
Subject: Re: SSH package for U++

Posted by [alkema_jm](#) on Fri, 01 Dec 2017 11:33:16 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Oblivion ,

>Thank you very much for your comments and suggestions.

Thank you that you ask feedback :p

>I've tested swish and it does a decent job for sure.

I tried (/test) to compile Swish. At one point i could start it, but it crashed.

> However, I believe SSH package and swish do not fall under the same categories. Their scope and intended audience seems to be different.

Ok, I think in software components. I like the Windows Explorer integration of Swish only.

>From its website: "Swish is a plugin for Microsoft Windows Explorer that adds support for SFTP."

It is a decent plugin for Windows Explorer, it falls under applications/plugins category, if I may.

Yes, I like Microsoft Windows Explorer integration part of Swish.

>Naturally, it is limited to Windows platform, and as far as I can understand, it uses OpenSSL exclusively.

I don't like. "One size fits all". Some solutions (for example Microsoft Windows Explorer integration) can be better in specific (native) solutions.

>And it seems to be limited to SFTP subsystem.

The FileZilla engine 'uses' more protocols (storj) :

> On the other hand SSH package isn't an app or plugin written by U++. It is an easy to use SSH2 library (it covers full SSH2 ecosystem: Sftp, channels, exec, scp, shell, agents) for U++, and its intended audience is exclusively U++ developers.

I like the SSH2 ecosystem very much :p

> SSH uses libssh2 under the hood. libssh2 is a widely used and extensively tested, multiplatform SSH2 library with BSD license. SSH package supports what libssh2 supports.

Ok, You limit SSH package to "libssh2"?

> Therefore SSH package is not limited to windows. In fact, it isn't developed on windows. In

theory SSH package can run on what U++ and libssh2 can run on and this includes, but not limited to, Linux and Windows.

I think it is usefull to think in interfaces (and api) of software components. I separated the (General) FileZilla engine code. I will try to interface with libssh2 (SSH package). I haven't look in the libssh2 code if it is possible.

> Also, while SSH package uses OpenSSL by default, it is also possible to use it with WinCNG and GnuTLS. (Currently through a config file, in the near future via compiler flag.)

FileZilla uses GnuTLS.

> In fact there is nothing that prevents a developer to come up with a similar plugin to swish, using SSH pacakge.(Maybe it'll take a lot less time to develop one using SSH package, since the main purpose of SSH package is to let developers focus on other aspects of their apps.)

I like your SSH package very much, because it Works :p

> One major advantage (for commercial app developers) of SSH package over swish is that SSH package (and underlying libssh2) uses BSD license.

I am not a company. Licenses are not very interesting subject for me.

> To sum up:

>1) Swish offers good functionality, but the functionality it offers is mostly on application-level. Hence they should be implemented by the application developers (using SSH package).

Ok.

2) Writing some example app similar to swish would be platform specific. Therefore I am writing a simple SFtpBrowser example, which will work on any platform that U++ supports.

Ok, a simple example is alway usefull to see that it Works. I think that a more complex example als FileZilla is not difficult to migrate to Ultimate++.

Greetings Jan Marco

File Attachments

1) [protocolinfos_engine_filezilla.jpg](#), downloaded 1459 times
