HACK-STACK Setup Part 2

Set up some Folders, Files, and Users. Add an FTP server.

We want to set up a web site that will allow users to upload content that the site will incorporate into web pages for display. We want to do this in a way that does not require us to trust users.

We need to control access to certain folders in such a way as to allow our code to write and read website user uploaded content with several security considerations.

First, we do not allow any online user to upload any content to our DOCUMENT_ROOT

Second, since we do want to display uploaded user content we need to set up a place to upload and store the content.

Third, we need to set up access permissions and ownership for those upload directories

If this is done we can enjoy a more secure website and still allow the upload and display of user uploaded content. These ideas can avoid the popular Mkdir: Permission denied: ... error message.

So the first thing we'll do is create a user group and control access to the upload areas through membership in the group.

We will use www-pub for the group name and we will add two users to the group. One user is daemon which we see by looking at the ownership of the htdocs folder. The second user we add is named ftpuser and we use the same password Hacker as for our other users.

```
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# clear
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# ll
total 8432
```

```
drwxr-xr-x 14 root root      4096 Oct 16 20:51 ./
drwxr-xr-x 20 root root      4096 Oct 16 20:52 ../
drwxr-xr-x  2 root root      4096 Oct 16 20:49 bin/
-r-xr--r--  1 root root   8573582 Oct  3 10:09 bnconfig*
drwxr-xr-x  2 root root      4096 Oct 16 20:49 build/
drwxr-xr-x  2 root root      4096 Oct 16 20:51 cgi-bin/
drwxr-xr-x  5 root root      4096 Oct 17 12:52 conf/
drwxr-xr-x  3 root root      4096 Oct 16 20:49 error/
drwxr-xr-x  3 root daemon    4096 Oct 17 15:11 htdocs/
drwxr-xr-x  3 root root      4096 Oct 16 20:49 icons/
drwxr-xr-x  2 root root      4096 Oct 16 20:49 include/
drwxr-xr-x  2 root root      4096 Oct 20 18:50 logs/
drwxr-xr-x  2 root root      4096 Oct 16 20:49 modules/
drwxr-xr-x  2 root root      4096 Oct 16 20:49 scripts/
drwxr-xr-x  3 root root      4096 Oct 16 20:49 var/
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# addgroup www-pub
Adding group `www-pub' (GID 1003) ...
Done.
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# sudo adduser daemon www-pub
Adding user `daemon' to group `www-pub' ...
Adding user daemon to group www-pub
Done.
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# adduser ftpuser
Adding user `ftpuser' ...
Adding new group `ftpuser' (1004) ...
Adding new user `ftpuser' (1002) with group `ftpuser' ...
Creating home directory `/home/ftpuser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password: [enter Hacker]
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for ftpuser
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n]

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# adduser ftpuser www-pub
Adding user `ftpuser' to group `www-pub' ...
Adding user ftpuser to group www-pub
Done.

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# getent group www-pub
www-pub:x:1003:daemon,ftpuser
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2#
```

So now we have a group and two users as described. Now let's set up a folder structure to accommodate our web site needs. We will create four folders at the same level as our apache2 folder.

```
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# ll
total 8432
drwxr-xr-x 14 root root      4096 Oct 16 20:51 ./
drwxr-xr-x 20 root root      4096 Oct 16 20:52 ../
drwxr-xr-x  2 root root      4096 Oct 16 20:49 bin/
-r-xr--r--  1 root root   8573582 Oct  3 10:09 bnconfig*
drwxr-xr-x  2 root root      4096 Oct 16 20:49 build/
drwxr-xr-x  2 root root      4096 Oct 16 20:51 cgi-bin/
drwxr-xr-x  5 root root      4096 Oct 17 12:52 conf/
drwxr-xr-x  3 root root      4096 Oct 16 20:49 error/
```

```
drwxr-xr-x  3 root daemon    4096 Oct 17 15:11 htdocs/
drwxr-xr-x  3 root root      4096 Oct 16 20:49 icons/
drwxr-xr-x  2 root root      4096 Oct 16 20:49 include/
drwxr-xr-x  2 root root      4096 Oct 20 18:50 logs/
drwxr-xr-x  2 root root      4096 Oct 16 20:49 modules/
drwxr-xr-x  2 root root      4096 Oct 16 20:49 scripts/
drwxr-xr-x  3 root root      4096 Oct 16 20:49 var/

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# mkdir uploads
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chown root:www-pub uploads
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chmod g+rwxs uploads

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# mkdir Storybook
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chown root:www-pub
Storybook
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chmod g+rwxs Storybook

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# mkdir includes
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chown root:www-pub includes
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chmod g+rwxs includes

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# mkdir session2DB
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chown root:www-pub
session2DB
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# chmod g+rwxs session2DB

root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# ll
total 8448
drwxr-xr-x 18 root root       4096 Oct 23 13:13 ./
drwxr-xr-x 20 root root       4096 Oct 16 20:52 ../
drwxrwsr-x  2 root www-pub    4096 Oct 23 12:47 Storybook/
drwxr-xr-x  2 root root       4096 Oct 16 20:49 bin/
-r-xr--r--  1 root root    8573582 Oct  3 10:09 bnconfig*
drwxr-xr-x  2 root root       4096 Oct 16 20:49 build/
drwxr-xr-x  2 root root       4096 Oct 16 20:51 cgi-bin/
drwxr-xr-x  5 root root       4096 Oct 17 12:52 conf/
drwxr-xr-x  3 root root       4096 Oct 16 20:49 error/
drwxr-xr-x  3 root daemon     4096 Oct 17 15:11 htdocs/
drwxr-xr-x  3 root root       4096 Oct 16 20:49 icons/
drwxr-xr-x  2 root root       4096 Oct 16 20:49 include/
drwxr-xr-x  2 root www-pub    4096 Oct 23 13:05 includes/
drwxr-xr-x  2 root root       4096 Oct 20 18:50 logs/
drwxr-xr-x  2 root root       4096 Oct 16 20:49 modules/
drwxr-xr-x  2 root root       4096 Oct 16 20:49 scripts/
drwxrwsr-x  2 root www-pub    4096 Oct 23 13:10 session2DB/
drwxrwsr-x  2 root www-pub    4096 Oct 23 12:45 uploads/
drwxr-xr-x  3 root root       4096 Oct 16 20:49 var/
```

We now have the folders with permissions and ownership as shown here in the directory above. Now we will create two subfolders under our newly created Storybook folder.

```
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# mkdir Storybook/htdocs
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# mkdir Storybook/conf
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2# ll Storybook
total 16
drwxrwsr-x  4 root www-pub 4096 Oct 23 13:13 ./
drwxr-xr-x 18 root root    4096 Oct 23 13:13 ../
drwxrwsr-x  2 root www-pub 4096 Oct 23 13:13 conf/
drwxrwsr-x  2 root www-pub 4096 Oct 23 13:13 htdocs/
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2#
```

Using vim we create the following file within our apache2/htdocs folder and save it as mkdirtest.php. You will need to delete the space character between the opening < and the question mark.

```
< ?php
/* filename: mkdirtest.php
 * this code tests the ability to make a directory
 * */

// disable error reporting for production code
 error_reporting(E_ALL);
 ini_set('display_errors', TRUE);
//
 echo 'Current Working Dir '. getcwd() .'
';
// echo phpinfo();
 $old = umask(0000);
 if(!(is_dir('/opt/hhvm-3.30.12-5/apache2/Storybook/htdocs/Test'))) {
                 echo 'Directory does not exist.'
';
        mkdir('/opt/hhvm-3.30.12-5/apache2/Storybook/htdocs/Test', 0775,
true); }
        umask($old);
?>
```

If we point our Windows browser at 127.0.0.1:8080/mkdirtest.php we get the expected message and if we look at the Storybook/htdocs folder we will see a newly added subfolder Test

Using vim we create the following file within our apache2/htdocs folder and save it as makefiletest.php. You will need to delete the space character between the opening < and the question mark.

```
< ?php
/*
 * filename: testMakeFile.php
 * this code tests creating a file
*/

// disable error reporting for production code
error_reporting(E_ALL);
ini_set('display_errors', TRUE);

$testString = "E Pluribus Unum";
$destinationPath = '/opt/hhvm-3.30.12-5/apache2/Storybook/htdocs/Test/';
$destinationFile = 'testFile.txt';
echo 'Current Working Dir '. getcwd() .'
';
//echo phpinfo();

if(is_dir($destinationPath)) {
        $result = file_put_contents($destinationPath.$destinationFile,
$testString); }
?>
```

If we point our Windows browser at 127.0.0.1:8080/Storybook/Test/testFile.txt we get the expected message and if we look at the Storybook/htdocs folder we will see a subfolder Test with the testFile.txt file in it.

Now we want to navigate to the /opt/hhvm-3.30.12-5/apache2/Storybook/conf folder and use vim to create three configuration files there.

The first files is httpd-vhosts.conf with the following content

```
  ServerName Storybook.example.com
  DocumentRoot "/opt/hhvm-3.30.12-5/apache2/Storybook/htdocs"
  Include "/opt/hhvm-3.30.12-5/apache2/Storybook/conf/httpd-app.conf"
```

The second file is httpd-prefix.conf with this content

```
Alias /Storybook/ "/opt/hhvm-3.30.12-5/apache2/Storybook/htdocs/"
Alias /Storybook "/opt/hhvm-3.30.12-5/apache2/Storybook/htdocs"
Include "/opt/hhvm-3.30.12-5/apache2/Storybook/conf/httpd-app.conf"
```

The third file is httpd-app.conf with this content

```
    Options Indexes MultiViews
    AllowOverride All

    Order allow,deny
    Allow from all

    = 2.3>
    Require all granted
```

These edits result in a folder with contents like this

```
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2/Storybook/conf# ll
total 20
drwxrwsr-x 2 root www-pub 4096 Oct 23 15:47 ./
drwxrwsr-x 4 root www-pub 4096 Oct 23 13:13 ../
-rw-r--r-- 1 root www-pub  270 Oct 23 15:34 httpd-app.conf
-rw-r--r-- 1 root www-pub  200 Oct 23 15:41 httpd-prefix.conf
-rw-r--r-- 1 root www-pub  229 Oct 23 15:47 httpd-vhosts.conf
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2/Storybook/conf#
```

Now let's restart the LAMH stack and continue after that.

```
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2/Storybook# /opt/hhvm-
```

```
3.30.12-5/ctlscript.sh restart
Syntax OK
/opt/hhvm-3.30.12-5/apache2/scripts/ctl.sh : httpd stopped
/opt/hhvm-3.30.12-5/hhvm/scripts/ctl.sh : hhvm stopped
/opt/hhvm-3.30.12-5/mysql/scripts/ctl.sh : mysql stopped
/opt/hhvm-3.30.12-5/mysql/scripts/ctl.sh : mysql  started at port 3306
/opt/hhvm-3.30.12-5/hhvm/scripts/ctl.sh : hhvm started
Syntax OK
/opt/hhvm-3.30.12-5/apache2/scripts/ctl.sh : httpd started at port 80
root@BobosRevenge:/opt/hhvm-3.30.12-5/apache2/Storybook#
```

Now if we point our browser at 127.0.0.1:8080/Storybook/Test/testFile.txt we will see the E Pluribus Unum motto as the result. If we copy our index.php file from out htdocs document root into the Storybook folder it can be executed from there with a request to 127.0.0.1:8080/Storybook or 127.0.0.1:8080/Storybook/index.php

What this means is that we can serve content that is not contained within the default htdocs folder.

Now that we can serve content in this fashion, let's set up an FTP server to help us "upload" content. Now you can easily mount Windows drives in the stack, but an FTP server is somewhat more familiar since it is more like our usual access to a server. I like the Windows Filezilla client but it is not available for Linux nor is the server.

We chose to install vsftpd as our FTP server with an apt install vsftpd command on the Ubuntu console. After installation we will use vim to edit the config file named /etc/vsftpd.conf. We want the end results of our edit to look like this following

Example config file /etc/vsftpd.conf

```
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd
options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
#
# Run standalone?  vsftpd can run either from an inetd or as a standalone
```

```
# daemon started from an initscript.
listen=YES
listen_port=8021
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on
specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=NO
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_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 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
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
#anon_mkdir_write_enable=YES
#
# Activate directory messages - messages given to remote users when they
# go into a certain directory.
dirmessage_enable=YES
#
# If enabled, vsftpd will display directory listings with the time
# in  your  local  time  zone.  The default is to display GMT. The
# times returned by the MDTM FTP command are also affected by this
# option.
use_localtime=YES
#
# Activate logging of uploads/downloads.
xferlog_enable=YES
#
# Make sure PORT transfer connections originate from port 20 (ftp-data).
#connect_from_port_20=YES
#
# If you want, you can arrange for uploaded anonymous files to be owned by
# a different user. Note! Using "root" for uploaded files is not
# recommended!
#chown_uploads=YES
#chown_username=whoever
#
# You may override where the log file goes if you like. The default is
shown
# below.
#xferlog_file=/var/log/vsftpd.log
#
# If you want, you can have your log file in standard ftpd xferlog format.
# Note that the default log file location is /var/log/xferlog in this
```

```
case.
#xferlog_std_format=YES
#
# You may change the default value for timing out an idle session.
#idle_session_timeout=600
#
# You may change the default value for timing out a data connection.
#data_connection_timeout=120
#
# It is recommended that you define on your system a unique user which the
# ftp server can use as a totally isolated and unprivileged user.
#nopriv_user=ftpsecure
#
# Enable this and the server will recognise asynchronous ABOR requests.
Not
# recommended for security (the code is non-trivial). Not enabling it,
# however, may confuse older FTP clients.
#async_abor_enable=YES
#
# By default the server will pretend to allow ASCII mode but in fact
ignore
# the request. Turn on the below options to have the server actually do
ASCII
# mangling on files when in ASCII mode.
# Beware that on some FTP servers, ASCII support allows a denial of
service
# attack (DoS) via the command "SIZE /big/file" in ASCII mode. vsftpd
# predicted this attack and has always been safe, reporting the size of
the
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
ftpd_banner=Welcome to HACK STACK's FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses.
Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
#
# You may restrict local users to their home directories.  See the FAQ for
# the possible risks in this before using chroot_local_user or
# chroot_list_enable below.
#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().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure
that
# the user does not have write access to the top level directory within
the
# chroot)
#chroot_local_user=YES
#chroot_list_enable=YES
# (default follows)
#chroot_list_file=/etc/vsftpd.chroot_list
#
# You may activate the "-R" option to the builtin ls. This is disabled by
# default to avoid remote users being able to cause excessive I/O on large
# sites. However, some broken FTP clients such as "ncftp" and "mirror"
assume
```

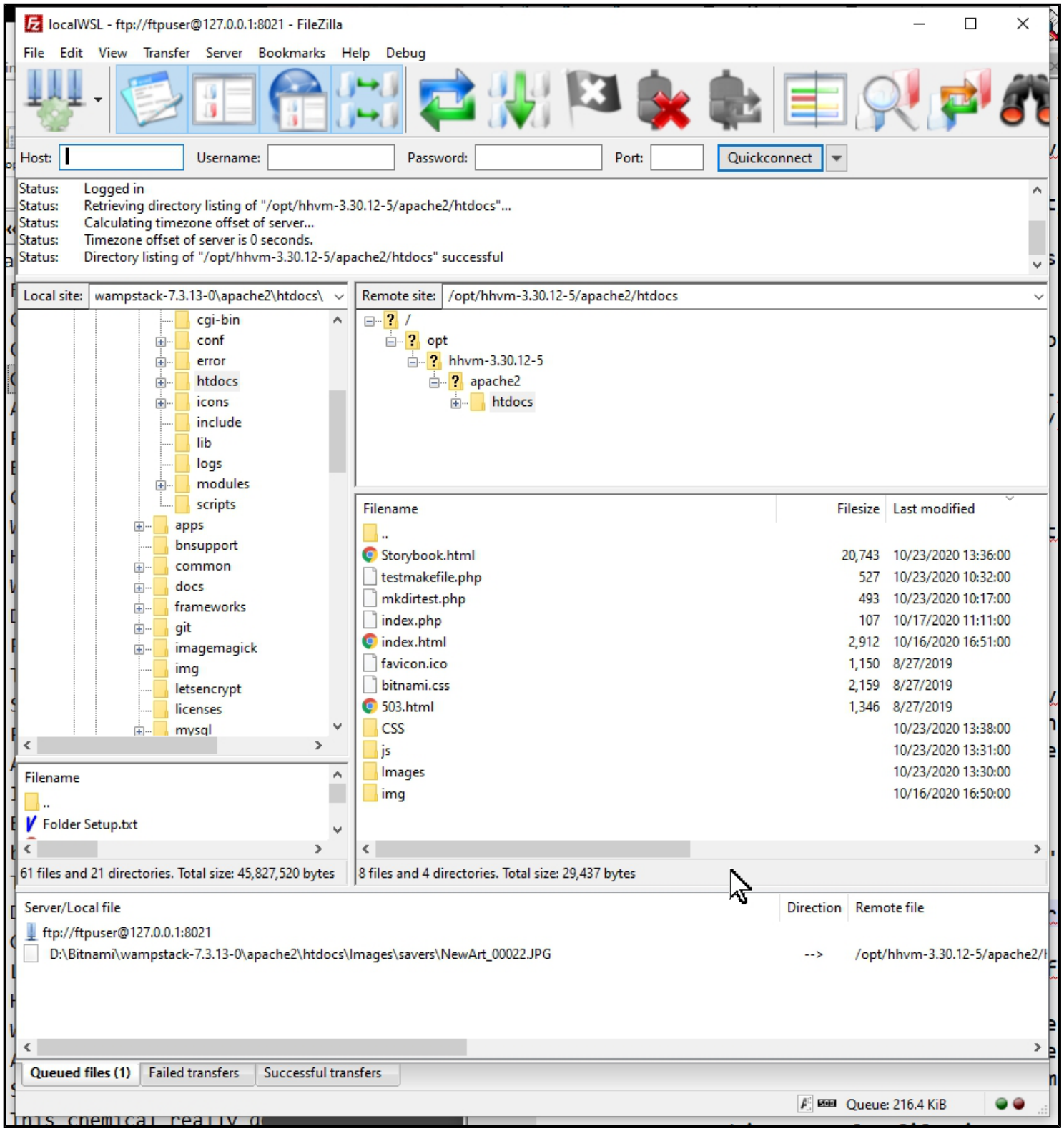
```
# the presence of the "-R" option, so there is a strong case for enabling
it.
#ls_recurse_enable=YES
#
# Customization
#
# Some of vsftpd's settings don't fit the filesystem layout by
# default.
#
# This option should be the name of a directory which is empty.  Also, the
# directory should not be writable by the ftp user. This directory is used
# as a secure chroot() jail at times vsftpd does not require filesystem
# access.
secure_chroot_dir=/var/run/vsftpd/empty
#
# This string is the name of the PAM service vsftpd will use.
pam_service_name=vsftpd
#
# This option specifies the location of the RSA certificate to use for SSL
# encrypted connections.
rsa_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
rsa_private_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
ssl_enable=NO

#
# Uncomment this to indicate that vsftpd use a utf8 filesystem.
#utf8_filesystem=YES
```

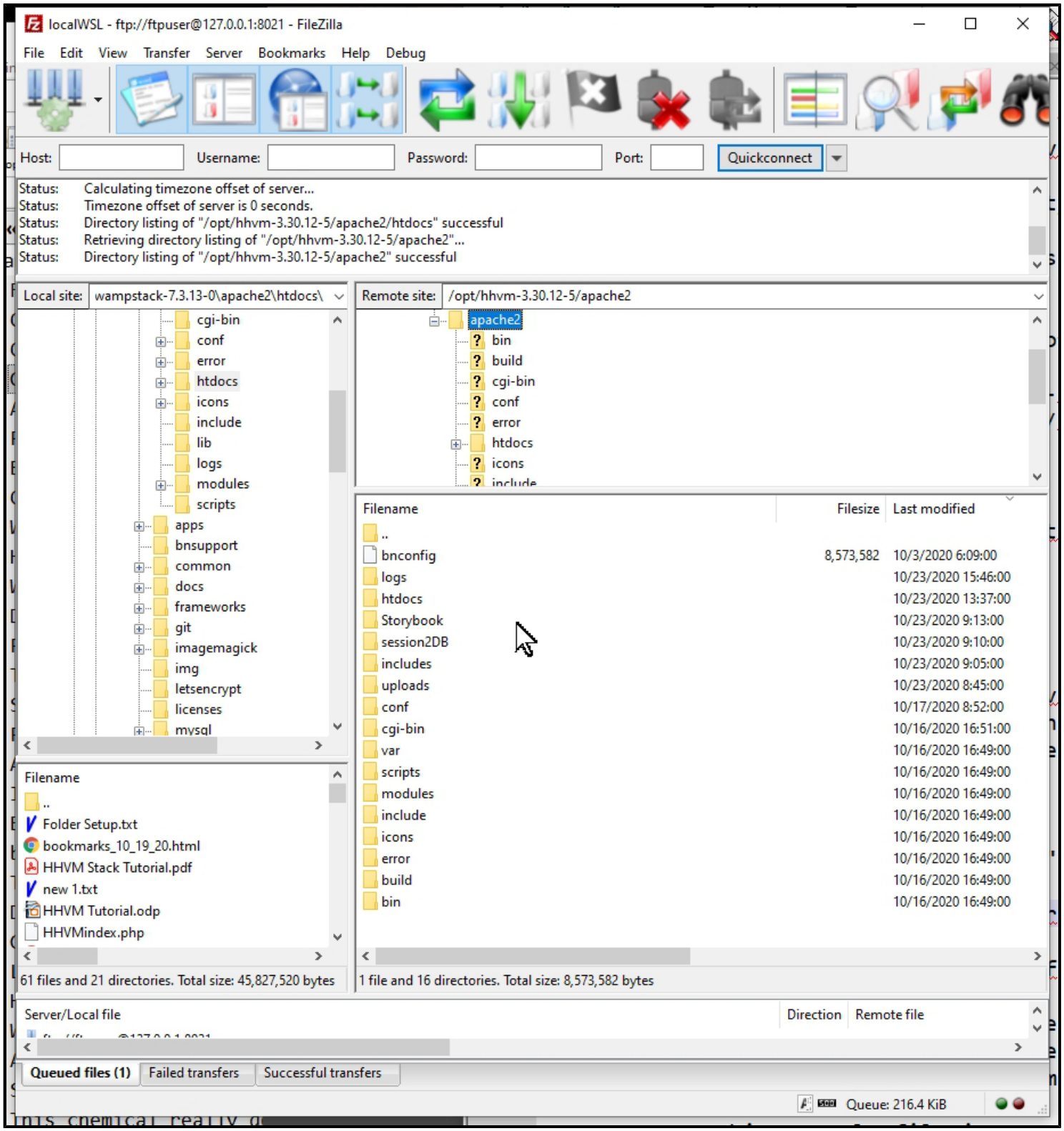
After we complete these edits we will restart the vsftpd service with the command

```
root@BobosRevenge:/# service restart vsftpd
No /usr/sbin/vsftpd found running; none killed.
root@BobosRevenge:/# service vsftpd restart                             [
OK ]
 * Starting FTP server vsftpd                                           [
OK ]
root@BobosRevenge:/#
```

Now we can connect to our "web server" using a Windows FTP client like FileZilla or even the Windows Command prompt FTP client using Port 8021 and the FTP username ftpuser with a password of Hacker. The two images show screens from the Windows FileZilla client connected to our server.



ftpClientScreen1.jpg



ftpClientScreen2.jpg

Note especially that the folder Storybook is not within the htdocs folder but is instead at the same level in the folder hierarchy. The LAMH stack here at The Palace is presently running a replica of a page served on an AWS Bitnami LAMP stack.

The final step we should take is to execute these commands in our Windows Admin Command prompt to back up the changes we have made.

```
Microsoft Windows [Version 10.0.19041.572]
(c) 2020 Microsoft Corporation. All rights reserved.
C:\WINDOWS\system32>wsl --list --all
Windows Subsystem for Linux Distributions:
Ubuntu-18.04 (Default)
C:\WINDOWS\system32>wsl --export Ubuntu-18.04 D:/UbuntuHHVMbase.tar
C:\WINDOWS\system32>
```

Next time we'll use more of this files-folders-users hierarchy in our LAMH stack and web sites.