



Connecting to NERSC and Transferring Data

February 1, 2012

David Turner
NERSC User Services Group



National Energy Research
Scientific Computing Center



Lawrence Berkeley
National Laboratory



Connecting with SSH

- **Secure Shell (ssh)**
 - Protocol version 2 since 2006
 - Encrypts all communications
 - Control
 - Prevents password “sniffing”
 - Data
 - Potential performance penalty usually not a problem



SSH Client/Server Model

- **Server**
 - sshd daemon
- **Clients**
 - ssh, scp, sftp
 - OpenSSH – Linux, Mac
 - PuTTY – Windows
 - Other clients layered on top of ssh authentication
 - Subversion (svn)
 - bbcop



SSH Authentication Models

- **password**
 - need “keyboard-interactive” method in client to support PAM (pluggable authentication modules) in server
- **key pairs**
 - private/public keys
 - password is never transmitted over network



Key Management

- Keys generated using “ssh-keygen” command
- Use “passphrase” to protect (encrypt) keys
- Private key typically kept on desktop
- Public key placed in \$HOME/.ssh/authorized_keys on remote systems
- “Agent” on desktop (typically a shell or X-server) stores private key in memory
- Be careful not to introduce newlines into keys!



Basic Key Example

```
desktop% ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/homes/dpturner/.ssh/
id_dsa): <return>
Enter passphrase (empty for no passphrase): <enter passphrase
here>
Enter same passphrase again: <enter passphrase again>
Your identification has been saved in /homes/dpturner/.ssh/
id_dsa.
Your public key has been saved in /homes/dpturner/.ssh/
id_dsa.pub.
desktop% scp .ssh/id_dsa.pub hopper.nersc.gov:~/.ssh/
authorized_keys
Password: <enter password>
desktop% ssh hopper.nersc.gov
Enter passphrase for key '/homes/dpturner/.ssh/id_dsa': <enter
passphrase>
hopper%
```



Better Key Example

- Assume public key placed in `authorized_keys`

```
desktop% ssh-agent tcsh
desktop% ssh-add
Enter passphrase for key '/homes/dpturner/.ssh/id_dsa': <enter
passphrase>
desktop% ssh hopper.nersc.gov
hopper% ssh franklin.nersc.gov
Password: <enter password>
franklin%
```



Even Better Key Example

- Use “agent forwarding”, with `.ssh/config` file
- Assume agent started on desktop, passphrase entered

```
desktop% cat .ssh/config
Host *
ForwardAgent yes
desktop% ssh hopper.nersc.gov
hopper% ssh franklin.nersc.gov
franklin% ssh carver.nersc.gov
carver%
```



Useful SSH Client Options

- **-Y**
 - Enable trusted X11 forwarding
- **-A**
 - Enable agent forwarding
- **-l <username>**
 - Username on remote system
 - Can also use “username@hostname”
- **-v/-vv/-vvv**
 - verbose/very verbose/very, very verbose mode



Password Security

- **Must be changed every 6 months**
- **Must be *at least* 8 characters long**
- **Must contain *at least* one of each of:**
 - upper-case letter
 - lower-case letter
 - numeral
 - “special” character (! @ # \$ % ^ & *)
- **Don’t use common words, names, etc.**
- **Account locked after 5 login failures**
 - Call NERSC Account Support to reset
- **DON’T SHARE PASSWORDS!**



Password Examples

- **Good**
 - j#K01vz\$ewP@!udls
- **Bad**
 - P@ssw0rd
- **My favorite method**
 1. computer security is very important for nersc users
 2. csivifnu
 3. C\$1v1fnu



Data Movement

- **Use NGF to minimize movement and reduce duplication**
 - global home, global scratch, global project
- **Use cp**
 - Within or between any NGF file system
 - Within or between `scratch` and `scratch2` on *either* Cray
- **Use scp or bbcp**
 - From either scratch on one Cray to either scratch on *other* Cray
 - To/from external locations



Secure Copy

- Uses ssh authentication
- Good for “small” (~100s of MB)
- Watch out for:
- “Chatty” dotfiles can cause silent failure
- Missing “:” results in silent *local* copy

```
scp mydata hopper:
```

- Copies mydata to home directory on Hopper

```
scp mydata hopper
```

- Copies mydata to hopper (file)



BaBar Copy (bbcp)

- **Developed for BaBar experiment at SLAC**
- **Peer-to-peer model (not client-server)**
 - Must be installed on each end
 - Easy to build
 - Available on all NERSC systems
- **Can use ssh authentication**
- **Many tuning options**
- **Good for larger files**
- **Somewhat complicated command-line**



Examples of bbcp

```
bbcp -T "ssh -x -a -oFallbackToRsh=no %I -l %U %H /usr/common/usg/bin/bbcp"  
bigfile remotesystem:
```

- **-T**
 - Target system options
- **-S**
 - Source system options
- **-Z**

```
bbcp: Accept timed out on port 5031  
bbcp: Unable to allocate more than 0 of 8 data streams.  
Killed by signal 15.
```



More Information

<http://www.nersc.gov/users/data-and-networking/connecting-to-nersc/>

<http://www.nersc.gov/users/accounts/user-accounts/passwords/>

<http://www.nersc.gov/users/data-and-networking/transferring-data/>

<http://fasterdata.es.net/fasterdata/data-transfer-tools/>

- **Use google to locate ssh tutorials!**