
Subject: Re: SFTP or full SSH2 support for U++?
Posted by [Oblivion](#) on Sat, 12 Aug 2017 20:49:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello everyone,

Next update to SSH package technical preview is finally here.

Aside from bug fixes and cleanups, SSH package has now gained full support for three authentication methods (password, publickey, keyboard-interactive) properly. The crucial part was keyboard interaction (challenge/response) authentication method. It is implemented using a Function.

Also, it is now possible to select between preferred transport methods.

Last but not least, API reference docs are added for the most crucial classes. :)

I am also writing a guide for its usage. But it's not finished yet.

Agents, X11 support, and forwarding support are on their way way too!

Now I'd like to ask you a question. This covers both SSH package and my FTP package. I'm planning to add to both package network proxy (HTTP/SOCKS4/4a/5) support using the new version of my NetworkProxy package (it will be renamed to NetProxy).

I can think of three strategies:

- 1) Adding proxy support as an integral part to SSH and FTP packages. This will cause SSH and FTP packages to require NetProxy and I don't really prefer this.
- 2) Using a compiler flag for enabling proxy support.
- 3) Since NetworkProxy (and upcoming NetProxy) is a delegate class that simply takes an already initialized TcpSocket, connects it, and hands it over back to user, I can add a callback, say WhenProxy, to Ssh and Ftp classes, and hand over the socket to NetProxy there, and when the connection is established we can proceed. This way both packages won't require NetworkProxy at all, it will be completely optional.

What do you think? Which strategy would be better?

I appreciate suggestions, bug reports, criticism, etc.

Here's a digest:

2017-08-11: Initial api reference docs for Ssh, Ssh::SubSystem, SFtp, Scp, Exec, Knownhosts, are added.

2017-07-31: It is now possible to query, get, and set the possible transportation methods and exchange

algorithms. Added Ssh::Method(), Ssh::Methods(), Ssh::GetMethod(), Ssh::GetMethods() methods.

Ssh::Host class is from now on KnownHosts class.

2017-07-21: Authentication methods (password, public key, keyboard interaction) are properly

implemented.

From now on it is possible to choose between authentication methods both on initialization,

and on-the-fly (i.e. while logging in, using WhenAuth callback).

KnownHosts class is added to the package. This class provides basic known hosts support.

It is now possible to verify and trust servers.

Best regards,
Oblivion

File Attachments

1) [SSH Package \(technical preview\)-2017811.zip](#), downloaded 269 times
