

StartOpenSSH v0.5.1 (Version 1 Beta)

Read Me for: v0.5.1(22)

Author Information

©2013, Richard Williamson All Rights Reserved.
 Written by: Richard Williamson
 Contact Info: richard@r1ch4rd.net

Quick Links on SSH

1. What is SSH: http://en.wikipedia.org/wiki/Secure_Shell
2. How to create SSH RSA Key: <http://macnugget.org/projects/publickeys/>
3. What is ssh tunneling: http://en.wikipedia.org/wiki/Tunneling_protocol
4. What is sshpass: <http://sshpass.sourceforge.net/>
5. What is copymykey: <http://r1ch4rd.net/>

Features of StartOpenSSH

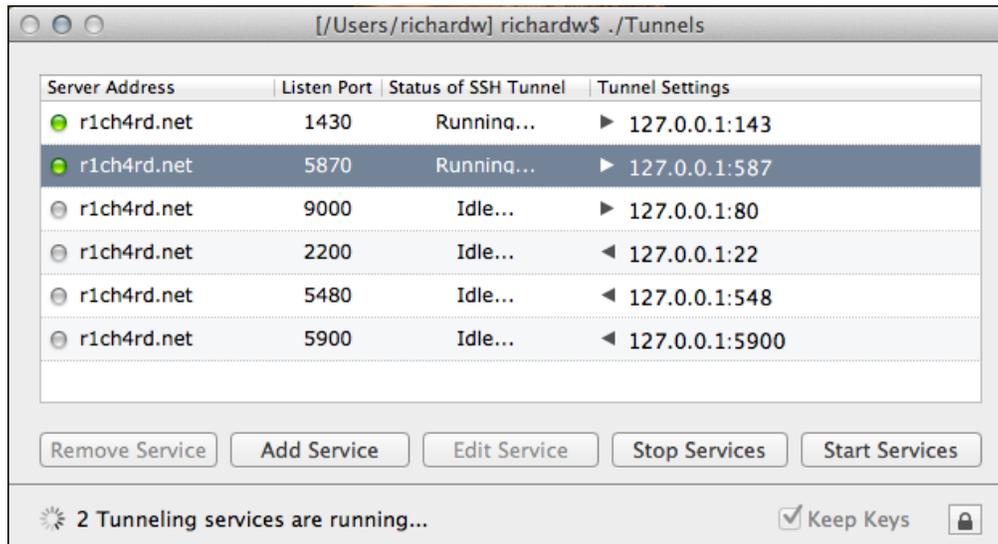
- Wraps around the command line app ssh.
- Allows you to add multiple RSA keys with or without passwords
- Allows you to securely lock the app so no one walking buy can change the state of the app or see/edit your data
- Allows you to create tunnels with multiple ssh servers
 - you can choose a specific service to forward to or you can use it as a socks proxy server
 - you can setup a reverse tunnel back to your computer
- Allows you to use a ssh-askpass proxy for entering passwords (10.7)
 - it can automate the login process with no user input of passwords
 - you can choose a method where it prompts you for your password
 - With the release of Mountain Lion (10.8) the ssh-askpass is no longer used by default
 - You can use sshpass (a third party app) to manage your passwords for you if no RSA key is available.
 - If you so choose you can use a different third party command line password manager
- Automatic reconnect if the connection is lost.
- It is the one size fits all solution to GUI SSH Tunneling and once released it will be the pinnacle in its class.

Using this app

You will need to have access to a SSH server to test this application.

ATTENTION: You must have connected to a ssh server previously before using StartOpenSSH, as you will need to download the server key, which is received the first time you connect. If you have never used SSH before I'll tell you how to do this further down.

Windows of StartOpenSSH



Main Window of StartOpenSSH

Features of the Main Window

This is the window where the average user will spend most of their time. It is the home place of the app! Here you will start and stop tunnel sessions, which we have named "service" in this app. As you can see there is a progress wheel which shows you that there are services running. Next to each "Server Address" there is a circle icon, which shows the current state of each service. White indicates that no service is running, yellow indicates that a service is starting, red indicates that the service has failed and green indicates that the service is running correctly. You can start each service individually or you can start them all at the same time. To start one service click on that service and then click on "Start Service". If you wish to start all services at once make sure no service is selected and click on "Start Services". You can also stop each service individually or you can stop all of them all at once. To stop one service click on that service and then click on "Stop Service". If you wish to stop all services at once make sure no service is selected and click on "Stop Services". If you wish to delete a service simply click on that service and then click on "Remove Service". Now that you've read all that to see a blank screen lets click on 'Lock' icon to unlock the application, then click the "Add Service" button and move on to our next window.

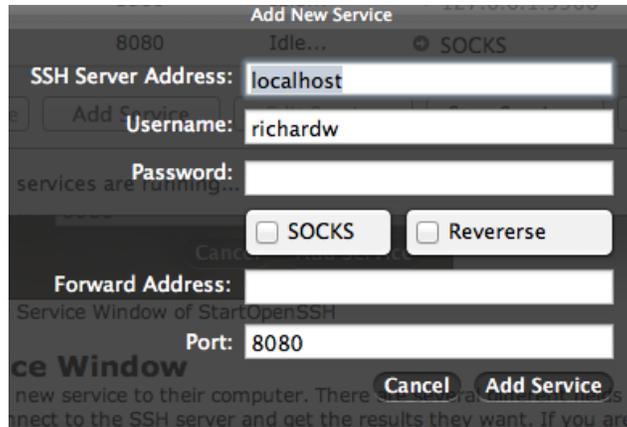
The application will lock after one minute of you not using the application or when you click the lock button. This has been implemented so other people can not edit/see your data or add services without approval. By default it will unload the key files you have specified to have loaded, to change this click the box next to "Keep Keys".

Icon Key

-> Triangle means forward will occur from your computer to the address through the host

<- Triangle means forward will occur from the host to your computer through the host

(->) circle with arrow in means that you have setup a SOCKS server with the host. Anything that supports socks will be allowed to tunnel through you to the host and then out to a different destination.



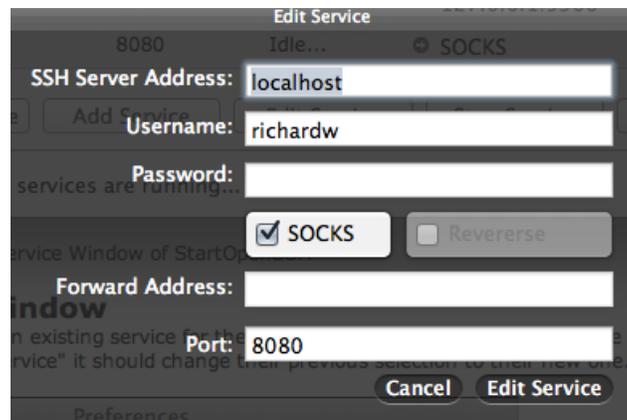
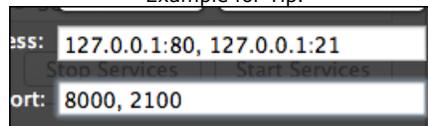
Add New Service Window of StartOpenSSH

Features of the Add New Service Window

This is the window where the users will add a new service to their computer. There are several different fields available to the user, which need to be filled out correctly to connect to the SSH server and get the results they want. If you are experienced with SSH then you can move on to the next window after reading the next sentence. If you are using sshpass or ssh-askpass (10.7) then put your password in if you do not have a public key, otherwise do not put your password here as it will not be needed. If you wish to setup a SOCKS server locally to send the requests through the ssh server select SOCKS, for a specific service, uncheck socks and put in the forward address i.e.: 127.0.0.1:110 (the servers pop3 service). If you want to setup a reverse tunnel to your computer from the host select the reverse option. Lets say you have ssh running on port 22 you would put in 127.0.0.1:22 for the Forward Address, and for port if you put in 2200 you could connect to 127.0.0.1:2000 on the ssh server and it would forward the connection to the ssh server on your computer.

Tip: You can add multiple services at the same time for one host by separating the Forward Address: Port: items with a comma. If you separate one with commas you'll need to do the same amount of commas for the other. However you will not be able to do that in the edit window.

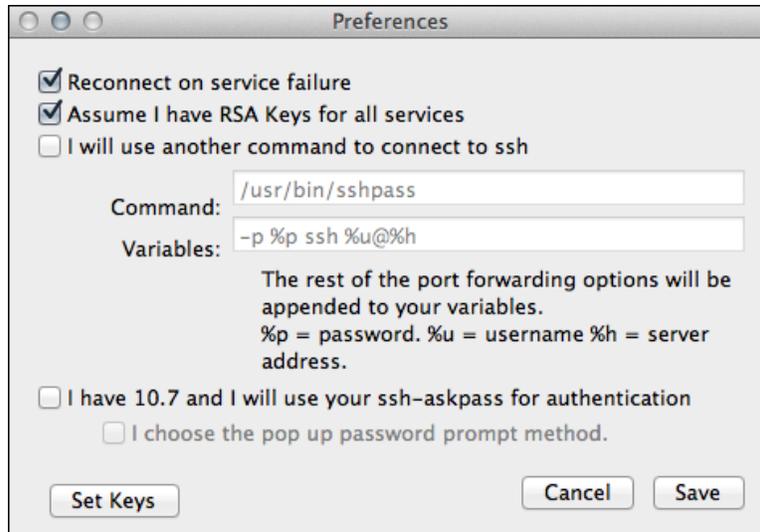
Example for Tip.



Edit Service Window of StartOpenSSH

Features of the Edit Service Window

This is the window where the users will edit an existing service for their computer. It should function just like the Add New Service Window. When the user clicks "Edit Service" it should change their previous selection to their new one.



Preferences Window of StartOpenSSH

Preferences Window

This is the window where the user will choose how StartOpenSSH will function for them.

The reconnect on failure feature allows a connect to retry if the process is killed or the connection was lost due to internet disruption or the connection was killed on the server side. By default if the service simply fails due to authentication errors it will not retry. There is a bug where if you specify the wrong command as in its not there it will loop trying to reconnect. At this time I am looking at the code to fix it.

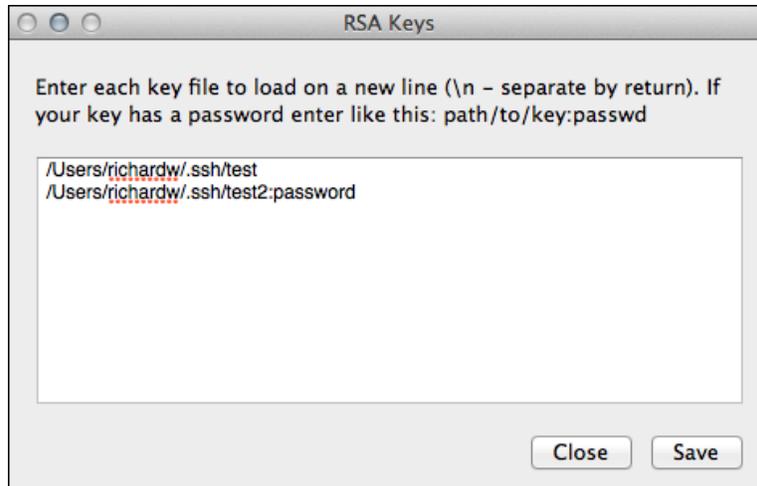
By default, according to Apples security settings and top security experts today, the application is setup to assume you will be using your public RSA Key to connect to SSH servers. When running StartOpenSSH in this mode you should not put any passwords in as it assumed you will not need them to connect to the server. If you don't have a matching key on the server it will fail regardless of you putting in a password or not as the password is simply ignored at all times, other then saving it in an encrypted text file. According to experts this is the most secure method of connecting to a SSH server, which is why it is the default method.

Sadly in the real world the most secure method is not a method that works for everyone else, so I have created two other ways to allow you to use this application if you don't have RSA keys. These password methods are less secure but effective if you password protect your computer, run a screen saver if you walk away, and encrypt your home directory. With the following features this app should easily become the pinnacle of SSH Tunneling GUI's.

One cutting edge feature of StartOpenSSH allows you to use a third party command line utility to handle your passwords for logging into servers where you do not have a RSA key. If you choose to install sshpass (or depending on which third party password utility you choose) you will need to put passwords in for the servers you do not have a RSA key for. If you don't have sshpass installed, every service will fail.

If you are using Mac OSX Lion (10.7), I have created a password solution for users that don't have an RSA key for a specified server. If you have Mac OSX Mountain Lion you can not use this service by default and its rather difficult to change the settings to make it work, do not install the package for ssh-askpass, as you will do nothing for you. If you have Mac OSX 10.7 you will need to run the package installer for ssh-askpass. By default it will install the password prompt method. The password prompt method allows you to not have to type your password in as when you start a service it will pop up with a window for you to type it in at that time. This is the most secure method of of this application if you do not have a RSA key. Sadly in 10.8 I have been unable to get it to work. If you wish to choose the automated method I have included that as well, but when you install the package you will need to specify custom install and uncheck the password prompt ssh-askpass and then check the ssh-askpass automated method.

To add different RSA Keys for StartOpenSSH to use click on the "Set Keys" button and you will see.



Enter the full path to the key on one line, if the key file is password protected enter a colon and then the password.
No Password Example: /Users/myuser/.ssh/id_rsa
Has Password Example: /Volumes/PenDrive/my_file:mypassword

So if those where your two keys you would enter it in on each line by itself like so...

```
/Users/myuser/.ssh/id_rsa  
/Volumes/PenDrive/my_file:mypassword
```

Final Thoughts

If you have questions please feel free to contact me, I am more then willing to help you figure out how to use this app. Hopefully the read me file is informative enough that the average computer user can figure out what this app is and how to use it.

Attention: The following is for people who don't have SSH Server access.

Turning on SSH Server

1. Go to Apple Menu -> System Preferences
2. Click on "Sharing"
3. You will see a list of check boxes. Check the box "Remote Login".
4. Make sure "All users" is selected or at the very least your username is the "Only these users" box.

Connecting to a SSH Server Your First Time

1. Go to your "Applications" folder, open "Utilities"
2. Open up "Terminal"
3. Type in ssh yourusername@127.0.0.1
5. It will ask you a yes or no questions type yes

That is all you need to do. Follow the steps in link two to setup your RSA key, also needs to be done in terminal. To put your key in place for automatic log ins do the following in terminal. cp .ssh/id_rsa.pub .ssh/authorized_keys . Make sure the period is included in .ssh/ .

I have included copymykey for ease of copying your keys over for the beta.