Interactive vs Non-interactive Sessions

The shell behaves in slightly different ways when you log on interactively versus when you just connect to run a single command. Here is an example to better illustrate the difference between interactive and non-interactive shells.

Interactive:

```
mac:~ bob$ ssh linuxsvr
Last login: Thu Nov  7 01:26:37 UTC 2013
Welcome to Ubuntu 12.04.3 LTS

 * Documentation:  https://help.ubuntu.com/

  System info as of Nov 14 01:26:52 UTC 2013

  System load:  0.42
  Usage of /:   3.1% of 40GB
  Memory usage: 32%
  Swap usage:   0%
  Processes:           89
  Users logged in:     0
  IP address for eth0: 10.0.0.7

bob@linuxsvr:~$ uptime
 11:49:16 up 97 days,  2:59,  5 users,  load average: 0.15, 0.25, 0.31

bob@linuxsvr:~$ ll
-rw-r--r-- 1 bob bob 221 Nov 13 11:30 file.txt
bob@linuxsvr:~$ exit
logout
Connection to 10.0.0.7 closed.
mac:~ bob$
```

Non interactive:

```
mac:~ bob$ ssh linuxsvr uptime
 11:49:16 up 97 days,  2:59,  5 users,  load average: 0.15, 0.25, 0.31
mac:~ bob$ ssh linuxsvr ll
bash: ll: command not found
mac:~ bob$
```

The contents of `.profile` or `.bash_profile` are only executed for interactive sessions. If you are not aware of this subtle difference it may leave you scratching your head as to why something works perfectly when you log in and type a command versus when you just ssh in to run that same command. For example, if you define an alias for `ll` in `~/.bash_profile` it will work during an interactive session but it will not be available during a non-interactive session.

You can save yourself some hassle by making your interactive and non-interactive sessions behave the same. To do this, configure `.bash_profile` to reference `.bashrc` and put all of your configuration in `.bashrc.` You can read in the contents of another file by using the `source` command or dot operator.

`source filename` - Read and execute commands from filename and return. Any variables created of modified in filename will remain available after the script completes.

`. filename` - Same as `source filename`.

```
$ cat .bash_profile
# Put our settings in .bashrc so we have the same environment for login and non-
```

```
login shells.
if [ -f ~/.bashrc ]; then
   source ~/.bashrc
fi
$ cat .bashrc
# use a vi-style line editing interface
set -o vi

# Set the prompt.
PS1="[\u@\h \w]\$ "
export PS1

# Set the PATH.
PATH=$PATH:~/bin
export PATH

# Aliases
alias grpe='grep'
alias ll='ls -l'
alias utc='TZ=UTC date'
alias vi='vim'
```

Now the aliases you have defined are available during interactive and non-interactive sessions. Here is how the sessions behave after this change.

Interactive:

```
mac:~ bob$ ssh linuxsvr
Last login: Thu Nov  7 01:26:37 UTC 2013
Welcome to Ubuntu 12.04.3 LTS

 * Documentation:  https://help.ubuntu.com/

  System info as of Nov 14 01:26:52 UTC 2013

  System load:  0.42
  Usage of /:   3.1% of 40GB
  Memory usage: 32%
  Swap usage:   0%
  Processes:           89
  Users logged in:     0
  IP address for eth0: 10.0.0.7

bob@linuxsvr:~$ ll
-rw-r--r-- 1 bob bob 221 Nov 13 11:30 file.txt
bob@linuxsvr:~$ exit
logout
Connection to 10.0.0.7 closed.
mac:~ bob$
```

Non interactive:

```
mac:~ bob$ ssh linuxsvr ll
-rw-r--r-- 1 bob bob 221 Nov 13 11:30 file.txt
mac:~ bob$
```