Virtual Hosting With PureFTPd And MySQL (Incl. Quota And Bandwidth Management) On CentOS 6.4

Last edited 03/22/2013

This document describes how to install a PureFTPd server that uses virtual users from a MySQL database instead of real system users. This is much more performant and allows to have thousands of ftp users on a single machine. In addition to that I will show the use of quota and upload/download bandwidth limits with this setup. Passwords will be stored encrypted as MD5 strings in the database.

For the administration of the MySQL database you can use web based tools like phpMyAdmin which will also be installed in this howto. phpMyAdmin is a comfortable graphical interface which means you do not have to mess around with the command line.

This howto is meant as a practical guide; it does not cover the theoretical backgrounds. They are treated in a lot of other documents in the web.

This document comes without warranty of any kind! I want to say that this is not the only way of setting up such a system. There are many ways of achieving this goal but this is the way I take. I do not issue any guarantee that this will work for you!

1 Preliminary Note

In this tutorial I use the hostname server1.example.com with the IP address 192.168.0.100. These settings might differ for you, so you have to replace them where appropriate.

2 Install MySQL And phpMyAdmin

First we enable the EPEL repository on our CentOS system as some packages that we are going to install in the course of this tutorial are not available in the official CentOS 6.4 repositories:

```
rpm --import https://fedoraproject.org/static/0608B895.txt
wget http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
rpm -ivh epel-release-6-8.noarch.rpm
yum install yum-priorities
```

Edit /etc/yum.repos.d/epel.repo...

```
vi /etc/yum.repos.d/epel.repo
```

... and add the line priority=10 to the [epel] section:

```
[epel]
name=Extra Packages for Enterprise Linux 6 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch
failovermethod=priority
enabled=1
priority=10
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
[...]
```

Now we can install MySQL and phpMyAdmin as follows:

```
yum install mysql mysql-server phpMyAdmin httpd
```

Afterwards, we must edit the file /etc/httpd/conf.d/phpMyAdmin.conf to make phpMyAdmin accessible from all IP addresses. By default, it's accessible only from 127.0.0.1. Comment out this section:

```
#<Directory /usr/share/phpMyAdmin/>
#   Order Deny,Allow
#   Deny from All
#   Allow from 127.0.0.1
#   Allow from ::1
#</Directory>
```

so that the file looks like this:

```
vi /etc/httpd/conf.d/phpMyAdmin.conf
```

```
# phpMyAdmin - Web based MySQL browser written in php
#
# Allows only localhost by default
#
# But allowing phpMyAdmin to anyone other than localhost should be considered
# dangerous unless properly secured by SSL

Alias /phpMyAdmin /usr/share/phpMyAdmin
Alias /phpmyadmin /usr/share/phpMyAdmin

#<Directory /usr/share/phpMyAdmin/>
#   Order Deny,Allow
#   Deny from All
#   Allow from 127.0.0.1
#   Allow from ::1
#</Directory>

<Directory /usr/share/phpMyAdmin/setup/>
   Order Deny,Allow
   Deny from All
   Allow from 127.0.0.1
   Allow from ::1
</Directory>

# These directories do not require access over HTTP - taken from the original
# phpMyAdmin upstream tarball
#
<Directory /usr/share/phpMyAdmin/libraries/>
    Order Deny,Allow
    Deny from All
    Allow from None
</Directory>

<Directory /usr/share/phpMyAdmin/setup/lib/>
    Order Deny,Allow
    Deny from All
    Allow from None
</Directory>

<Directory /usr/share/phpMyAdmin/setup/frames/>
    Order Deny,Allow
    Deny from All
    Allow from None
</Directory>

# This configuration prevents mod_security at phpMyAdmin directories from
# filtering SQL etc.  This may break your mod_security implementation.
```

```
#
#<IfModule mod_security.c>
#    <Directory /usr/share/phpMyAdmin/>
#        SecRuleInheritance Off
#    </Directory>
#</IfModule>
```

Then we create the system startup links for MySQL and Apache (so that both start automatically whenever the system boots) and start both services:

```
chkconfig --levels 235 mysqld on
/etc/init.d/mysqld start
chkconfig --levels 235 httpd on
/etc/init.d/httpd start
```

Create a password for the MySQL user root (replace yourrootsqlpassword with the password you want to use):

```
mysql_secure_installation
```

```
[root@server1 ~]# mysql_secure_installation


NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user.  If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none): <-- ENTER
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

Set root password? [Y/n] <-- ENTER
New password: <-- yourrootsqlpassword
Re-enter new password: <-- yourrootsqlpassword
Password updated successfully!
Reloading privilege tables..
 ... Success!

By default, a MySQL installation has an anonymous user, allowing anyone
to log into MySQL without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] <-- ENTER
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] <-- ENTER
 ... Success!

By default, MySQL comes with a database named 'test' that anyone can
access.  This is also intended only for testing, and should be removed
before moving into a production environment.
```

```
Remove test database and access to it? [Y/n] <-- ENTER
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] <-- ENTER
 ... Success!

Cleaning up...

All done!  If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!

[root@server1 ~]#
```

3 Install PureFTPd With MySQL Support

The CentOS PureFTPd package supports various backends, such as MySQL, PostgreSQL, LDAP, etc. Therefore, all we have to do is install the normal PureFTPd package:

```
yum install pure-ftpd
```

Then we create an ftp group (**ftpgroup**) and user (**ftpuser**) that all our virtual users will be mapped to. Replace the group- and userid 2001 with a number that is free on your system:

```
groupadd -g 2001 ftpgroup
useradd -u 2001 -s /bin/false -d /bin/null -c "pureftpd user" -g ftpgroup ftpuser
```

4 Create The MySQL Database For PureFTPd

Now we create a database called pureftpd and a MySQL user named pureftpd which the PureFTPd daemon will use later on to connect to the pureftpd database:

```
mysql -u root -p
     CREATE DATABASE pureftpd;

     GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP ON pureftpd.*
     TO 'pureftpd'@'localhost' IDENTIFIED BY 'ftpdpass';

     GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP ON pureftpd.*
     TO 'pureftpd'@'localhost.localdomain' IDENTIFIED BY 'ftpdpass';
FLUSH PRIVILEGES;
```

Replace the string ftpdpass with whatever password you want to use for the MySQL user pureftpd. Still on the MySQL shell, we create the database table we need (yes, there is only one table!):

```
USE pureftpd;
CREATE TABLE `ftpd` (
`User` VARCHAR(16) NOT NULL DEFAULT '' COLLATE 'utf8_unicode_ci',
`status` ENUM('0','1') NOT NULL DEFAULT '0' COLLATE 'utf8_unicode_ci',
`Password` VARCHAR(64) NOT NULL DEFAULT '' COLLATE 'utf8_unicode_ci',
`Uid` VARCHAR(11) NOT NULL DEFAULT '-1' COLLATE 'utf8_unicode_ci',
`Gid` VARCHAR(11) NOT NULL DEFAULT '-1' COLLATE 'utf8_unicode_ci',
`Dir` VARCHAR(128) NOT NULL DEFAULT '' COLLATE 'utf8_unicode_ci',
`ULBandwidth` SMALLINT(5) NOT NULL DEFAULT '0',
`DLBandwidth` SMALLINT(5) NOT NULL DEFAULT '0',
`comment` TINYTEXT NOT NULL COLLATE 'utf8_unicode_ci',
`ipaccess` VARCHAR(15) NOT NULL DEFAULT '*' COLLATE 'utf8_unicode_ci',
`QuotaSize` BIGINT(10) NOT NULL DEFAULT '0',
`QuotaFiles` INT(11) NOT NULL DEFAULT '0',
PRIMARY KEY (`User`),
UNIQUE INDEX `User` (`User`)
)
COMMENT='10240000'
COLLATE='utf8_unicode_ci'
ENGINE=MyISAM
```

As you may have noticed, with the quit; command we have left the MySQL shell and are back on the Linux shell.

BTW, (I'm assuming that the hostname of your ftp server system is server1.example.com) you can access phpMyAdmin under http://server1.example.com/phpMyAdmin/ (you can also use the IP address instead of server1.example.com) in a browser and log in as the user pureftpd. Then you can have a look at the database. Later on you can use phpMyAdmin to administrate your PureFTPd server.

5 Configure PureFTPd

Edit /etc/pure-ftpd/pure-ftpd.conf and make sure that the ChrootEveryone, MySQLConfigFile, and CreateHomeDir lines are enabled and look like this:

```
vi /etc/pure-ftpd/pure-ftpd.conf
```

```
[...]
ChrootEveryone              yes #Không cho user nhìn ngoài thư mục của mình
[...]
MySQLConfigFile               /etc/pure-ftpd/pureftpd-mysql.conf
[...]
CreateHomeDir               yes #Nếu chưa có folder home của user thì tạo ra
[...]
```

The ChrootEveryone setting will make PureFTPd chroot every virtual user in his home directory so he will not be able to browse directories and files outside his home directory. The CreateHomeDir line will make PureFTPd create a user's home directory when the user logs in and the home directory does not exist yet.

Then we edit /etc/pure-ftpd/pureftpd-mysql.conf. It should look like this:

```
cp /etc/pure-ftpd/pureftpd-mysql.conf /etc/pure-ftpd/pureftpd-mysql.conf_orig
cat /dev/null > /etc/pure-ftpd/pureftpd-mysql.conf
vi /etc/pure-ftpd/pureftpd-mysql.conf
```

```
####################################
#/etc/pure-ftpd/pureftpd-mysql.conf#
####################################
MYSQLSocket     /opt/lampp/var/mysql/mysql.sock
#MYSQLServer     localhost
#MYSQLPort       3306
MYSQLDatabase    pureftpd

#OLD VALUE:
MYSQLUser        pureftpd
MYSQLPassword    ftpdpass

#MYSQLCrypt md5, cleartext, crypt() or password() - md5 is VERY RECOMMENDABLE
uppon cleartext
```

```
MYSQLCrypt             md5
MYSQLGetPW             SELECT Password        FROM ftpd WHERE User="\L"
        AND status="1" AND (ipaccess = "*" OR ipaccess LIKE "\R")
MYSQLGetUID            SELECT Uid                     FROM ftpd WHERE User="\L"
        AND status="1" AND (ipaccess = "*" OR ipaccess LIKE "\R")
MYSQLGetGID            SELECT Gid                     FROM ftpd WHERE User="\L"
MYSQLGetDir            SELECT Dir                     FROM ftpd WHERE User="\L"
MySQLGetBandwidthUL SELECT ULBandwidth        FROM ftpd WHERE User="\L"
        AND status="1" AND (ipaccess = "*" OR ipaccess LIKE "\R")
MySQLGetBandwidthDL SELECT DLBandwidth        FROM ftpd WHERE User="\L"
        AND status="1" AND (ipaccess = "*" OR ipaccess LIKE "\R")
MySQLGetQTASZ          SELECT QuotaSize       FROM ftpd WHERE User="\L"
        AND status="1" AND (ipaccess = "*" OR ipaccess LIKE "\R")
MySQLGetQTAFS          SELECT QuotaFiles      FROM ftpd WHERE User="\L"
        AND status="1" AND (ipaccess = "*" OR ipaccess LIKE "\R")
```

Make sure that you replace the string ftpdpass with the real password for the MySQL user pureftpd in the line MYSQLPassword! Please note that we use md5 as MYSQLCrypt method, which means we will store the users' passwords as an MD5 string in the database which is far more secure than using plain text passwords!

Now we create the system startup links for PureFTPd and start it:

```
chkconfig --levels 235 pure-ftpd on
/etc/init.d/pure-ftpd start
```

6 Populate The Database And Test

To populate the database you can use the MySQL shell:

```
mysql -u root -p
USE pureftpd;
```

Now we create the user exampleuser with the status 1 (which means his ftp account is active), the password secret (which will be stored encrypted using MySQL's MD5 function), the UID and GID 2001 (use the userid and groupid of the user/group you created at the end of step two!), the home directory /home/www.example.com, an upload and download bandwidth of 100 KB/sec. (kilobytes per second), and a quota of 50 MB:

```
INSERT INTO `ftpd` (`User`, `status`, `Password`, `Uid`, `Gid`, `Dir`, `ULBandwidth`,
`DLBandwidth`, `comment`, `ipaccess`, `QuotaSize`, `QuotaFiles`) VALUES ('exampleuser',
'1', MD5('secret'), '2001', '2001', '/home/www.example.com', '100', '100', '', '*',
'50', '0');
quit;
```

Now open your FTP client program on your work station (something like WS_FTP or SmartFTP if you are on a Windows system or gFTP on a Linux desktop) and try to connect. As hostname you use server1.example.com (or the IP address of the system), the username is exampleuser, and the password is secret.

If you are able to connect - congratulations! If not, something went wrong.

Now, if you run

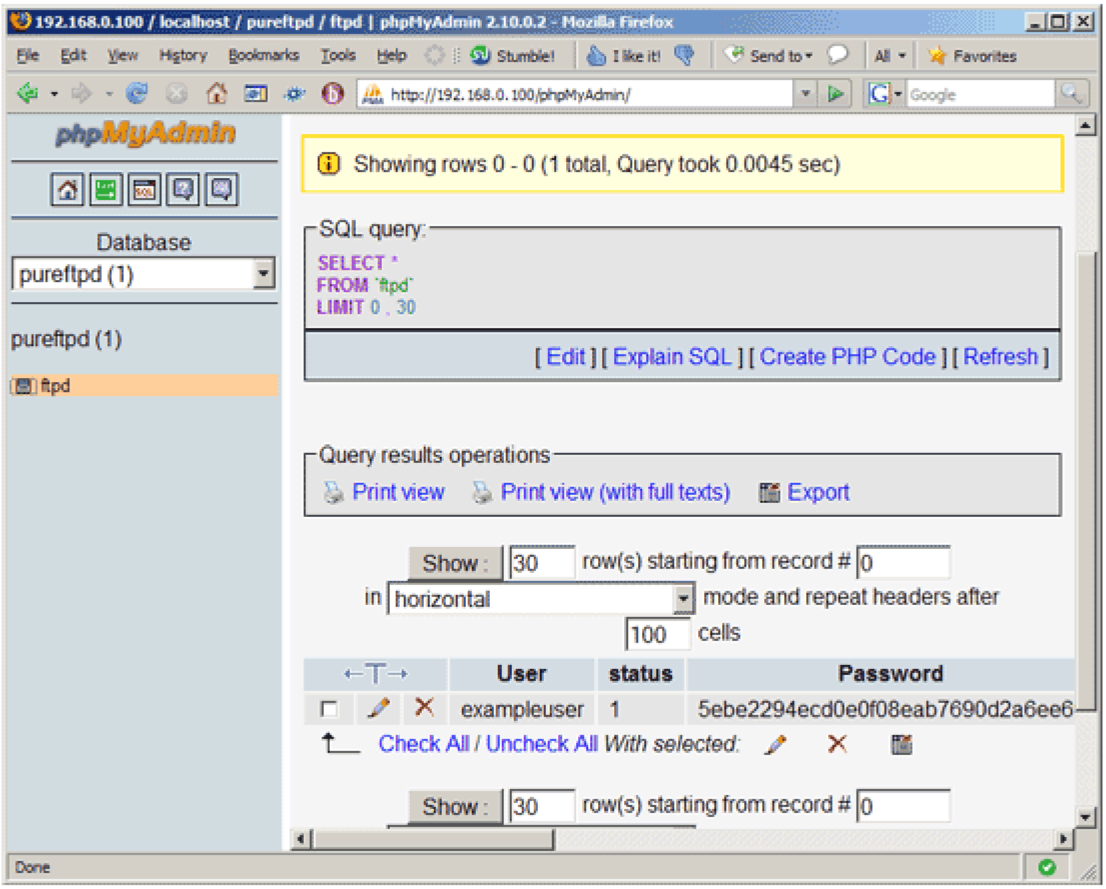
```
ls -l /home
```

you should see that the directory /home/www.example.com (exampleuser's home directory) has been created automatically, and it is owned by ftpuser and ftpgroup (the user/group we created at the end of step two):

```
[root@server1 ~]# ls -l /home
total 4
drwxr-xr-x 2 ftpuser ftpgroup 4096 Mar  5 02:13 www.example.com
[root@server1 ~]#
```

7 Database Administration

For most people it is easier if they have a graphical front-end to MySQL; therefore you can also use phpMyAdmin (in this example under **http://server1.example.com/phpMyAdmin/**) to administrate the pureftpd database.



192.168.0.100 / localhost / pureftpd / ftpd | phpMyAdmin 2.10.0.2 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Stumble! I like it! Send to All Favorites
http://192.168.0.100/phpMyAdmin/
Google
phpMyAdmin
Database
pureftpd (1)
pureftpd (1)
ftpd
Done

Showing rows 0 - 0 (1 total, Query took 0.0045 sec)

SQL query:

```
SELECT *
FROM `ftpd`
LIMIT 0 , 30
```

[Edit] [Explain SQL] [Create PHP Code] [Refresh]

Query results operations

Print view Print view (with full texts) Export

Show : 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

←T→	User	status	Password
□	exampleuser	1	5ebe2294ecd0e0f08eab7690d2a6ee6

Check All / Uncheck All *With selected:*

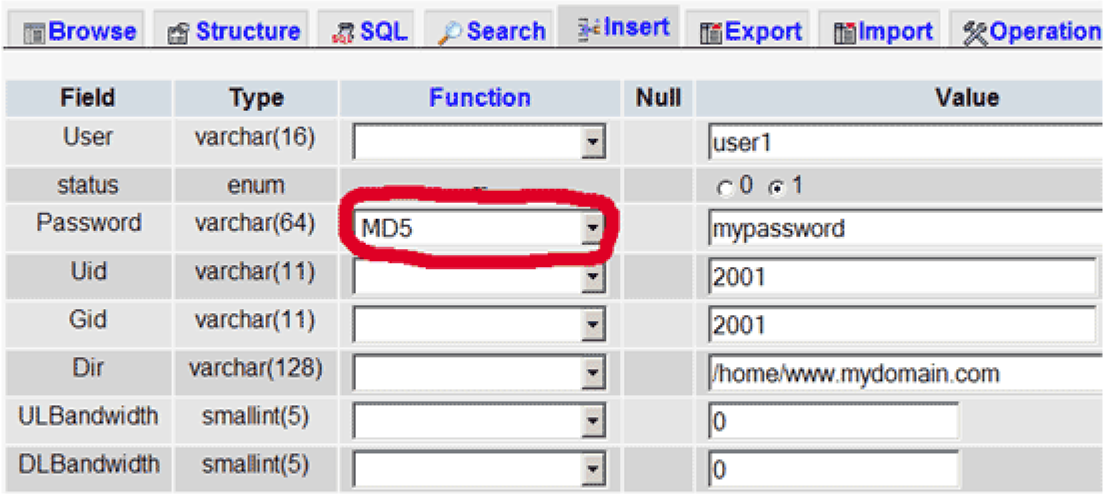
Show : 30 row(s) starting from record # 0

Whenever you want to create a new user, you have to create an entry in the table ftpd so I will explain the columns of this table here:

ftpd Table:

- **User:** The name of the virtual PureFTPd user (e.g. exampleuser).
- **status:** 0 or 1. 0 means the account is disabled, the user cannot login.

- **Password**: The password of the virtual user. Make sure you use MySQL's MD5 function to save the password encrypted as an MD5 string:



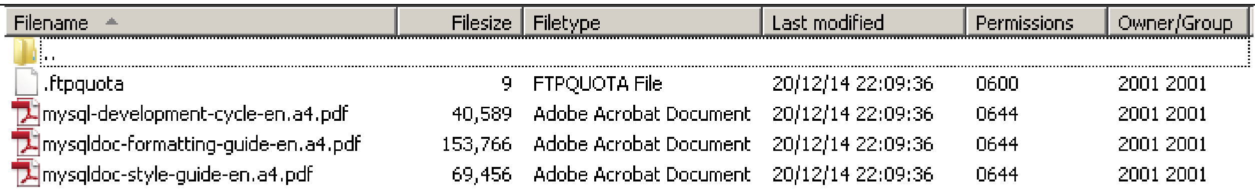
Field	Type	Function	Null	Value
User	varchar(16)			user1
status	enum			○ 0 ● 1
Password	varchar(64)	MD5		mypassword
Uid	varchar(11)			2001
Gid	varchar(11)			2001
Dir	varchar(128)			/home/www.mydomain.com
ULBandwidth	smallint(5)			0
DLBandwidth	smallint(5)			0

- **UID**: The userid of the ftp user you created at the end of step two (e.g. 2001).
- **GID**: The groupid of the ftp group you created at the end of step two (e.g. 2001).
- **Dir**: The home directory of the virtual PureFTPd user (e.g. /home/www.example.com). If it does not exist, it will be created when the new user logs in the first time via FTP. The virtual user will be jailed into this home directory, i.e., he cannot access other directories outside his home directory.
- **ULBandwidth**: Upload bandwidth of the virtual user in KB/sec. (kilobytes per second). 0 means unlimited.
- **DLBandwidth**: Download bandwidth of the virtual user in KB/sec. (kilobytes per second). 0 means unlimited.
- **comment**: You can enter any comment here (e.g. for your internal administration) here. Normally you leave this field empty.
- **ipaccess**: Enter IP addresses here that are allowed to connect to this FTP account. * means any IP address is allowed to connect.
- **QuotaSize**: Storage space in MB (not KB, as in ULBandwidth and DLBandwidth!) the virtual user is allowed to use on the FTP server. 0 means unlimited.
- **QuotaFiles**: amount of files the virtual user is allowed to save on the FTP server. 0 means unlimited.

```
-- user: manga247_com
-- status: 1-enable | 0-disable
-- password: md5('<string>')
-- uid: 2001
-- gid: 2001
-- dir: /opt/xampp_www/manga247.com
-- ULBandwidth: 1024 KB/sec ~ 1 MB/s | 0~unlimited
-- DLBandwidth: 1024 KB/sec ~ 1 MB/s | 0~unlimited
-- comment: ftp user manga247_com
-- ipaccess: *-all | 192.168.1.0/24 | ...
-- QuotaSize: 4096 MB
-- QuotaFiles: 0 means unlimited.
INSERT INTO `ftpd` (`User`, `status`, `Password`, `Uid`, `Gid`, `Dir`,
`ULBandwidth`, `DLBandwidth`, `comment`, `ipaccess`, `QuotaSize`, `QuotaFiles`)
VALUES ('exampleuser', '1', MD5('secret'), '2001', '2001', '/home/www.example.com',
'1024', '1024', 'ftp user exampleuser', '*', '1024', '0');
#




Bản chất là quyền 644 trên thư mục.
```



Filename	Filesize	Filetype	Last modified	Permissions	Owner/Group
..					
.ftpquota	9	FTPQUOTA File	20/12/14 22:09:36	0600	2001 2001
mysql-development-cycle-en.a4.pdf	40,589	Adobe Acrobat Document	20/12/14 22:09:36	0644	2001 2001
mysqldoc-formatting-guide-en.a4.pdf	153,766	Adobe Acrobat Document	20/12/14 22:09:36	0644	2001 2001
mysqldoc-style-guide-en.a4.pdf	69,456	Adobe Acrobat Document	20/12/14 22:09:36	0644	2001 2001

```
#
```

```
#############################################################
#         Configuration file for pure-ftpd wrappers         #
#         /etc/pure-ftpd/pure-ftpd.conf                     #
#############################################################
# If you want to run Pure-FTPd with this configuration
# instead of command-line options, please run the
# following command :
#
# /usr/sbin/pure-config.pl /etc/pure-ftpd/pure-ftpd.conf
#
# Please don't forget to have a look at documentation at
# http://www.pureftpd.org/documentation.shtml for a complete list of
# options.

# Cage in every user in his home directory
ChrootEveryone              yes

# If the previous option is set to "no", members of the following group
# won't be caged. Others will be. If you don't want chroot()ing anyone,
# just comment out ChrootEveryone and TrustedGID.
# TrustedGID                    100

# Turn on compatibility hacks for broken clients
BrokenClientsCompatibility  no

# Maximum number of simultaneous users
MaxClientsNumber            50

# Fork in background
Daemonize                   yes

# Maximum number of sim clients with the same IP address
MaxClientsPerIP             8

# If you want to log all client commands, set this to "yes".
# This directive can be duplicated to also log server responses.
VerboseLog                  no

# List dot-files even when the client doesn't send "-a".
DisplayDotFiles             yes

# Don't allow authenticated users - have a public anonymous FTP only.
AnonymousOnly               no

# Disallow anonymous connections. Only allow authenticated users.
NoAnonymous                 no

# Syslog facility (auth, authpriv, daemon, ftp, security, user, local*)
# The default facility is "ftp". "none" disables logging.
SyslogFacility              ftp

# Display fortune cookies
# FortunesFile              /usr/share/fortune/zippy

# Don't resolve host names in log files. Logs are less verbose, but
# it uses less bandwidth. Set this to "yes" on very busy servers or
# if you don't have a working DNS.
DontResolve                 yes

# Maximum idle time in minutes (default = 15 minutes)
MaxIdleTime                 15

# LDAP configuration file (see README.LDAP)
# LDAPConfigFile                /etc/pure-ftpd/pureftpd-ldap.conf

# MySQL configuration file (see README.MySQL)
### MySQLConfigFile               /etc/pure-ftpd/pureftpd-mysql.conf
MySQLConfigFile               /etc/pure-ftpd/pureftpd-mysql.conf

# Postgres configuration file (see README.PGSQL)
# PGSQLConfigFile               /etc/pure-ftpd/pureftpd-pgsql.conf

# PureDB user database (see README.Virtual-Users)
# PureDB                        /etc/pure-ftpd/pureftpd.pdb

# Path to pure-authd socket (see README.Authentication-Modules)
# ExtAuth                       /var/run/ftpd.sock

# If you want to enable PAM authentication, uncomment the following line
PAMAuthentication             yes

# If you want simple Unix (/etc/passwd) authentication, uncomment this
# UnixAuthentication            yes

# Please note that LDAPConfigFile, MySQLConfigFile, PAMAuthentication and
# UnixAuthentication can be used only once, but they can be combined
# together. For instance, if you use MySQLConfigFile, then UnixAuthentication,
# the SQL server will be asked. If the SQL authentication fails because the
# user wasn't found, another try # will be done with /etc/passwd and
# /etc/shadow. If the SQL authentication fails because the password was wrong,
# the authentication chain stops here. Authentication methods are chained in
# the order they are given.

# 'ls' recursion limits. The first argument is the maximum number of
# files to be displayed. The second one is the max subdirectories depth
LimitRecursion              10000 8

# Are anonymous users allowed to create new directories ?
AnonymousCanCreateDirs      no

# If the system is more loaded than the following value,
# anonymous users aren't allowed to download.
MaxLoad                     4

# Port range for passive connections replies. - for firewalling.
# PassivePortRange          30000 50000

# Force an IP address in PASV/EPSV/SPSV replies. - for NAT.
# Symbolic host names are also accepted for gateways with dynamic IP
# addresses.
# ForcePassiveIP                192.168.0.1

# Upload/download ratio for anonymous users.
# AnonymousRatio                1 10

# Upload/download ratio for all users.
# This directive superscedes the previous one.
# UserRatio                 1 10
# Disallow downloading of files owned by "ftp", ie.
# files that were uploaded but not validated by a local admin.
AntiWarez                    yes

# IP address/port to listen to (default=all IP and port 21).
# Bind                       127.0.0.1,21

# Maximum bandwidth for anonymous users in KB/s
# AnonymousBandwidth             8

# Maximum bandwidth for *all* users (including anonymous) in KB/s
# Use AnonymousBandwidth *or* UserBandwidth, both makes no sense.
# UserBandwidth              8

# File creation mask. <umask for files>:<umask for dirs> .
# 177:077 if you feel paranoid.
Umask                        133:022

# Minimum UID for an authenticated user to log in.
MinUID                       500

# Do not use the /etc/ftpusers file to disable accounts. We're already
# using MinUID to block users with uid < 500
UseFtpUsers no

# Allow FXP transfers for authenticated users.
AllowUserFXP                 no

# Allow anonymous FXP for anonymous and non-anonymous users.
AllowAnonymousFXP            no

# Users can't delete/write files beginning with a dot ('.')
# even if they own them. If TrustedGID is enabled, this group
# will have access to dot-files, though.
ProhibitDotFilesWrite        no

# Prohibit *reading* of files beginning with a dot (.history, .ssh...)
ProhibitDotFilesRead         no

# Never overwrite files. When a file whoose name already exist is uploaded,
# it get automatically renamed to file.1, file.2, file.3, ...
AutoRename                   no

# Disallow anonymous users to upload new files (no = upload is allowed)
AnonymousCantUpload          yes

# Only connections to this specific IP address are allowed to be
# non-anonymous. You can use this directive to open several public IPs for
# anonymous FTP, and keep a private firewalled IP for remote administration.
# You can also only allow a non-routable local IP (like 10.x.x.x) to
# authenticate, and keep a public anon-only FTP server on another IP.
#TrustedIP                   10.1.1.1

# If you want to add the PID to every logged line, uncomment the following
# line.
#LogPID                      yes

# Create an additional log file with transfers logged in a Apache-like format :
# fw.c9x.org - jedi [13/Dec/1975:19:36:39] "GET /ftp/linux.tar.bz2" 200 21809338
# This log file can then be processed by www traffic analyzers.
AltLog                      clf:/var/log/pureftpd.log

# Create an additional log file with transfers logged in a format optimized
# for statistic reports.
# AltLog                     stats:/var/log/pureftpd.log

# Create an additional log file with transfers logged in the standard W3C
# format (compatible with most commercial log analyzers)
# AltLog                     w3c:/var/log/pureftpd.log

# Disallow the CHMOD command. Users can't change perms of their files.
#NoChmod                     yes

# Allow users to resume and upload files, but *NOT* to delete them.
#KeepAllFiles                yes



# Automatically create home directories if they are missing
###CreateHomeDir               yes
CreateHomeDir               yes

# Enable virtual quotas. The first number is the max number of files.
# The second number is the max size of megabytes.
# So 1000:10 limits every user to 1000 files and 10 Mb.
#Quota                       1000:10

# If your pure-ftpd has been compiled with standalone support, you can change
# the location of the pid file. The default is /var/run/pure-ftpd.pid
#PIDFile                     /var/run/pure-ftpd.pid

# If your pure-ftpd has been compiled with pure-uploadscript support,
# this will make pure-ftpd write info about new uploads to
# /var/run/pure-ftpd.upload.pipe so pure-uploadscript can read it and
# spawn a script to handle the upload.
# Don't enable this option if you don't actually use pure-uploadscript.
#CallUploadScript yes

# This option is useful with servers where anonymous upload is
# allowed. As /var/ftp is in /var, it save some space and protect
# the log files. When the partition is more that X percent full,
# new uploads are disallowed.
MaxDiskUsage               99

# Set to 'yes' if you don't want your users to rename files.
#NoRename                  yes

# Be 'customer proof' : workaround against common customer mistakes like
# 'chmod 0 public_html', that are valid, but that could cause ignorant
# customers to lock their files, and then keep your technical support busy
# with silly issues. If you're sure all your users have some basic Unix
# knowledge, this feature is useless. If you're a hosting service, enable it.
CustomerProof              yes

# Per-user concurrency limits. It will only work if the FTP server has
# been compiled with --with-peruserlimits (and this is the case on
# most binary distributions) .
# The format is : <max sessions per user>:<max anonymous sessions>
# For instance, 3:20 means that the same authenticated user can have 3 active
# sessions max. And there are 20 anonymous sessions max.
# PerUserLimits            3:20

# When a file is uploaded and there is already a previous version of the file
# with the same name, the old file will neither get removed nor truncated.
# Upload will take place in a temporary file and once the upload is complete,
# the switch to the new version will be atomic. For instance, when a large PHP
# script is being uploaded, the web server will still serve the old version and
# immediatly switch to the new one as soon as the full file will have been
# transfered. This option is incompatible with virtual quotas.
# NoTruncate                 yes

# This option can accept three values :
# 0 : disable SSL/TLS encryption layer (default).
# 1 : accept both traditional and encrypted sessions.
# 2 : refuse connections that don't use SSL/TLS security mechanisms,
#     including anonymous sessions.
# Do _not_ uncomment this blindly. Be sure that :
# 1) Your server has been compiled with SSL/TLS support (--with-tls),
# 2) A valid certificate is in place,
# 3) Only compatible clients will log in.
# TLS                        1

# Listen only to IPv4 addresses in standalone mode (ie. disable IPv6)
# By default, both IPv4 and IPv6 are enabled.
# IPV4Only                   yes

# Listen only to IPv6 addresses in standalone mode (ie. disable IPv4)
# By default, both IPv4 and IPv6 are enabled.
# IPV6Only                   yes

# UTF-8 support for file names (RFC 2640)
# Define charset of the server filesystem and optionnally the default charset
# for remote clients if they don't use UTF-8.
# Works only if pure-ftpd has been compiled with --with-rfc2640

# FileSystemCharset     big5
# ClientCharset         big5
##############################################################END
```

8 Anonymous FTP <<Bỏ phần này

If you want to create an anonymous ftp account (an ftp account that everybody can login to without a password), you need a user and a group called ftp. Both have been created automatically when you installed the pure-ftpd package, so you don't need to create them manually. However, ftp's homedir is /var/ftp by default, but I'd like to create the anonymous ftp directory in /home/ftp (the normal users' ftp directories are in /home as well, e.g. /home/www.example.com). But of course, you can use the /var/ftp directory for anonymous ftp, if you prefer it.

If you want to use /home/ftp, open /etc/passwd and change the ftp user's homedir from /var/ftp to /home/ftp (don't do this if you want to use /var/ftp):

```
vi /etc/passwd
```

```
[...]
#ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
ftp:x:14:50:FTP User:/home/ftp:/sbin/nologin
[...]
```

Then move /var/ftp to /home (don't do this if you want to use /var/ftp):

```
mv /var/ftp /home
```

Then we create the directory /home/ftp/incoming which will allow anonymous users to upload files. We will give the /home/ftp/incoming directory permissions of 311 so that users can upload, but not see or download any files in that directory. The /home/ftp directory will have permissions of 555 which allows seeing and downloading of files:

```
chown ftp:nobody /home/ftp
cd /home/ftp
mkdir incoming
chown ftp:nobody incoming/
chmod 311 incoming/
cd ../
chmod 555 ftp/
```

(If you want to use /var/ftp instead, replace /home/ftp with /var/ftp in the above commands.)

Anonymous users will be able to log in, and they will be allowed to download files from /home/ftp, but uploads will be limited to /home/ftp/incoming (and once a file is uploaded into /home/ftp/incoming, it cannot be read nor downloaded from there; the server admin has to move it into /home/ftp first to make it available to others).

Now we have to configure PureFTPd for anonymous ftp. Open /etc/pure-ftpd/pure-ftpd.conf and make sure that you have the following settings in it:

```
vi /etc/pure-ftpd/pure-ftpd.conf
```

```
[...]
NoAnonymous                 no
[...]
AntiWarez                   no
[...]
AnonymousBandwidth            8
[...]
AnonymousCantUpload         no
[...]
```

(The AnonymousBandwidth setting is optional - it allows you to limit upload and download bandwidths for anonymous users. 8 means 8 KB/sec. Use any value you like, or comment out the line if you don't want to limit bandwidths.)

Finally, we restart PureFTPd:

```
/etc/init.d/pure-ftpd restart
```

9 Links

- PureFTPd: http://www.pureftpd.org/
- MySQL: http://www.mysql.com/
- phpMyAdmin: http://www.phpmyadmin.net/
- CentOS: http://centos.org/

How To Configure PureFTPd To Accept TLS Sessions

```
apt-get install pure-ftpd-mysql
http://www.howtoforge.com/virtual-hosting-with-pureftpd-and-mysql-incl-quota-and-bandwidth-management-on-ubuntu-14.04-lts
groupadd -g 2001 ftpgroup
useradd -u 2001 -s /bin/false -d /bin/null -c "pureftpd user" -g ftpgroup ftpuser
#
#If you want to accept TLS sessions only (no FTP), run
#
#echo 2 > /etc/pure-ftpd/conf/TLS
############################################################
#         Configuration file for pure-ftpd wrappers        #
#         /etc/pure-ftpd/pure-ftpd.conf                    #
############################################################
# This option can accept three values :
# 0 : disable SSL/TLS encryption layer (default).
# 1 : accept both traditional and encrypted sessions.
# 2 : refuse connections that don't use SSL/TLS security mechanisms,
#     including anonymous sessions.
# Do _not_ uncomment this blindly. Be sure that :
# 1) Your server has been compiled with SSL/TLS support (--with-tls),
# 2) A valid certificate is in place,
# 3) Only compatible clients will log in.
### TLS                        1
TLS                        2
```

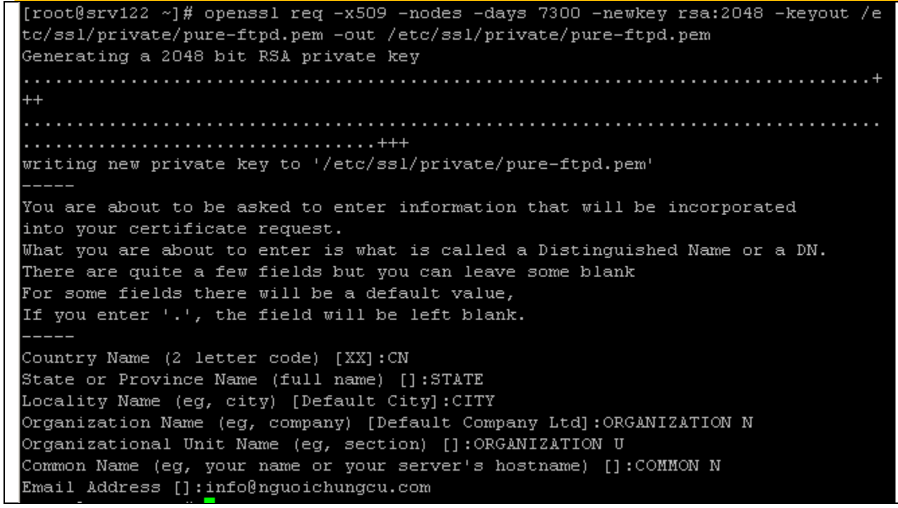
Creating The SSL Certificate For TLS

In order to use TLS, we must create an SSL certificate. I create it in /etc/ssl/private/, therefore I create that directory first:

```
mkdir -p /etc/ssl/private/
```

Afterwards, we can generate the SSL certificate as follows:

```
openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /etc/ssl/private/pure-ftpd.pem -out /etc/ssl/private/pure-ftpd.pem
```



```
[root@srv122 ~]# openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout /e
tc/ssl/private/pure-ftpd.pem -out /etc/ssl/private/pure-ftpd.pem
Generating a 2048 bit RSA private key
...............................................................................+
++
................................................................................
................................+++
writing new private key to '/etc/ssl/private/pure-ftpd.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:CN
State or Province Name (full name) []:STATE
Locality Name (eg, city) [Default City]:CITY
Organization Name (eg, company) [Default Company Ltd]:ORGANIZATION N
Organizational Unit Name (eg, section) []:ORGANIZATION U
Common Name (eg, your name or your server's hostname) []:COMMON N
Email Address []:info@nguoichungcu.com
```

Change the permissions of the SSL certificate:

```
chmod 600 /etc/ssl/private/pure-ftpd.pem
```

Finally restart PureFTPd:

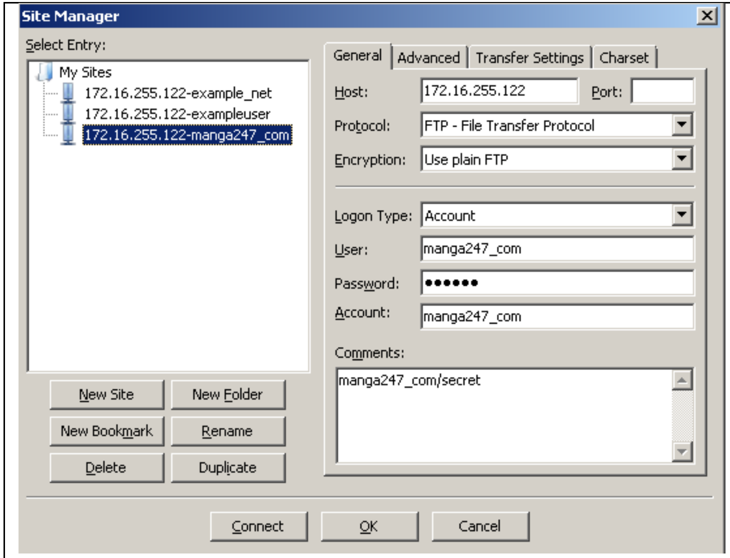
```
[root@srv122 ~]# /etc/init.d/pure-ftpd restart
```

That's it. You can now try to connect using your FTP client; however, you should configure your FTP client to use TLS - see the next chapter how to do this with FileZilla.

Configuring FileZilla For TLS

In order to use FTP with TLS, you need an FTP client that supports TLS, such as FileZilla.

In FileZilla, open the Server Manager:



My Sites
- 172.16.255.122-example_net
- 172.16.255.122-exampleuser
- 172.16.255.122-manga247_com

Host: 172.16.255.122 Port:

Protocol: FTP - File Transfer Protocol

Encryption: Use plain FTP

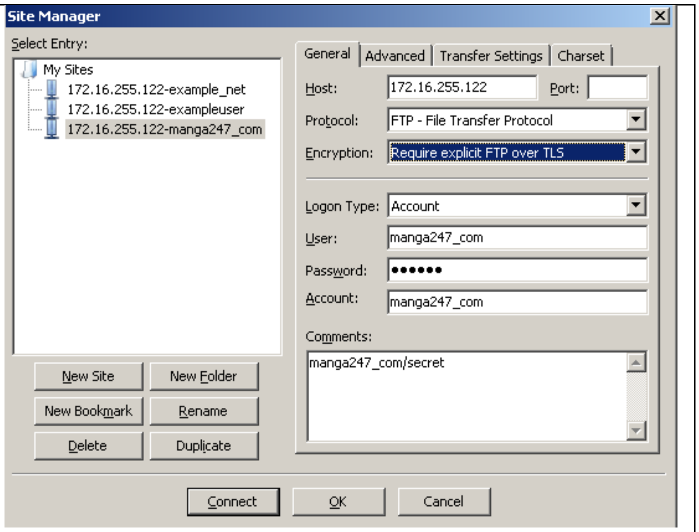
Logon Type: Account

User: manga247_com

Password: ••••••

Account: manga247_com

Comments:
manga247_com/secret



My Sites
- 172.16.255.122-example_net
- 172.16.255.122-exampleuser
- 172.16.255.122-manga247_com

Host: 172.16.255.122 Port:

Protocol: FTP - File Transfer Protocol

Encryption: Require explicit FTP over TLS

Logon Type: Account

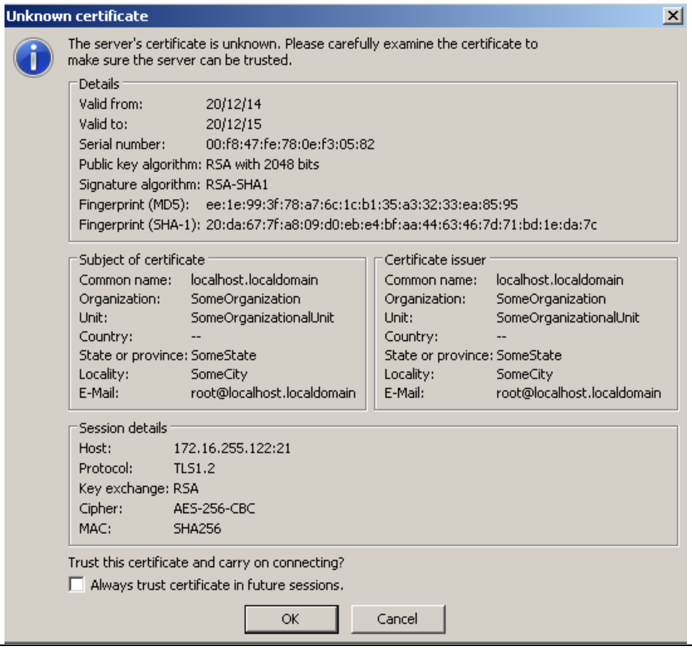
User: manga247_com

Password: ••••••

Account: manga247_com

Comments:
manga247_com/secret

#



The server's certificate is unknown. Please carefully examine the certificate to make sure the server can be trusted.

Details

Valid from:	20/12/14
Valid to:	20/12/15
Serial number:	00:f8:47:fe:78:0e:f3:05:82
Public key algorithm:	RSA with 2048 bits
Signature algorithm:	RSA-SHA1
Fingerprint (MD5):	ee:1e:99:3f:78:a7:6c:1c:b1:35:a3:32:33:ea:85:95
Fingerprint (SHA-1):	20:da:67:7f:a8:09:d0:eb:e4:bf:aa:44:63:46:7d:71:bd:1e:da:7c

Subject of certificate

Common name:	localhost.localdomain
Organization:	SomeOrganization
Unit:	SomeOrganizationalUnit
Country:	--
State or province:	SomeState
Locality:	SomeCity
E-Mail:	root@localhost.localdomain

Certificate issuer

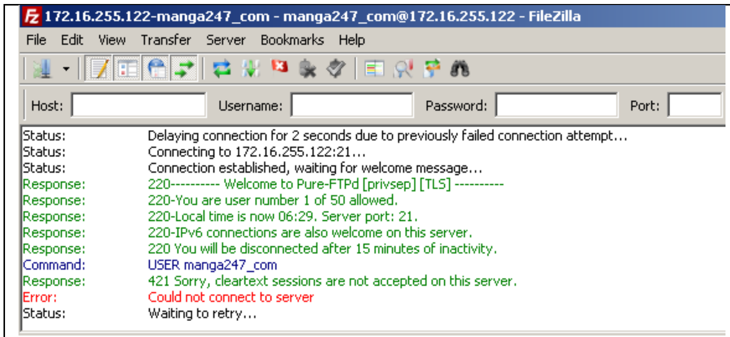
Common name:	localhost.localdomain
Organization:	SomeOrganization
Unit:	SomeOrganizationalUnit
Country:	--
State or province:	SomeState
Locality:	SomeCity
E-Mail:	root@localhost.localdomain

Session details

Host:	172.16.255.122:21
Protocol:	TLS1.2
Key exchange:	RSA
Cipher:	AES-256-CBC
MAC:	SHA256

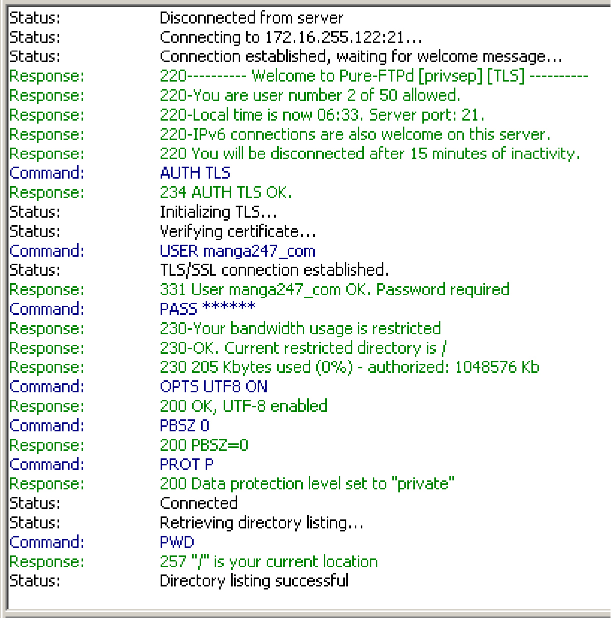
Trust this certificate and carry on connecting?

□ Always trust certificate in future sessions.


172.16.255.122-manga247_com - manga247_com@172.16.255.122 - FileZilla
File Edit View Transfer Server Bookmarks Help
Host:
Username:
Password:
Port:

```
Status:      Delaying connection for 2 seconds due to previously failed connection attempt...
Status:      Connecting to 172.16.255.122:21...
Status:      Connection established, waiting for welcome message...
Response:    220---------- Welcome to Pure-FTPd [privsep] [TLS] ----------
Response:    220-You are user number 1 of 50 allowed.
Response:    220-Local time is now 06:29. Server port: 21.
Response:    220-IPv6 connections are also welcome on this server.
Response:    220 You will be disconnected after 15 minutes of inactivity.
Command:     USER manga247_com
Response:    421 Sorry, cleartext sessions are not accepted on this server.
Error:       Could not connect to server
Status:      Waiting to retry...
```

Không enable TLS connection: ERROR



```
Status:      Disconnected from server
Status:      Connecting to 172.16.255.122:21...
Status:      Connection established, waiting for welcome message...
Response:    220---------- Welcome to Pure-FTPd [privsep] [TLS] ----------
Response:    220-You are user number 2 of 50 allowed.
Response:    220-Local time is now 06:33. Server port: 21.
Response:    220-IPv6 connections are also welcome on this server.
Response:    220 You will be disconnected after 15 minutes of inactivity.
Command:     AUTH TLS
Response:    234 AUTH TLS OK.
Status:      Initializing TLS...
Status:      Verifying certificate...
Command:     USER manga247_com
Status:      TLS/SSL connection established.
Response:    331 User manga247_com OK. Password required
Command:     PASS *******
Response:    230-Your bandwidth usage is restricted
Response:    230-OK. Current restricted directory is /
Response:    230 205 Kbytes used (0%) - authorized: 1048576 Kb
Command:     OPTS UTF8 ON
Response:    200 OK, UTF-8 enabled
Command:     PBSZ 0
Response:    200 PBSZ=0
Command:     PROT P
Response:    200 Data protection level set to "private"
Status:      Connected
Status:      Retrieving directory listing...
Command:     PWD
Response:    257 "/" is your current location
Status:      Directory listing successful
```

Enablt TLS connection: OK.
Transfer data/delete data OK