pgFtp v 0.9.2 Readme
=====================

Contents:

1. Legal
========

This software is Copyright (C) 2004 University of Calgary

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

pgFtp is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but is provided AS IS, WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, and NON-INFRINGEMENT. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

In addition, as a special exception, the copyright holders give permission to link the source code of this program with the OpenSSL library, and distribute linked combinations including the two. You must obey the GNU General Public License in all other respects for all of the code used other than OpenSSL. If you modify this version of pgFtp, you may extend this exception to your version, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

2. Changes
==========

v 0.9.2

- pgFtp now includes support for the SFTP protocol!

- pgFtp can now act as an authentication plugin in addition to (or instead of) providing roaming profile functionality. Users can be authenticated against either an FTP or SFTP (SSH) server.

- Added a progress dialog for roaming profiles on login / logout

- Improved the level of detail given in error messages

- Made it a bit easier to translate user messages into other languages. System generated errors will be displayed in the native language of the system, and all other strings that pgFtp uses are stored as resources in pgftp.dll. To modify or translate the text, simply load up the dll in any Windows resource editor.

- Passive mode is now the default for FTP connections. Previously, the plugin was mistakenly hard coded to use active mode, which was causing connection problems with firewalls (the client Windows firewall was particularly troublesome since it would sometimes allow you to connect and sometimes not depending on whether the firewall was fully loaded when you logged in after a reboot). You can still use active mode if you want, but I wouldn't recommend it unless you have the Windows firewall turned off.

- Fixed a bug where the plugin couldn't detect the difference between logging in and unlocking a workstation in newer versions of pGina. Special thanks goes to Christian in the pgFtp forum for providing a temporary fix to this problem and the one above!

- Improved the upload process -- a failure during upload won't leave the user with a corrupt, partially saved profile

- Changed the way users get an initial roaming profile. Previously, you were either forced to setup a central default profile or manually create a profile.zip file for each user. Now, if there is no default profile path configured, new users will automatically start roaming whatever local profile they get when they log on with pgFtp for the first time. If you do set up a central default profile, then the user will start off with a copy of that one instead of any local one.

- To accommodate some feedback and requests, profile filenames have been changed. Instead of profile.zip (or profile.7z), profiles are now named username.profile.zip (.7z). This makes it possible to store profiles in the same directory if you prefer. Note that if you are using regular FTP, you may still prefer to store each profile in a separate directory (see Configuration section). For backwards compatibility, pgFtp will still recognize the old naming convention and automatically switch to the new one.

- server port is now a configurable option

- Version information for the plugin is now displayed in the configuration dialog and in the property page in Windows Explorer

- Created an installer for the plugin

- Due to the use of OpenSSL, the plugin now has a slightly modified license. Basically, it's the GPL with a special exception clause for allowing linking with the OpenSSL library, whose license is in conflict with the GPL. See http://www.gnome.org/~markmc/openssl-and-the-gpl.html if you want to read up on the issue. If you do redistribute your own version of pgFtp compiled with OpenSSL, you will need to obey the terms of the OpenSSL license (namely, the infamous "advertising clause").

v 0.9.1

- Replaced the ZipArchive implementation with 7Zip since the zip file format does not work properly with files that have Unicode extended characters in their file names.

- Profiles are now backed up on the server before they are overwritten

v 0.9.0

- Initial public release

3. About

pgFtp is a pGina plugin that allows administrators to authenticate users against an FTP or SSH server.

Furthermore, users can have central roaming profiles stored on an FTP or SFTP server. Although pGina itself contains roaming profile functionality, it is somewhat limited since it requires that the account used to login has access to the profile server. In many configurations of pGina, the authentication is done via a plugin (e.g. this one) and the user logs in using a temporary Windows account on the local computer. Depending on your environment, it can be difficult or impossible to get your profile from the server using this account.

In addition to bypassing these issues, the pgFtp plugin gives you more flexibility by allowing roaming over any TCP/IP connection. Also, by compressing the profile, pgFtp can reduce central profile size and provide performance increases over slow connections.

Please note that this project should be considered as "beta" software. Although I do my best to ensure it works as it should, I cannot test all possible configurations and environments. Please make sure you test it thoroughly before deploying it in any production environment.

4. Requirements

- pGina version 1.7.7.4 or higher
- Windows 2000 or XP client machines with:
    - Windows shell version 6.0.1 or higher (comes with Internet Explorer 6.0 Service Pack 1 or later)
    - at least one NTFS partition if you are using roaming profiles
- If you plan to use the plugin for profiles only, then you will also need:
    - pGina Chaining Plugin
    - pGina authentication plugin of some sort

4. Configuration

Follow these steps for any configuration of pgFtp:

1. If you intend to use pgFtp for profiles only, you must first configure pgina to use the Chaining Plugin and another authentication plugin.

WARNING: If pgFtp is in "profiles only" mode, it allows ANY user to log in regardless of what they type for the user name or password. When using pgFtp this way, always make sure you have another plugin properly configured for authentication.

When using the Chaining Plugin, always configure pgFtp to be last in the chain.

2. In the configuration dialog for pgFtp, set the Role value to the appropriate setting. Choose whether or not the plugin is required (this only affects authentication when not using chaining).

3. On the Server tab, specify the host name, port, and protocol to use. Do not append any url prefix like "ftp://," just the host name or ip address.

The username and password are optional and are primarily intended for allowing anonymous FTP access to profiles. Note that the authentication portion of pgFtp always ignores these values; user will be authenticated based on the credentials they type in.

4. If you are using SFTP, you can optionally enter the host key. The host key is the 128-bit MD5 hash of the server's private key, specified as a 16-digit hex string. If you omit the host key, then pgFtp will show a message the first time you log in, displaying the key and asking you if you want to save it for future reference. By pre-entering the key, you can avoid having users ever see this message.

Follow these additional steps ONLY if you are using pgFtp for roaming profiles:

1. On the Plugin tab of the configuration dialog, specify the full path to the temporary directory used for downloading / uploading profiles. This directory MUST reside on an NTFS partition (see Known Issues section). In most cases, just leave it at its default. pgFtp will create the directory if it doesn't already exist.

2. Adjust the desired level of compression. In most cases, the default setting of "Fastest" will provide the best overall performance. If the machine has a particularly slow connection to the server, you may want to increase the level of compression. If you have a fast connection, but a slow machine, you may want to disable compression. The profile will still be saved in zip or 7-zip format, but it won't be compressed (essentially acting like a tar file, but not in tar format).

3. On the Profile tab of the pGina (not pgFtp) configuration dialog, setup the profile path (required) and default profile path (optional). These are paths on the server; as such, they need to follow the path-naming conventions that the server understands (typically UNIX-style, case-sensitive path names).

The profile path is where each user's individual profile is stored. All users must have read / write access to this directory. Typically, this setting points to somewhere in the user's home directory, although you may choose to store all profiles in one directory.

SECURITY NOTE: When using FTP, pgFtp does not explicitly set any permissions on the profile file; the permissions for the file will be entirely dependant on the behaviour of the server. The FTP standard does not have any command for setting file permissions. Many servers implement the "SITE CHMOD" quasi-standard, but not all servers do, and some may have the command disabled for security reasons. To avoid inconsistent behaviour in pgFtp, it is up to you to ensure that permissions are properly set. Storing the profiles in the users' home directories is usually the safest bet. If you use SFTP, then the profile is always written with mode 600.

By default, profiles will be saved as username.profile.zip . If you are using 7-zip compression, then they will be named username.profile.7z (see Using 7-zip Compression below).

4.1 Default Profiles

Whenever a user logs on to a roaming profile enabled machine, and they don't already have a roaming profile, pgFtp will automatically create one for the user. The profile they get will be a copy of whatever profile Windows assigned them when they logged in. If they already had an account on the local machine, then they will get their existing profile. Otherwise, they get a copy of the local Default User profile.

You can change this behaviour by assigning a specific default profile to users who don't already have one. This option is useful when you want to have more centralized control over users' desktops. To create a default profile, follow these steps:

1. Create a new local user account on a machine, and login as that user.
2. Make any adjustments to the user environment, registry, etc. that you want to take effect for new users.
3. IMPORTANT! When you are done, open the registry editor (regedit.exe) and explicitly grant Full Control permissions to the Everyone group for the HKEY_CURRENT_USER registry key.
4. Log off, then log back in as administrator. Follow the instructions in the section "Creating a Roaming Profile Manually" for the user account you just created. The file name must be called "default.profile.zip" (or .7z).
5. Make the file available on the FTP / SFTP server. All users must have read permissions for the file.
6. Configure the default profile path in pGina to point to this file. Do not include the filename in the path.

NOTE: Once you configure a default profile path, pgFtp will enforce that new users can only become roaming profile users if they receive a copy of the profile at that location. If for some reason a new user cannot download the default profile, they will get an error message and be logged in with a local one instead. However, this local profile will not be uploaded on logout.

4.2 Using 7-zip Compression

By default, profiles are stored in standard zip file format. If you prefer, you can use 7-zip compression instead. At the time of this writing, 7-zip provides the highest level of compression as compared to other common formats (zip, gz, arj, rar, etc.). To read up on 7-zip, visit www.7-zip.org.

More importantly, 7-zip is one of the few formats that can handle Unicode filenames correctly. Standard zip files store filenames within the archive using 8-bit characters. This creates a problem for users that have files with international characters. Although you can get around this somewhat using code pages, it still prevents a roaming user from logging on to a machine with a different code page than the files were created in. If any of your users use an extended character set for their file names, then you should use 7-zip compression.

- To enable 7-zip compression for all new users, setup a default profile and compress it using 7-zip.
- To convert an existing roaming profile from zip to 7-zip, simply unzip the profile into a temp directory and recompress it with 7-zip, giving it a .7z extension instead of .zip. Replace the orignial zip file with this one.

NOTE! If you have both username.profile.zip and username.profile.7z in the profile path, pgFtp ignores the 7-Zip version and always uses the zip one. To avoid confusion, make sure you only have one copy.

pgFtp comes with 7ZA.exe, which is a command-line tool for working with both 7-zip and zip files, but you are free to use whatever utility you want.

4.3 Creating a Roaming Profile Manually

Sometimes it may be necessary to manually convert a local profile to a roaming one. To do so, follow these instructions:

1. Locate the profile you wish to convert in the local profiles folder (by default, this is C:\Documents And Settings).

IMPORTANT! If you are converting anything other than Default User, you must first log in as that user, open the registry editor (regedit.exe) and explicitly grant Full Control permissions to the Everyone group for the HKEY_CURRENT_USER registry key.

2. Zip up the contents of the user's profile directory. Do not include the

directory itself in the archive. Also, in almost all cases, you should exclude the contents of the "Local Settings" directory. This folder contains temp files and the Internet Explorer file cache.

IMPORTANT! If you use the 7ZA.exe utility that comes with pgFtp to create a .zip file, make sure you specify the -tzip parameter. 7ZA.exe defaults to 7-zip format regardless of the file extension you use.

3. Save the archive as username.profile.zip (or .7z, depending on the file format) and upload it to the profile path on the server. Make sure the file has appropriate read/write permissions for the user.

The next time the user logs in, they will get this profile.

5. Troubleshooting Tips

Q. I upgraded to version 0.9.2, but I don't see any of the new features?

A. The new installer for 0.9.2 installs to C:\Program Files\pgFtp by default, which is a different location than the previous instructions for manual installation suggested. You will need to reconfigure pGina and/or the chaining plugin to point to the new location if you accepted the default.

Q. I get the message "Unable to run the archiving utility." on login / logout

A. 7ZA.exe must be present in the same folder as pgftp.dll. If it's already there, make sure you can run it manually from a command prompt.

Q. I get the message "The archiving utility reported an unknown error while processing your profile."

A. More than likely, the profile is corrupt. Try uncompressing the profile manually with 7ZA.exe to see what the problem is. If you get output that reads "error: error," that's its cryptic way of telling you that the file is toast. pgFtp always backs up the most recent version of the profile to username.profile.zip.bak, so look for the presence of this file on the server and remove the .bak extension. If that doesn't work, then you will have to delete the profile and give the user a fresh one.

Q. Profiles take ages to transfer. Is there any way to speed it up?

A. There are several factors that can affect performance:

First, there is currently a known problem with SFTP transfer speed (see Known Bugs / Limitations). If you don't mind clear text passwords, then you should consider using regular FTP instead for now since it will be faster.

An easy way to cut down on profile size is to redirect certain shell folders to network drives. You can do this by modifying these two registry subkeys of HKEY_CURRENT_USER:

```
        \Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
        \Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders
```

To do this, have pGina map a drive (say H:) to the user's home directory and point these keys to the H: drive. Move the files from the local folders to the network drive. This way, My Documents, etc won't be transferred every time users log in and out. I recommend against redirecting the Desktop folder to a network drive, however, since in our testing at the U of C, Windows started behaving erratically if the network connection was ever lost, even for a second. Also, do not store the Local Settings folder on a network drive. It's called Local for a reason... performance for the user will seriously degrade. Of course, I can't guarantee that there aren't other problems associated with this redirection technique. Make sure you test it thoroughly.

Finally, try playing around with the compression settings, and test the performance of using 7-zip versus zip.

Q. Roaming profile functionality seems to work but the profile I get seems strange -- e.g. I get the "classic" Windows visual theme and many items are missing from the start menu. Changes that I make (like changing the wallpaper) don't seem to stick.

A. More than likely this is a permissions issue with the registry. Using regedit, make sure you assign Full Control to Everyone for the HKEY_CURRENT_USER key as detailed in the Configuration section. Also, make sure any default profile is configured this way as well (if not, then this was probably the original source of the problem)

Q. None of these problems apply. What should I do?

A. The first step is to try and simulate the plugin manually. Using an FTP or SFTP client, download the profile to the temp directory, unzip with 7ZA.exe, rezip it, rename the old profile to username.profile.zip.bak, and reupload the file. If you cannot perform these steps, then pgFtp can't either. Often this will catch problems you might not have thought of, like invalid file permissions and such. Other than that, have a look at the log file for your FTP / SSH server. If you are adventurous, have a look at the source code to see exactly what is going on.

If the problem is with the profile itself, then bear in mind that any problems with Windows roaming profiles in general will carry over in pgFtp. All the plugin is doing is tricking Windows into thinking that a local temp directory is a central profile repository. Try looking up the problem on http://support.microsoft.com or any other source that provides information on Windows profiles in general.

Failing that, post a description of the problem in the pgFtp forum at http://forums.xpasystems.com and hopefully someone will be able to help you out.

7. Known Bugs / Limitations

- There is currently a problem with the speed of SFTP transfers being slower than they should. It appears to be related to libssh2, but I'm not entirely sure. If you think you can help solve this, have a look at my post at

http://sourceforge.net/forum/forum.php?thread_id=1366787&forum_id=428318

- Changing passwords is not implemented in this plugin. Unfortunately, neither the FTP nor the SFTP protocol implements this functionality. If you intend to use pgFtp as your primary authentication plugin, then you will need to provide users with an alternate way to change their passwords (e.g. a web page or instructions for doing it via a shell prompt). In the future, I may implement a way to run a user-specified script via telnet / SSH.

- If you specify SFTP as the protocol, but your host / port combo refers to something other than an SSH server (e.g. you type port 80 which is HTTP), then pgFtp may potentially crash. Unfortunately, this appears to be a problem with the current version of libssh2. The configuration dialog always switches the default port when you change protocols. For this reason, it's recommended that you always use the configuration tool instead of manually changing settings in the registry.

- If using pgFtp for authentication only, then your SSH server must implement and accept SFTP sessions, even though no files are being transferred. This limitation will probably be removed in future versions.

- The temp profile path must point to a directory on an NTFS partition. Windows only recognizes a path as a profile path if it has specific permissions and ownership set. If you use FAT32, it fails with an error message complaining about the permissions.