Getting Started with Secure Shell Telnet Client and Vulcan

Developed by SSH Communications Security Ltd., Secure Shell is a program to log into another computer over a network, to execute commands on a remote machine, and to move files from your local machine to a remote machine. It provides strong authentication and secure communications over insecure channels.

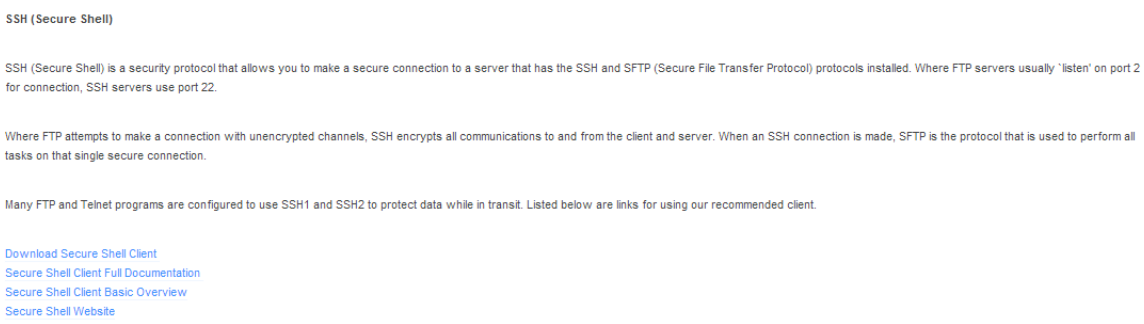
Two features that are very important in any Internet related course are: **telnet** and **ftp.**

Telnet: Telnet is a protocol for accessing (logging into) a remote computer on the Internet. To telnet to a remote machine, you need a program (called telnet client) on your machine. Of course, for telnet to work, the remote machine must be configured as a telnet server.

Ftp: FTP stands for File Transfer Protocol, a protocol for transferring files between a remote server and your computer. Of course, for ftp to work, the remote machine must be configured as an ftp server.

For improved security this server requires the Secure Shell Telnet Client (commonly called **ssh**) to access these accounts.

Download SSH (Download Secure Shell Client) from the Web site:
http://csis.pace.edu/support/support.html:



SSH (Secure Shell)

SSH (Secure Shell) is a security protocol that allows you to make a secure connection to a server that has the SSH and SFTP (Secure File Transfer Protocol) protocols installed. Where FTP servers usually 'listen' on port 2 for connection, SSH servers use port 22.

Where FTP attempts to make a connection with unencrypted channels, SSH encrypts all communications to and from the client and server. When an SSH connection is made, SFTP is the protocol that is used to perform all tasks on that single secure connection.

Many FTP and Telnet programs are configured to use SSH1 and SSH2 to protect data while in transit. Listed below are links for using our recommended client.

Download Secure Shell Client
Secure Shell Client Full Documentation
Secure Shell Client Basic Overview
Secure Shell Website

Both **telnet** and **ftp** are included in **ssh** software – integrated into one program.

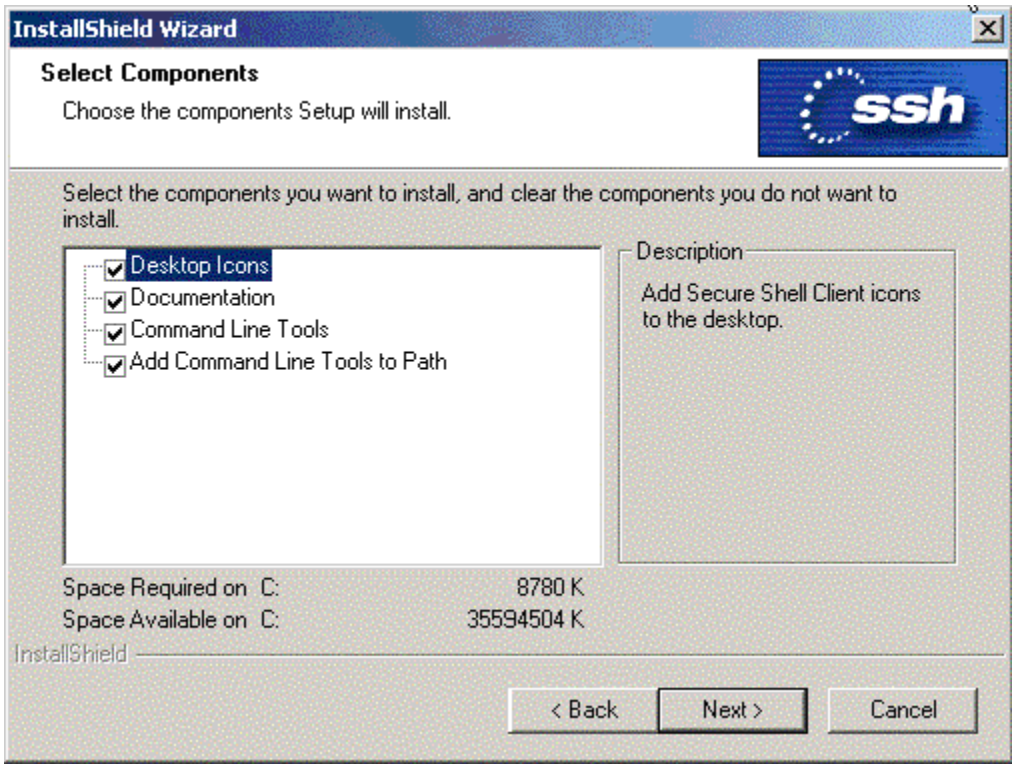
This following document is not intended to be an in depth examination of the SSH client, but should provide you with enough guidance to get you up and running. Full documentation can be found at http://www.ssh.com.

Installation of ssh
The installation file is compressed in a zip format to minimize download time. If you don't already have a utility to uncompress (unzip) the file – get one. Most files available for download on the Internet will be compressed to minimize their size and

help download times. Newer versions of many operating systems actually have the utility built in. But if you need one, an example of an unzip/zip application, one is available at http://www.winzip.com/.

After you uncompress the zip file you will have a single executable file called SSHWinClient.exe.

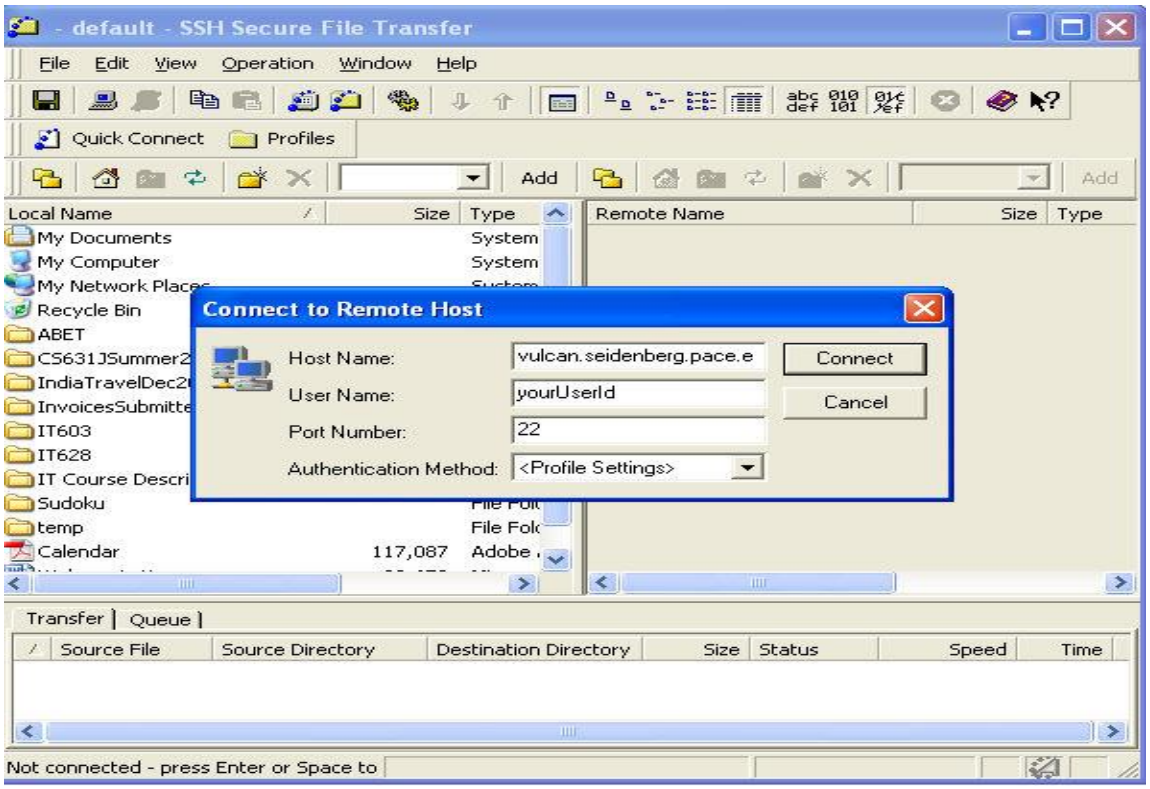
Launch this executable to begin the installation. You will be shown a few screens where you will be prompted to click on "Next". After clicking "Yes" for the license agreement, you will be prompted to enter where you want the installed files to be placed on your computer. The default value is usually acceptable. After this step, you will be shown a screen where you will select various components. The default with all checkboxes selected is usually acceptable. Click on "Next", and the final step is to click on "Finish" and your SSH program is installed.



Using SSH

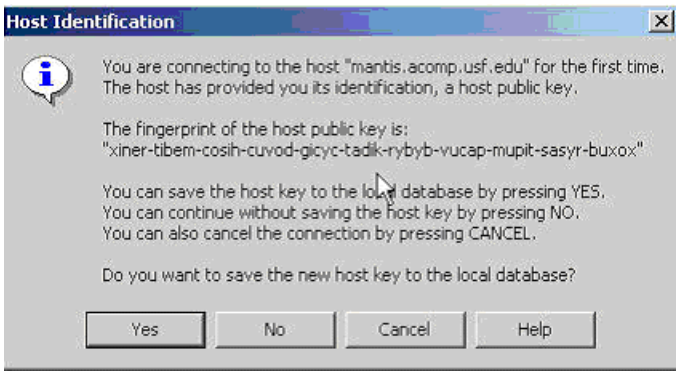
When installation is complete, run the Secure Shell Client from either the desktop icon or Windows Start Menu. The gray window text indicates an inactive SSH connection:

Press Enter or Space to initiate a Quick Connect.
Complete the screen as shown:
Host Name: vulcan.seidenberg.pace.edu
User Name: *your user id* (as given by me)
Post Number: 22
Authentication Method: <Profile Settings> (leave this as is)



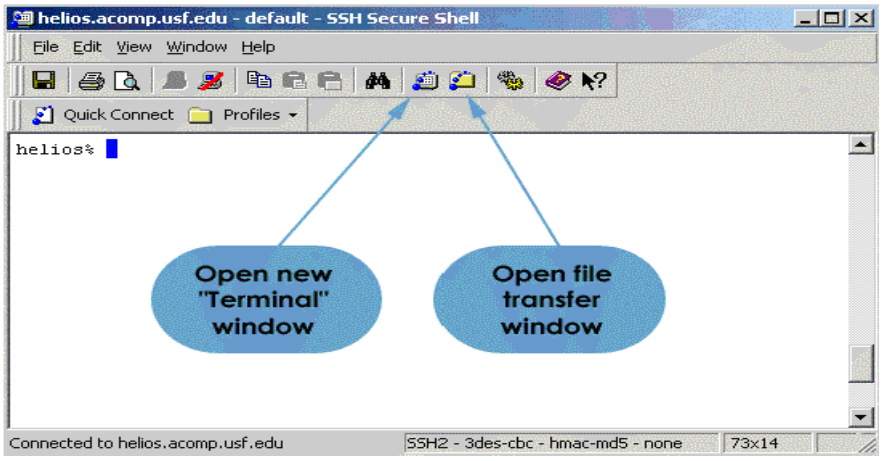
Press **Connect**.

You will be informed that this is the first time you have connected to this host and asked if you want to save its public key. Select **Yes** to remember the host's unique identification.



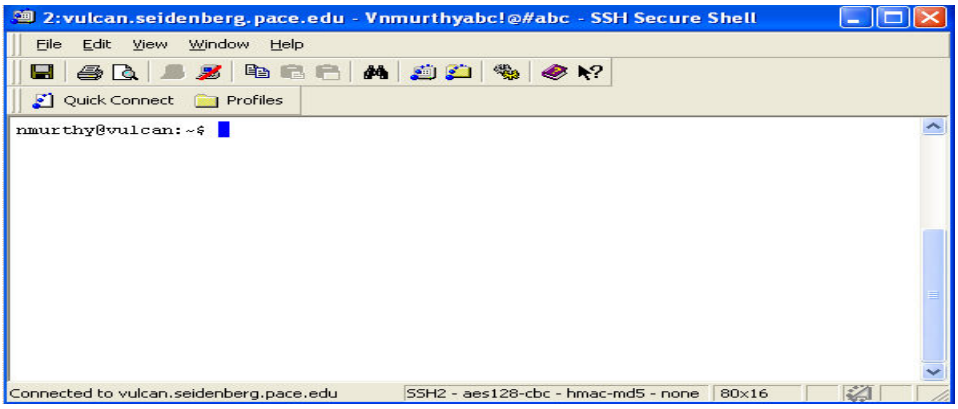
When prompted, enter your password to complete the secure connection. Terminal based commands and programs may now be used.

To transfer files across the secure connection, click on the "Open file transfer window" icon. You may then drag and drop files to and from Windows Explorer.

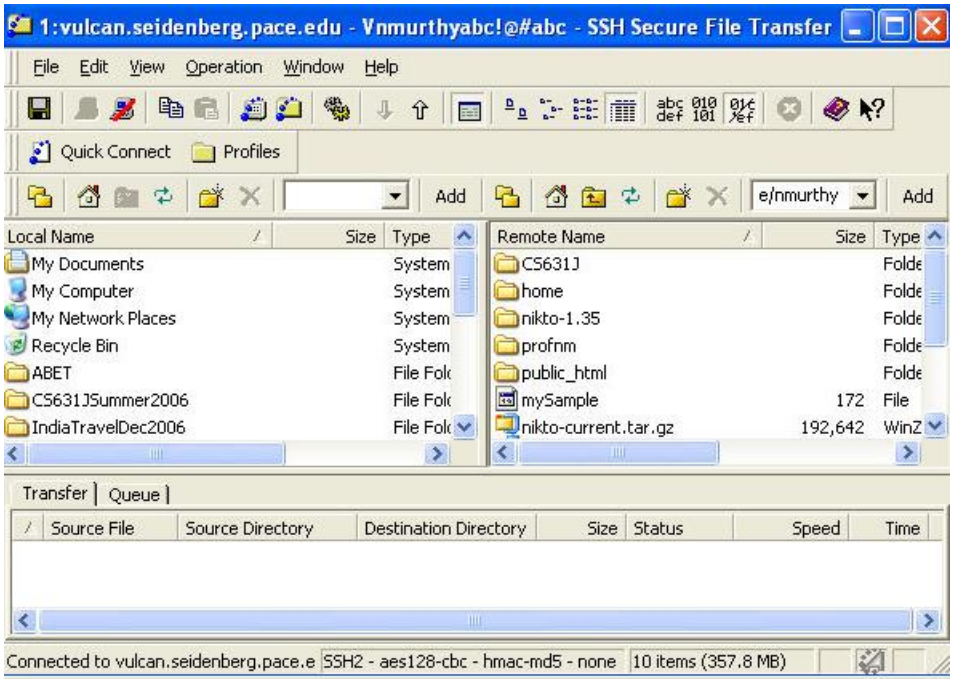


(Your prompt will be something different from helios%.)

Here is my sample "Terminal Window". This is a Linux machine. Following the prompt, you can give any Linux command. The prompt in the example screen below, is nmurthy@vulcan:~$. This is prompt for my account on Vulcan. Your prompt will have your user id instead of nmurthy.



Here is my sample "File Transfer Window" (use this interface to 'move' files between your computer to server - by dragging the file):



In the middle, the left panel is a directory on your local machine and the right panel is the "home directory" of your account on the remote machine. You can open a directory by double clicking it. You can the usual traversing, using mouse. Play with it.

An important note:
One of the **ftp** activities you normally do is to upload HTML files from your machine to the directory **public_html** in your home directory on the remote machine. First open the appropriate folder on your local machine (left panel) and open the directory

public_html (on the right panel). Now drag the file (using mouse pointer) from left panel to the right panel. See the file name on the right panel. Now the HTML file in in the public_html directory on the server.

Use the following URL to access the file in a browser:
http://vulcan.seidenberg.pace.edu/~xxxxxxxxx/example.html

Instead of xxxxxxxxx, you actually type your Vulcan user id. You need ~. Notice that you don't type public_html in the URL.
