau des Steuerungskanals
- Aufbau des Datenkanals
- Beenden der Verbindung

FTP benutzt als Transportprotokoll das verbindungsorientierte TCP (Transmission Control Protocol), welches sicherstellt, daß die Daten auch wirklich beim Empfänger ankommen. Somit braucht sich FTP nicht um einen möglichen Paketverlust bzw. eine Fehlerkontrolle bei der Dateiübertragung kümmern. Grob formuliert sorgt TCP also dafür, daß jedes einzelne Datenpaket nur einmal ankommt – fehlerfrei bei der Übertragung und in der richtigen Reihenfolge.

Bei der Dateiübertragung unterscheidet man drei Transferarten, wobei der Abschluß des Transfers im Stream−Modus durch ein End−of−File (EOF), bei den beiden anderen Übertragungsarten durch ein

End−of−Record (EOR) gekennzeichnet wird.

- Stream
- Block
- Compressed

Des Weiteren gibt es zwei verschiedene Transfermodi:

- ASCII
- Binary

Der ASCII−Modus dient zur Übertragung von Textdateien, wohingegen der Binary−Modus beispielsweise zum Transfer von Programmen oder dergleichen dient. Der Benutzer muß den Transfermodus für gewöhnlich nicht explizit auswählen, da mittlerweile alle FTP−Clients die zu übertragende Datei erkennen und automatisch umschalten.

Da die Übermittlung der Benutzerkennung und des Passworts zur Authentifizierung nicht verschlüsselt wird, ist es sehr wichtig, ausdrücklich auf dieses potentielle Sicherheitsrisiko hinzuweisen. Aus diesem Grund machte man sich Gedanken über die Sicherheit von FTP. Im Oktober 1997 wurde schließlich das RFC 2228 [2] veröffentlicht, welches sicherheitsspezifische Erweiterungen für das File Transfer Protocol definiert.

vsftpd

vsftpd stellt einen FTP−Server für unixoide Betriebssysteme dar und läuft somit auf Plattformen wie Linux, *BSD, Solaris, HP−UX und IRIX. Dabei unterstützt vsftpd viele Merkmale, die man bei anderen FTP−Servern unter Umständen schmerzlich vermisst. Einige davon sind beispielsweise:

- sehr hohe Sicherheitsansprüche
- Bandbreitenbegrenzung
- gute Skalierbarkeit
- Möglichkeit, virtuelle User zu erstellen
- IPnG−Unterstützung
- überdurchschnittliche Performance
- Möglichkeit, virtuelle IPs zu vergeben
- hohe Geschwindigkeit

Der Name vsftpd steht für "Very Secure FTP Daemon", welcher auch gleich eines der Hauptanliegen des Entwicklers Chris Evans wiederspiegelt. Bei der Entwicklung und dem Design des FTP−Servers wurde von Anfang an sehr viel Wert auf Sicherheit gelegt.

Als Beispiel hierfür kann die Tatsache genannt werden, daß vsftpd im chroot−Modus betrieben wird. chroot bedeutet, daß einem Programm (in diesem Fall vsftpd) ein neues Wurzelverzeichnis (/) zugewiesen wird und es somit nicht mehr auf Programme oder Dateien außerhalb dieses Verzeichnisses zugreifen darf – es wird sozusagen in einem Gefängnis eingesperrt. Sollte nun ein potentieller Angreifer den FTP−Server kompromittieren, ist er vom übrigen System abgeschottet und kann dadurch keinen größeren Schaden anrichten. Wer weiterführende Informationen zu chroot sucht, sollte sich den Artikel unter [3] ansehen. Wer sich jedoch besonders für die Implementierung und das Design der diversen Sicherheitsmechanismen von vsftpd interessiert, dem sei [4] empfohlen.

Durch diese umfangreichen Merkmale – wobei der Anspruch an die Sicherheit des FTP–Dienstes höchste Priorität genießen sollte – hebt sich vsftpd deutlich von anderen FTP–Servern ab. Als Negativbeispiel sei hier der WU–FTPD [5] genannt, welcher in den vergangenen Jahren ständig durch diverse Sicherheitslücken auffiel.

Installation

Die Installation des vsftp–Daemons verläuft recht einfach, da jede größere Distribution fertige RPM–Pakete zu vsftpd bereitstellt, welche in den meisten Fällen sogar schon installiert sind. Alternativ besorgt man sich über [6] die Quellen und übersetzt das Programm manuell.

Hat man sich die Quellen beschafft, entpackt man den Tarball, wechselt in das soeben enstandene Verzeichnis und führt make aus. Nachfolgend werden die dazu benötigten Befehle demonstriert:

```
neo5k@phobos> tar xzvf vsftpd–x.x.x.tar.gz
neo5k@phobos> cd vsftpd–x.x.x
neo5k@phobos> make
```

Zuvor sollte man jedoch überprüfen, ob der Benutzer "nobody" und das Verzeichnis "/usr/share/empty" existiert und gegebenenfalls neu anlegen. Plant man Zugriffsmöglichkeiten für anonyme Benutzer, muß der User "ftp" mitsamt Homeverzeichnis "/var/ftp" angelegt werden. Letzteres erreicht man durch die Eingabe der folgenden beiden Befehle:

```
neo5k@phobos> mkdir /var/ftp
neo5k@phobos> useradd –d /var/ftp ftp
```

Aus Sicherheitsgründen sollte das Verzeichnis "/var/ftp" dem Benutzer "ftp" weder gehören, noch sollte dieser darin Schreibrechte besitzen. Wenn der Benutzer bereits existiert, genügen die nächsten beiden Kommandos, um den Besitzer zu ändern und anderen Benutzern die Schreibrechte zu entziehen:

```
neo5k@phobos> chown root.root /var/ftp
neo5k@phobos> chmod og–w /var/ftp
```

Sofern alle Voraussetzungen erfüllt sind, kann man den vsftp–Daemon installieren:

```
neo5k@phobos> make install
```

Jetzt werden normalerweise die Manpages und das Programm an den richtigen Ort im Dateisystem kopiert. Wenn es wider Erwarten zu Komplikationen kommt, hilft jedoch auch ein manuelles Kopieren der Dateien.

```
neo5k@phobos> cp vsftpd /usr/sbin/vsftpd
neo5k@phobos> cp vsftpd.conf.5 /usr/share/man/man5
neo5k@phobos> cp vsftpd.8 /usr/share/man/man8
```

Da die Beispiel–Konfigurationsdatei nicht mit kopiert wird, diese aber den Einstieg erleichtert, muß man auch hier noch einmal Hand anlegen:

```
neo5k@phobos> cp vsftpd.conf /etc
```

Konfiguration

Die Konfigurationsdatei zu vsftpd läßt sich unter "/etc/vsftpd.conf" finden. Wie bei den meisten Konfigurationsdateien werden auch bei vsftpd Kommentare mit einer einleitenden Raute gekennzeichnet.

Kommentarzeile

Eine beispielhafte Konfiguration könnte so aussehen:

 # Anonymen FTP−Zugriff erlauben? YES/NO
anonymous_enable=NO

 # Anonymen Upload erlauben? YES/NO
anon_upload_enable=NO

 # Dürfen anonyme User Verzeichnisse erstellen? YES/NO
anon_mkdir_write_enable=NO

 # Dürfen anonyme User andere Schreiboperationen wie Umbenennen oder Löschen durchführen? YES/NO
anon_other_write_enable=NO

 # Anmeldung von lokalen Usern erlauben? YES/NO
local_enable=YES

 # Sollen lokale Benutzer in ihrem Homeverzeichnis eingesperrt werden? YES/NO
chroot_local_user=YES

 # Die maximal erlaubte Datentransferrate in Bytes/Sekunde für lokal angemeldete User. Vorgabe = 0 (unbegrenzt)
local_max_rate=7200

 # Schreibrechte prinzipiell erlauben? YES/NO
write_enable=YES

 # Nachrichten bei Verzeichniswechsel anzeigen? YES/NO
dirmessage_enable=YES

 # Bannermeldung, welche der sich anmeldende User sieht.
ftpd_banner="Welcome to neo5k's FTP service."

 # Protokollierung aktivieren? YES/NO
xferlog_enable=YES

 # Sämtliche FTP−Aktivitäten protokollieren? YES/NO
 # Achtung! Durch diesen Eintrag können sehr große Datenmengen entstehen.
log_ftp_protocol=NO

 # Versichern, daß Verbindungen nur an Port 20 (ftp−data) zustande kommen. YES/NO
connect_from_port_20=YES

 # Unterbrechung (time out) bei Leerlaufzeiten (idle sessions)
idle_session_timeout=600

```
# Zeit, nach der eine Datenverbindung unterbrochen wird.
data_connection_timeout=120

# Zugriff wird über Pluggable Authentication Modules (PAM) geregelt.
pam_service_name=vsftpd

# Standalone−Betrieb? YES/NO − abhängig vom Betriebsmodus (inetd, xinetd, Standalone)
# Des Autors FTP−Dienst wird per xinetd gestartet, deswegen lautet der Wert hier NO.
listen=NO
```

Starten des FTP−Dienstes

vsftpd läßt sich auf drei verschiedene Arten betreiben. Zum einen über inetd oder xinetd, zum anderen im Standalone−Betrieb.

inetd

Soll der FTP−Dienst via inetd betrieben werden, öffnet man die Konfigurationsdatei "/etc/inetd.conf" mit einem Editor:

```
neo5k@phobos> vi /etc/inetd.conf
```

Dann sucht man sich die entsprechenden Zeilen zu den FTP−Diensten und entfernt nur noch das Kommentarzeichen vor dem vsftpd−Eintrag. Sollte kein entsprechender Eintrag vorhanden sein, kann man ihn auch manuell erstellen. Dabei ist zu beachten, daß nach den durchgeführten Änderungen der inetd zwingend neu gestartet werden muß. Der Eintrag sollte dann so aussehen:

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

xinetd

Empfehlenswert ist es jedoch, den vsftp−Daemon per xinetd starten zu lassen, da dieser zahlreiche Erweiterungen gegenüber inetd besitzt. Einige davon sind bspw. Protokollierung von Anfragen, Zugriffssteuerung, Bindung des Dienstes an bestimmte Netzwerkschnittstellen, et cetera. Eine sehr gute Einführung zum Thema xinetd läßt sich unter [7] finden. Auch hier ist nach erfolgter Modifikation ein Neustart des xinetd nötig. Die Konfiguration des xinetd könnte folgendermaßen aussehen:

```
# vsftp daemon.
service ftp
{
    disable = no
    socket_type = stream
    wait = no
```

```
    user = root
    server = /usr/sbin/vsftpd
    per_source = 5
    instances = 200
    no_access = 192.168.1.3
    banner_fail = /etc/vsftpd.busy_banner
    log_on_success += PID HOST DURATION
    log_on_failure += HOST
    nice = 10
}
```

Standalone−Betrieb

Zusätzlich besteht die Möglichkeit, den vsftp−Daemon im Standalone−Modus zu betreiben. Dazu öffnet man wieder die Datei "/etc/vsftpd.conf" und führt die folgende Änderung durch:

 # Soll der vsftp−Daemon im Standalone−Betrieb laufen? YES/NO
listen=YES

Nach erfolgtem Eintrag kann man den Daemon dann durch die nachfolgend genannte Eingabe starten.

neo5k@phobos> /usr/sbin/vsftpd &

Sofern die Einstiegspfade richtig gesetzt sind, genügt zum Starten auch ein schlichtes

neo5k@phobos> vsftpd &

Durch die nächste Eingabe kann geprüft werden, ob die Einstiegspfade richtig gesetzt wurden:

neo5k@phobos> echo $PATH
/usr/sbin:/bin:/usr/bin:/sbin:/usr/X11R6/bin

Natürlich sollte man beim Betrieb im Standalone−Modus darauf achten, daß der vsftp−Daemon weder per inetd noch xinetd gestartet wird.

Funktionstest

Hat man die Installation und Konfiguration erfolgreich hinter sich gelassen, kann man das erste Mal auf seinen FTP−Server zugreifen.

```
neo5k@phobos> ftp phobos
Connected to phobos
220 "Welcome to neo5k's FTP service."
Name (phobos:neo5k): testuser
331 Please specify the password.
Password:
230 Login successful
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls -l
```

```
229 Entering Extended Passive Mode
150 Here comes the directory listing
drwxr-xr-x      11  500      100         400  May 07 16:22  docs
drwxr-xr-x       9  500       100         464  Feb   01 23:05  hlds
drwxr-xr-x      39  500      100        4168  May 10 09:15  projects
226 Directory send OK.
ftp>
```

Fazit

Wie man sehen konnte, ist der vsftp−Daemon weder schwer aufzusetzen noch schwierig zu konfigurieren. Trotzdem bietet er zahlreiche Funktionsmerkmale und ein hohes Maß an Sicherheit.

Es versteht sich von selbst, daß diese Einführung nur einen kleinen Ausschnitt aus der Welt von vsftpd bieten kann, da der FTP−Server äußerst umfangreiche Konfigurationsmöglichkeiten zur Verfügung stellt. Wer sich nach diesem Artikel eingehender mit vsftpd beschäftigen möchte, sollte die Projektseite unter [6] besuchen und sich dort die umfangreiche Dokumentation zu Gemüte führen.

Links

[1] ftp://ftp.rfc−editor.org/in−notes/rfc959.txt [RFC 959 − File Transfer Protocol]
[2] ftp://ftp.rfc−editor.org/in−notes/rfc2228.txt [RFC 2228 − FTP Security Extensions]
[3] www.linuxfocus.org article 225, January2002 [chroot]
[4] http://vsftpd.beasts.org/DESIGN [Sicherheitsmechanismen von vsftpd]
[5] http://www.wu−ftpd.org/ [WU−FTPD]
[6] http://www.vsftpd.beasts.org/ [Home of vsftpd]
[7] www.linuxfocus.org article 175, November2000 [xinetd]

Webpages maintained by the LinuxFocus Editor team © Mario M. Knopf "some rights reserved" see linuxfocus.org/license/ http://www.LinuxFocus.org	Translation information: de −−> −− : Mario M. Knopf <netzmeister/at/neo5k/dot/org>

2005−01−11, generated by lfparser_pdf version 2.51