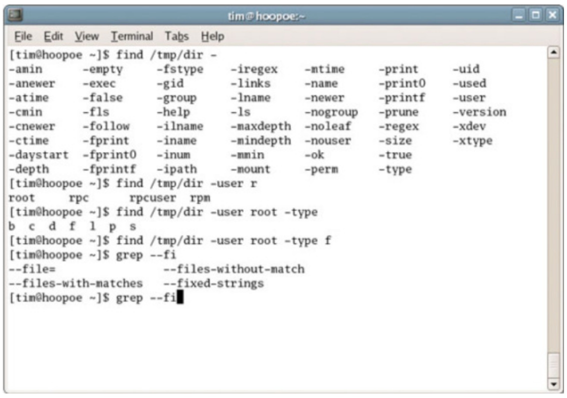
FTP and **SSH** are both network protocols that run on top of the TCP/IP layer, just like HTTP. In plain English, it's a certain way for devices to communicate with each other over a network.

Shell & Shell Accounts

Let's get some basic terminology out of the way. To understand the purpose of SSH, you need to be familiar with some of the underlying elements.

The **shell** of a computer is a piece of software that allows users to directly communicate with the kernel, the very core of an operating system. The shell can have either a graphical or command-line interface (read: text input), or both.



```
[tim@hoopoe ~]$ find /tmp/dir -
-amin      -empty     -fstype    -iregex    -mtime     -print     -uid
-anewer    -exec      -gid       -links     -name      -print0    -used
-atime     -false     -group     -lname     -newer     -printf    -user
-cmin      -fls       -help      -ls        -nogroup   -prune     -version
-cnewer    -follow    -ilname    -maxdepth  -noleaf    -regex     -xdev
-ctime     -fprint    -iname     -mindepth  -nouser    -size      -xtype
-daystart  -fprint0   -inum      -mmin      -ok        -true
-depth     -fprintf   -ipath     -mount     -perm      -type
[tim@hoopoe ~]$ find /tmp/dir -user r
root     rpc      rpcuser  rpm
[tim@hoopoe ~]$ find /tmp/dir -user root -type
b  c  d  f  l  p  s
[tim@hoopoe ~]$ find /tmp/dir -user root -type f
[tim@hoopoe ~]$ grep --fi
--file=                --files-without-match
--files-with-matches   --fixed-strings
[tim@hoopoe ~]$ grep --fi
```

A shell account, on the other hand, is a personal account that gives the user access to a shell on a different computer. They used to be commonplace and supplied by the Internet Service Providers, used to work with file storage, email accounts, newsgroups and more. The common denominator is that a shell account is used to enter commands on a remote computer.

Secure Shell Protocol (SSH)

Just like a web browser uses the HTTP protocol to talk with websites, a shell account needs a certain protocol to enable data exchange.

SSH uses a public key encryption and was developed to replace Telnet and other insecure shell protocols. The two major versions, SSH-1 and SSH-2, are now the dominating protocols to access shell accounts.

These days, SSH is used to log into and execute code on remote hosts, browse the web using encrypted proxy clients, and transfer files – even setting up a Virtual Private Network.

SSH clients are available for all major operating systems. Unix-based systems, including Linux and Mac OS X, can use OpenSSH. Also check the OpenSSH website for Mac OS and Windows alternatives. PuTTY is one of the most prominent Windows clients.

Secure File Transfer Protocol (SFTP) versus FTP

File transfer and VPN applications don't run on SSH by default, but make use of SFTP – the **SSH File Transfer Protocol**. Mind you, SFTP is not the FTP protocol running over SSH, but a different file transfer protocol developed as an extension for SSH-2. SFTP is always used to transfer files over SSH, but it's actually designed so it can be used in compliance with other protocols.

SFTP *can* be seen as a secure relative of FTP. The FTP transmits all data in plain-text. Packet intercepts can thus reveal crucial and private data, including your user name and password! SFTP, being an SSH-2 extension, uses public key security. This means the data is encrypted when it is being transmitted and potential intercepts are relatively useless.