

# Using the St Andrews Research Computing Resource

An introduction to doing computational research  
on the cluster **kennedy**

Herbert Früchtl

# Overview

The cluster kennedy: hardware setup

Connect to a Linux computer from your desktop

Introduction to Linux

- Basic Linux commands
- The emacs and vi editors
- Transferring files between desktop and Linux server

Running calculations through SLURM

- A typical job script
- How to submit and monitor jobs

Rules of use

How to be a good citizen

Security and privacy

# The university HPC resource kennedy



110 x 32-core compute nodes  
128GB – 1.5TB memory each  
2 nodes have GPUs (Tesla V100)  
400 TB disk space

Available to all St Andrews researchers

High-priority access:  
£0.0118 per core hour

Low priority access:  
free at the point of delivery  
depending on School contributions

To apply for access, contact  
[herbert.fruchtl@st-andrews.ac.uk](mailto:herbert.fruchtl@st-andrews.ac.uk)

# Connecting Remotely

You need:

- Terminal program
- ssh client
- X11 server for graphical applications

## Linux:

You have all you need

## Mac:

Install XQuartz  
(free from Apple Store)

**Windows:** Multiple possibilities

Terminal+ssh: PuTTY, MobaXterm, ssh, Cygwin

X11 server: VcXsrv, Xming, Cygwin/X

(Cygwin is a complete Linux implementation for Windows)

# Connecting from Windows: VcXsrv and PuTTY

## Installation

### VcXsrv:

Download from

<https://sourceforge.net/projects/vcxsrv/>

and install

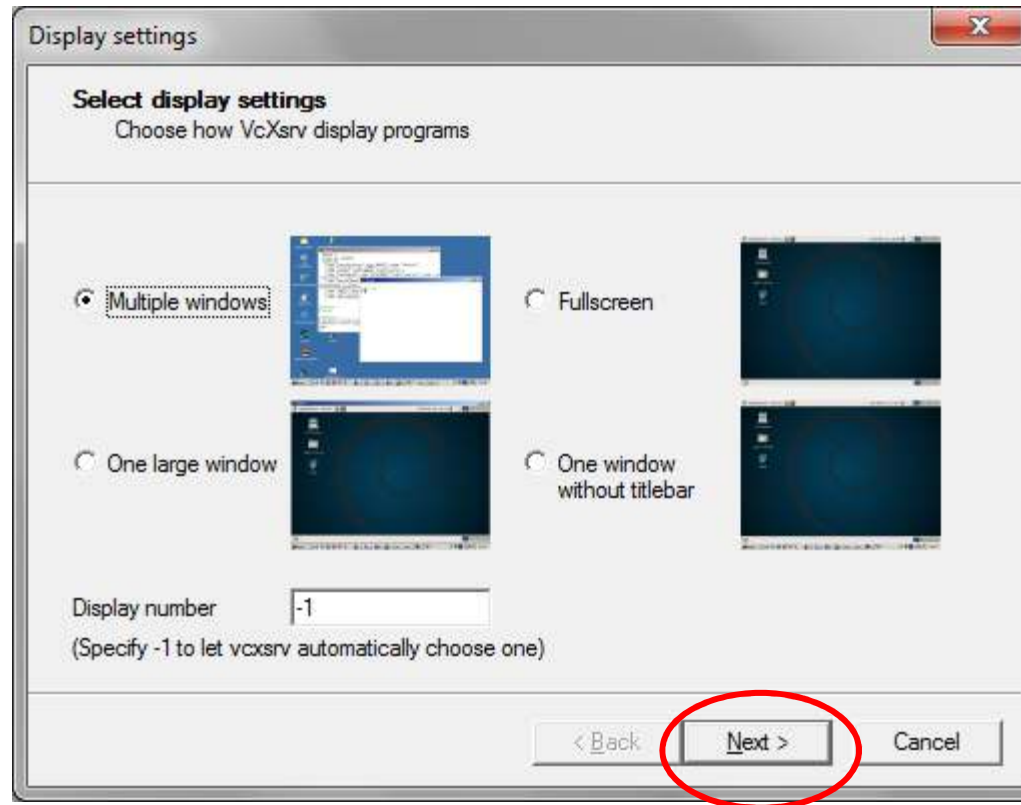
### PuTTY

Download Windows installer from

<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

and execute it

# Starting VcXsrv:

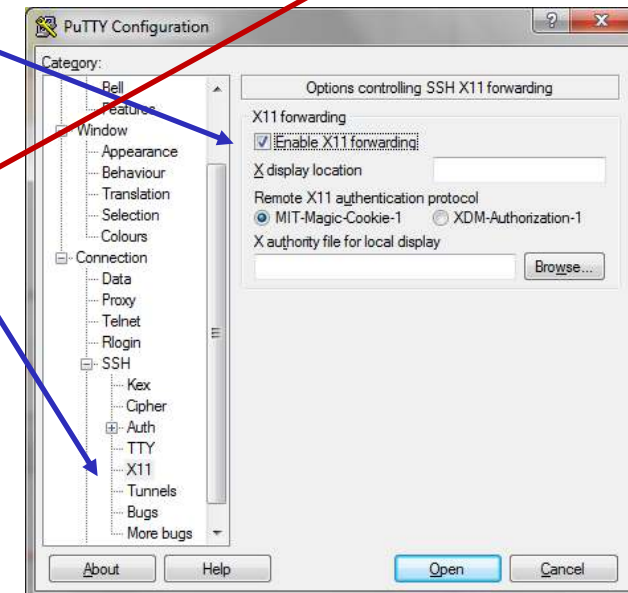
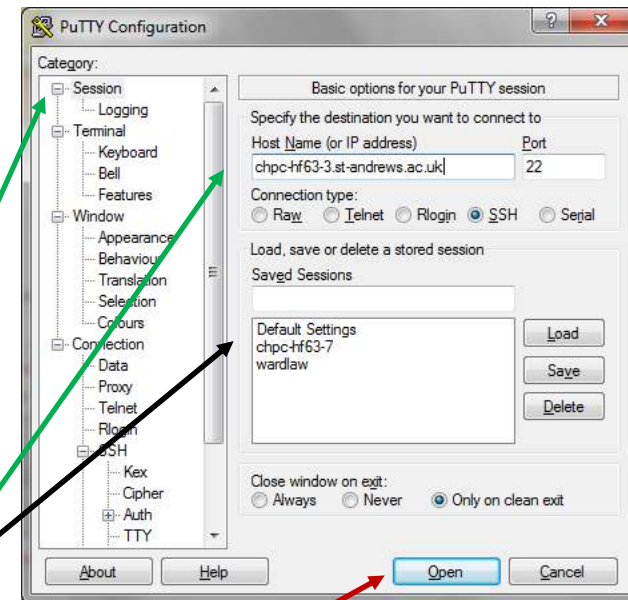


Accept all defaults  
(also on following pages)

Save when prompted, so you  
don't have to go through again

# Starting PuTTY Terminal

- Start XLaunch  
(VcXsrv executable;  
runs in background)
  - Accept defaults at first start;  
save anywhere
- Start PuTTY
  - Make sure X11 forwarding is enabled
  - Can save session settings and recall
  - Specify host in Session window
  - Start terminal



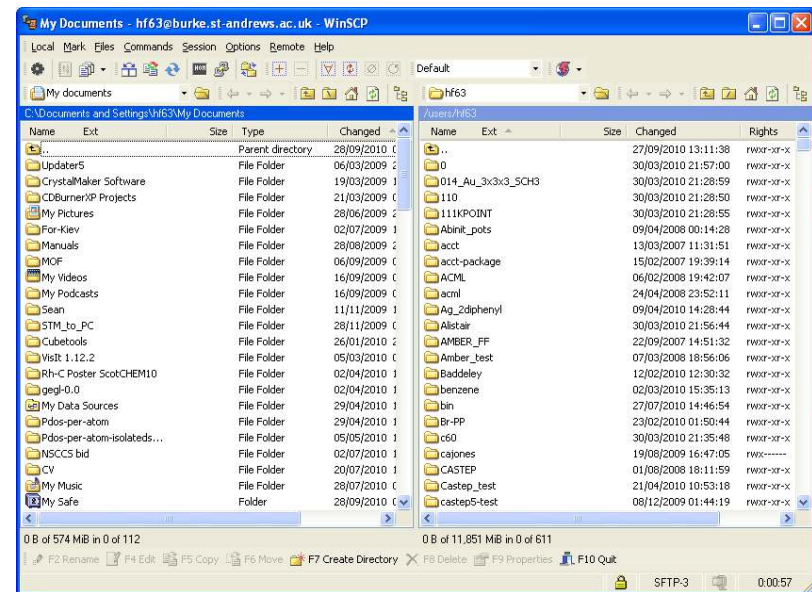
# File Transfer

Using cygwin (or Mac, or Linux) command line:

```
sftp <user>@kennedy.st-andrews.ac.uk  
login, password...  
cd <dir>  
put <file>           PC -> Linux server  
get <file>          Linux-> PC  
exit
```

Alternative on Windows: WinSCP

<http://winscp.net/eng/index.php>  
graphical user interface (drag and drop)



Convert text files (newline character differs between UNIX and DOS/Windows)

```
dos2unix <file>       Windows -> Linux  
unix2dos <file>      Linux -> Windows
```

both commands should work  
on cluster and in CYGWIN

**Not doing this can lead to weird errors!**

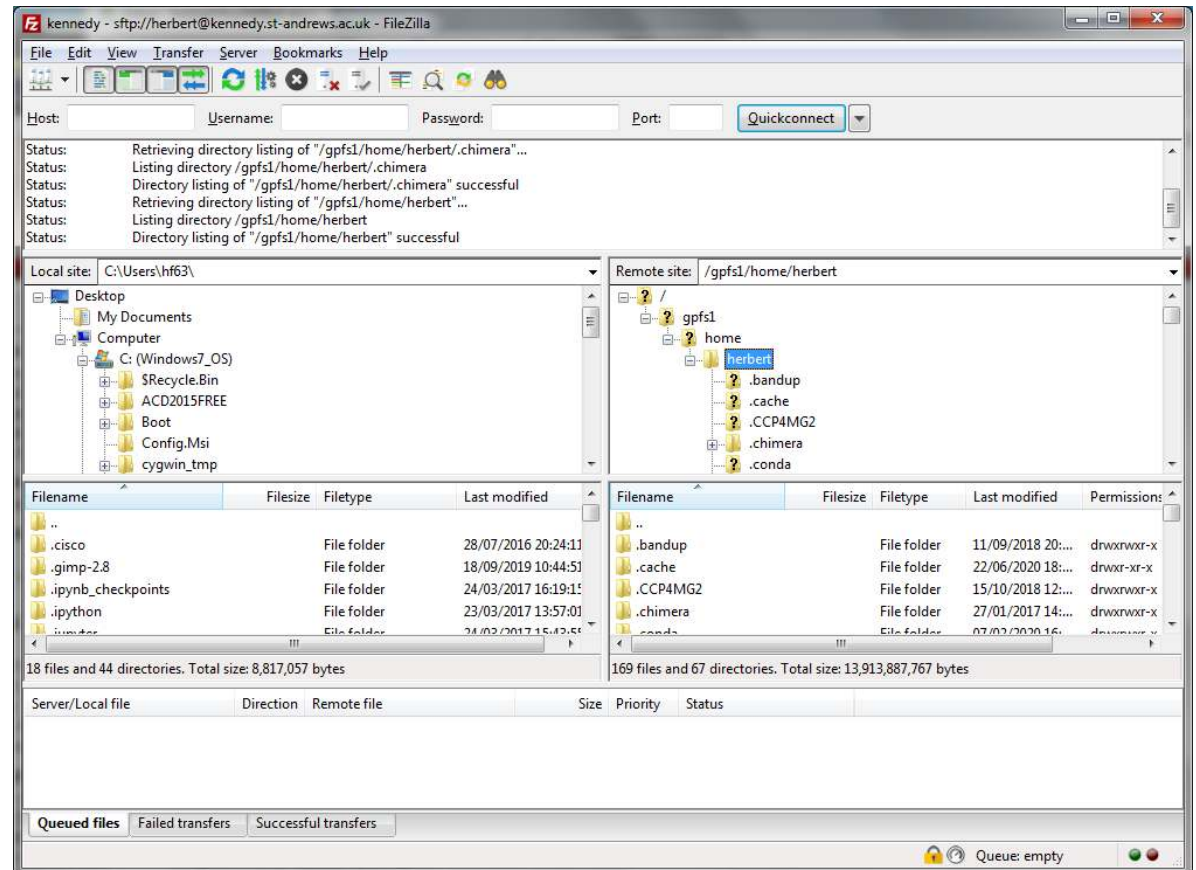


# File Transfer: FileZilla

Similar to WinSCP

Available for Windows,  
Mac and Linux

Download from  
<https://filezilla-project.org/>



# Login Security

Login to kennedy requires ssh key pair and password

## **ssh keys:**

- Private key stored on your computer. Must not be shared, sent by email or made accessible in any other way
- Public key to be installed on remote computer (kennedy). On its own does not allow login anywhere, so it can be emailed.

## **Password:**

- Requirement: at least 10 characters
- At least one special character
- Non-obvious (checked against known bad password list)

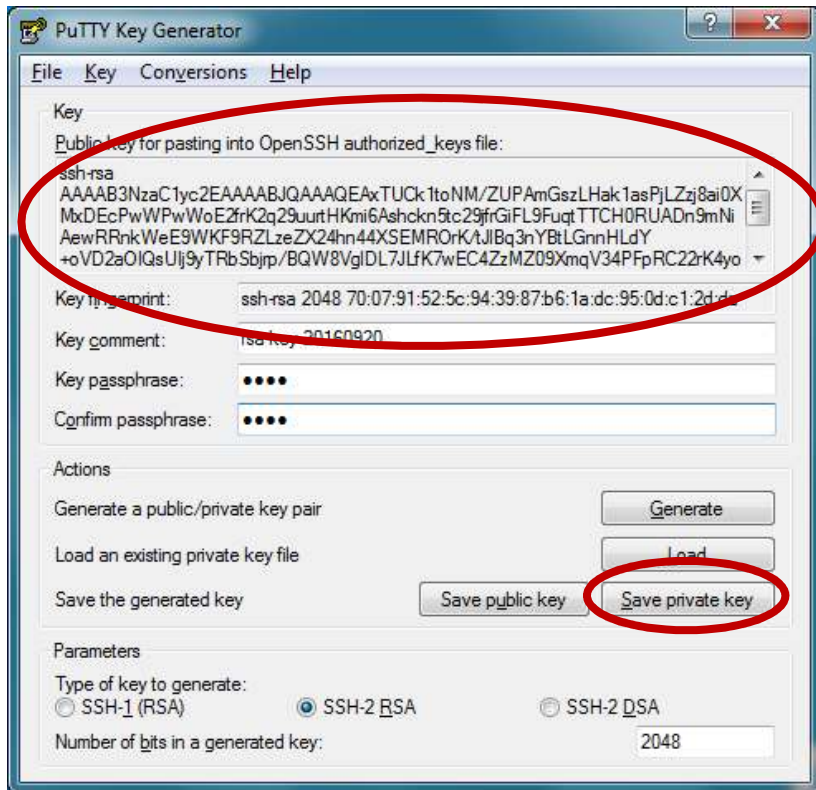
# Creating and Using ssh Keys

On Linux, Mac or Cygwin:

- Enter command  
`ssh-keygen`
- Accept all default filenames
- Choose (and remember) passphrase when prompted  
(empty passphrase allowed, because we have a separate password)
- Email file `~/.ssh/id_rsa.pub` to system administrator.  
(In Cygwin, this is `C:\cygwin64\home\\.ssh\id_rsa.pub`)  
Do **NOT** send the private key (`id_rsa` without suffix)

# Creating ssh Keys with PuTTYgen

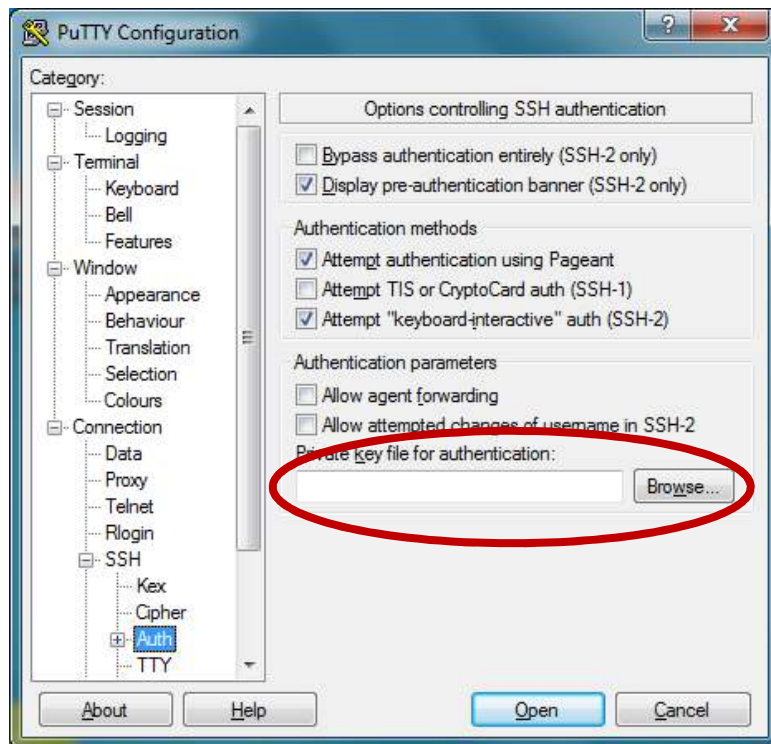
## PuTTYgen



- Start PuttyGEN (in PuTTY folder in start menu).
- Select "Generate"
- Move mouse randomly through grey area in window until finished.
- Save private key. You will need this to connect.
- Copy long string in upper window into text file.
- Email this file to sysadmin
- Do **NOT** send the private key (.ppk file)!

# Using ssh Key with PuTTY

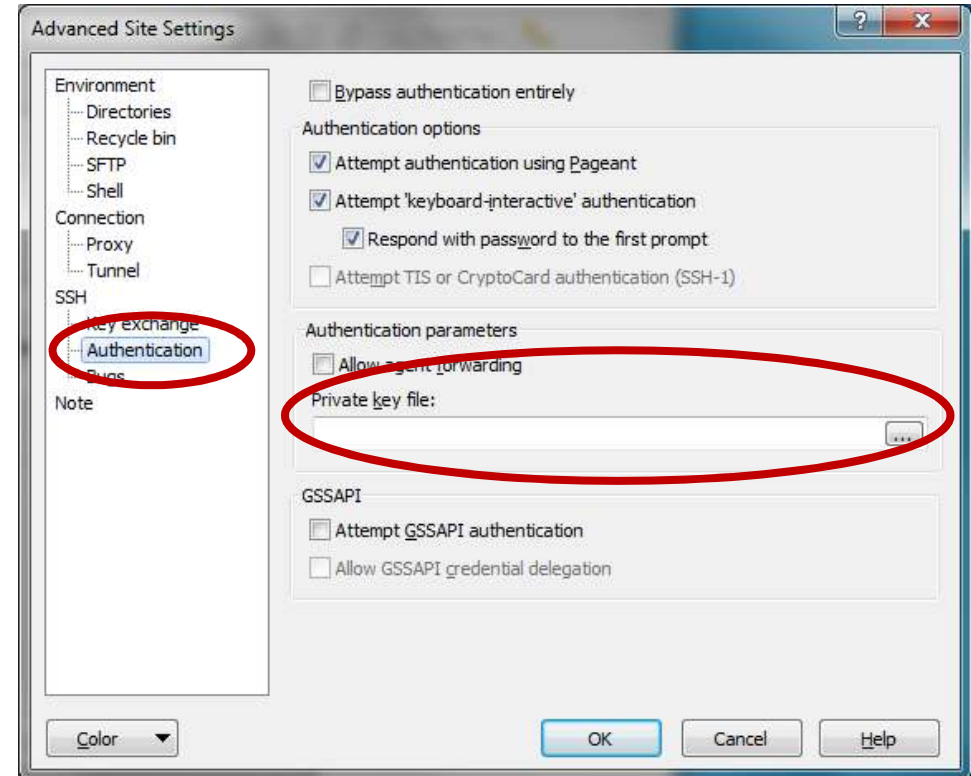
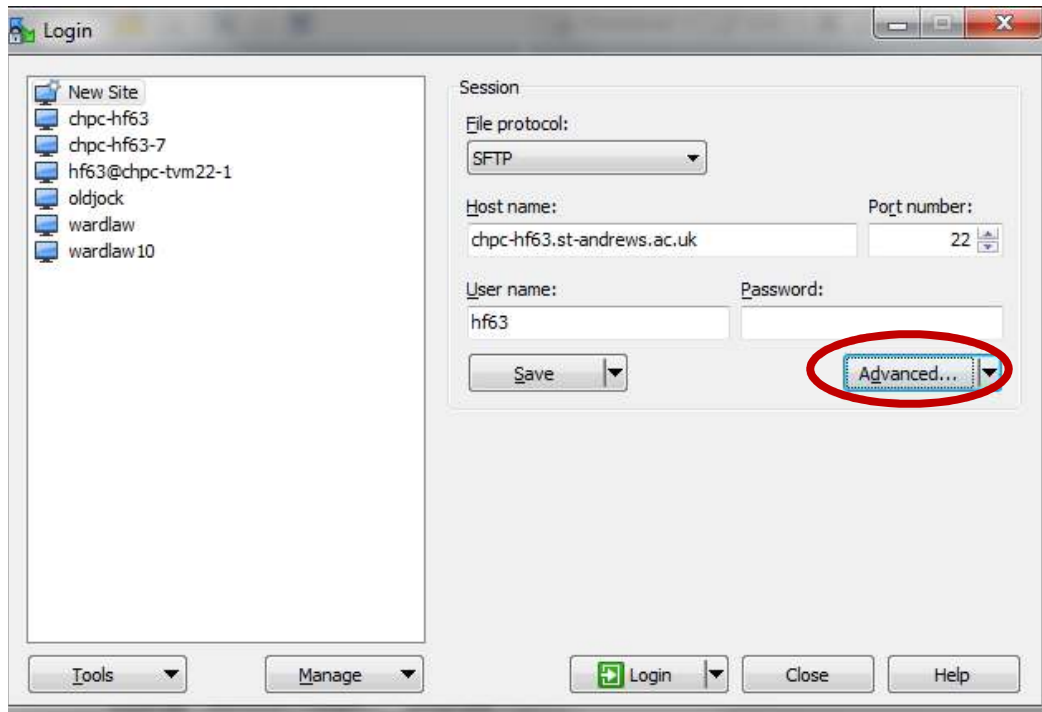
## PuTTY



- Select **Auth** tab under **Connection -> SSH**
- Use browser menu to find private key (.ppk)

You can save settings (key location and X11 enabling) in the Session tab.

# Using ssh Key with WinSCP

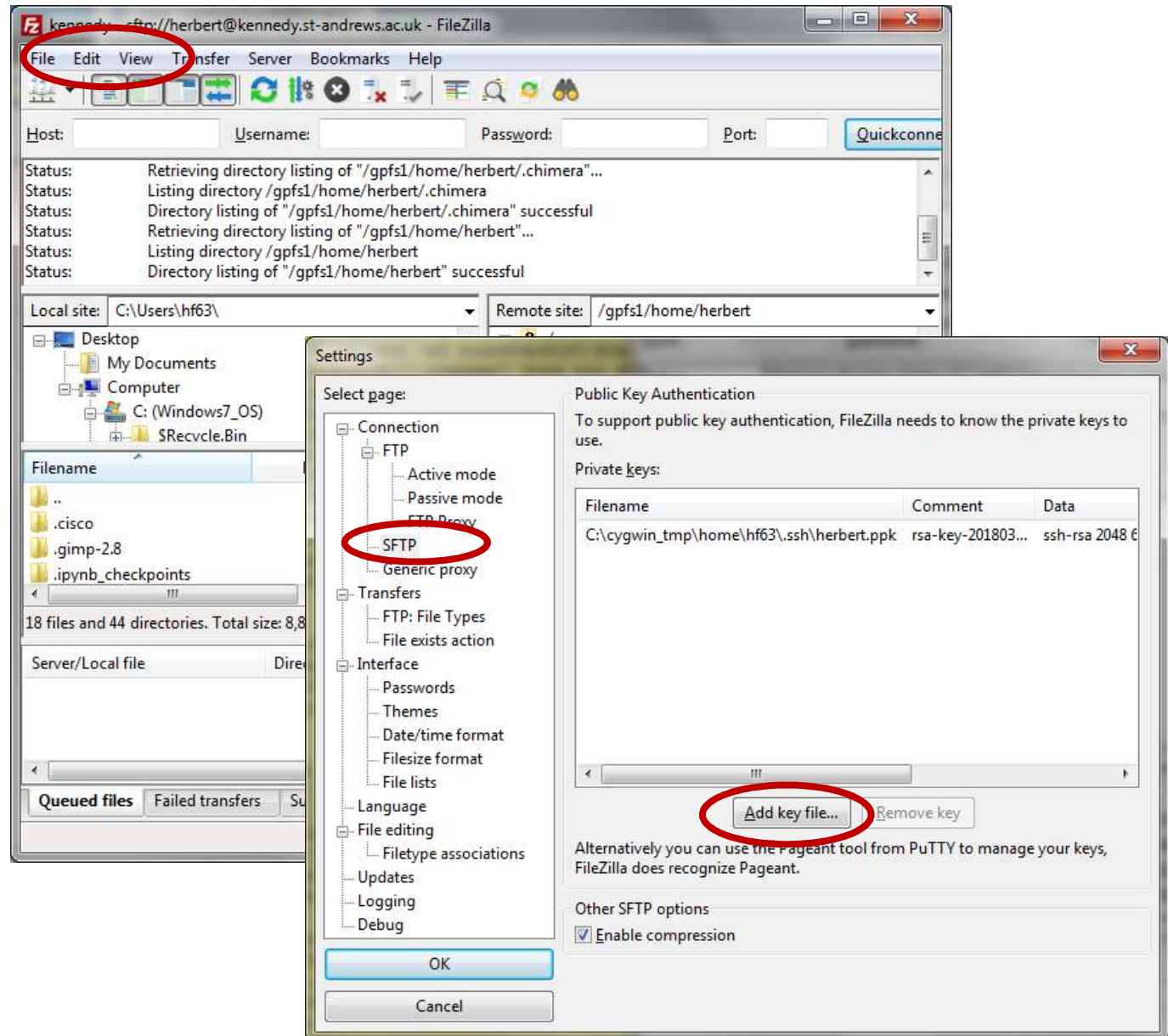


- Select **Advanced** settings
- **SSH** -> **Authentication** tab
- Use browser menu to find private key (.ppk)

# Using ssh Key with FileZilla

## Edit

- Settings...
- Connection
- SFTP
- Add key file...



# Linux commands

## **man <command>**

- manual page for <command>

## **exit**

- quit current shell (log out if this is the login shell)

## **passwd**

- change password
- see earlier for rules
- do this on your first login!



# Linux commands

**ls**

list current directory

**ls -l**

list directory in detail

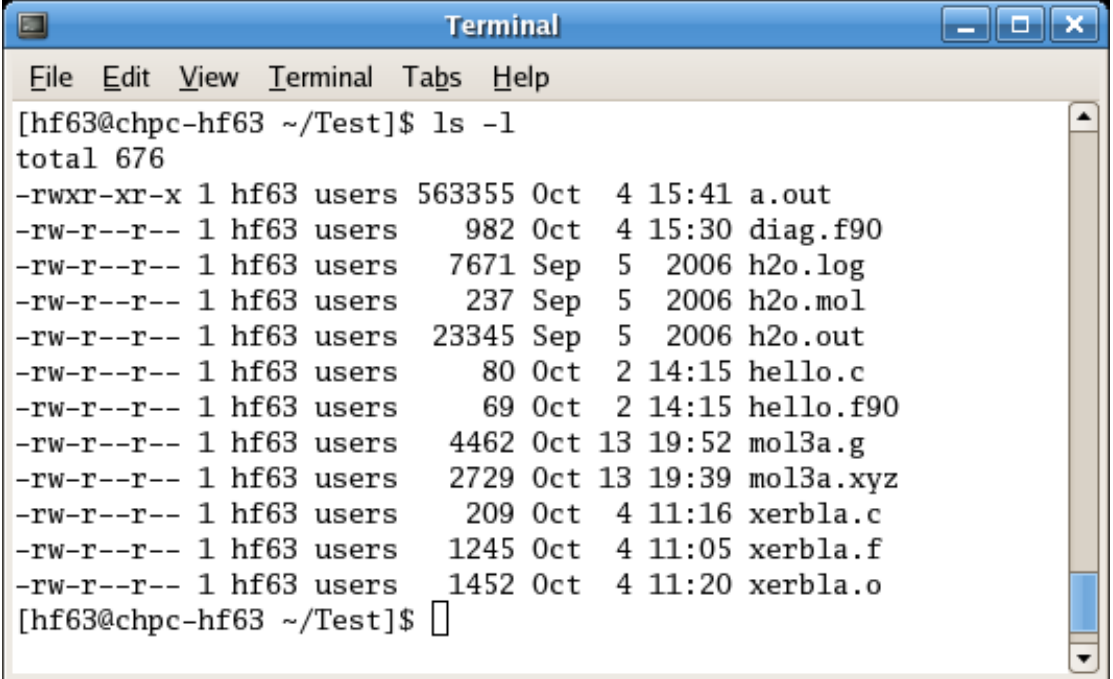
**ls -lrt** (same as **ls -l -t -r**)

list current directory in detail,  
ordered by time, in reverse order

**ls <dir>**

list directory <dir>

**man ls ...**



```
Terminal
File Edit View Terminal Tabs Help
[hf63@chpc-hf63 ~/Test]$ ls -l
total 676
-rwxr-xr-x 1 hf63 users 563355 Oct  4 15:41 a.out
-rw-r--r-- 1 hf63 users   982 Oct  4 15:30 diag.f90
-rw-r--r-- 1 hf63 users  7671 Sep  5 2006 h2o.log
-rw-r--r-- 1 hf63 users   237 Sep  5 2006 h2o.mol
-rw-r--r-- 1 hf63 users 23345 Sep  5 2006 h2o.out
-rw-r--r-- 1 hf63 users    80 Oct  2 14:15 hello.c
-rw-r--r-- 1 hf63 users    69 Oct  2 14:15 hello.f90
-rw-r--r-- 1 hf63 users  4462 Oct 13 19:52 mol3a.g
-rw-r--r-- 1 hf63 users  2729 Oct 13 19:39 mol3a.xyz
-rw-r--r-- 1 hf63 users   209 Oct  4 11:16 xerbla.c
-rw-r--r-- 1 hf63 users  1245 Oct  4 11:05 xerbla.f
-rw-r--r-- 1 hf63 users  1452 Oct  4 11:20 xerbla.o
[hf63@chpc-hf63 ~/Test]$
```

# more Linux commands

**cd** <directory>

“change directory”: change working directory (folder)

- .. parent directory (one up in the tree)
- . current directory
- / between parent and child (not \ as in Windows)

**pwd**

“print working directory”: show where we are

**mkdir** <directory>

create directory

**rmdir** <directory>

delete (empty) directory

# Still more Linux commands

**cp** <file1> <file2>

copy

**mv** <file1> <file2>

rename (“move”)

**mv** <file1> <dir>

move <file1> into directory <dir> (works also with cp)

**rm** <file>

delete <file>

**rm -r** <dir>

delete <dir> **and all subdirectories!**

# A few more useful ones...

**head <file>**

show first 10 lines of file

**head -20 <file>**

show first 20 lines (works for any number)

**tail <file>**

show last 10 lines (e.g. to check if output file is complete)

**tail -20 <file>**

guess what...

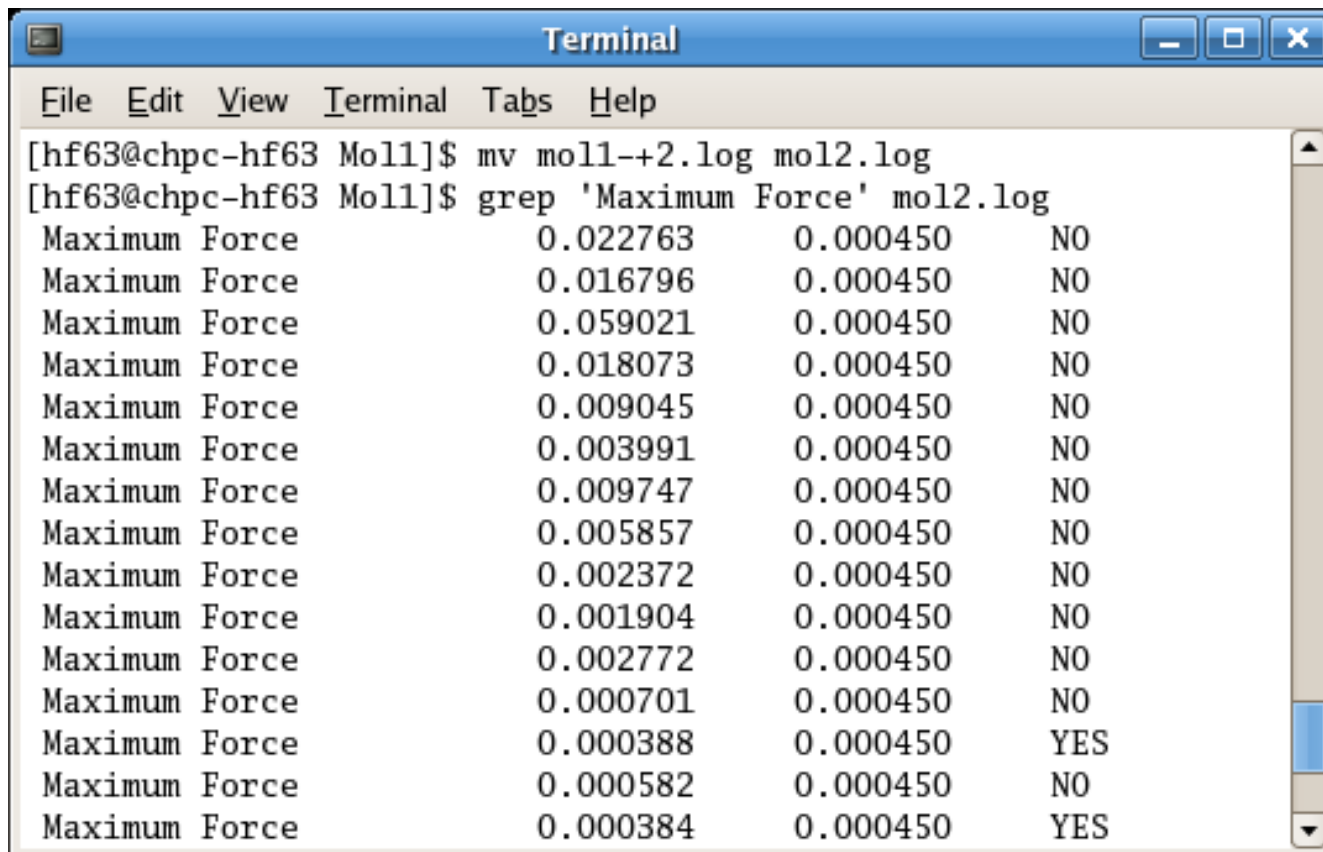
**tail -f <file>**

“follow”: keep showing newly added lines. Exit with Ctrl-c

# And the last one for now

**grep** <pattern> <file>

show all lines in <file> containing <pattern>

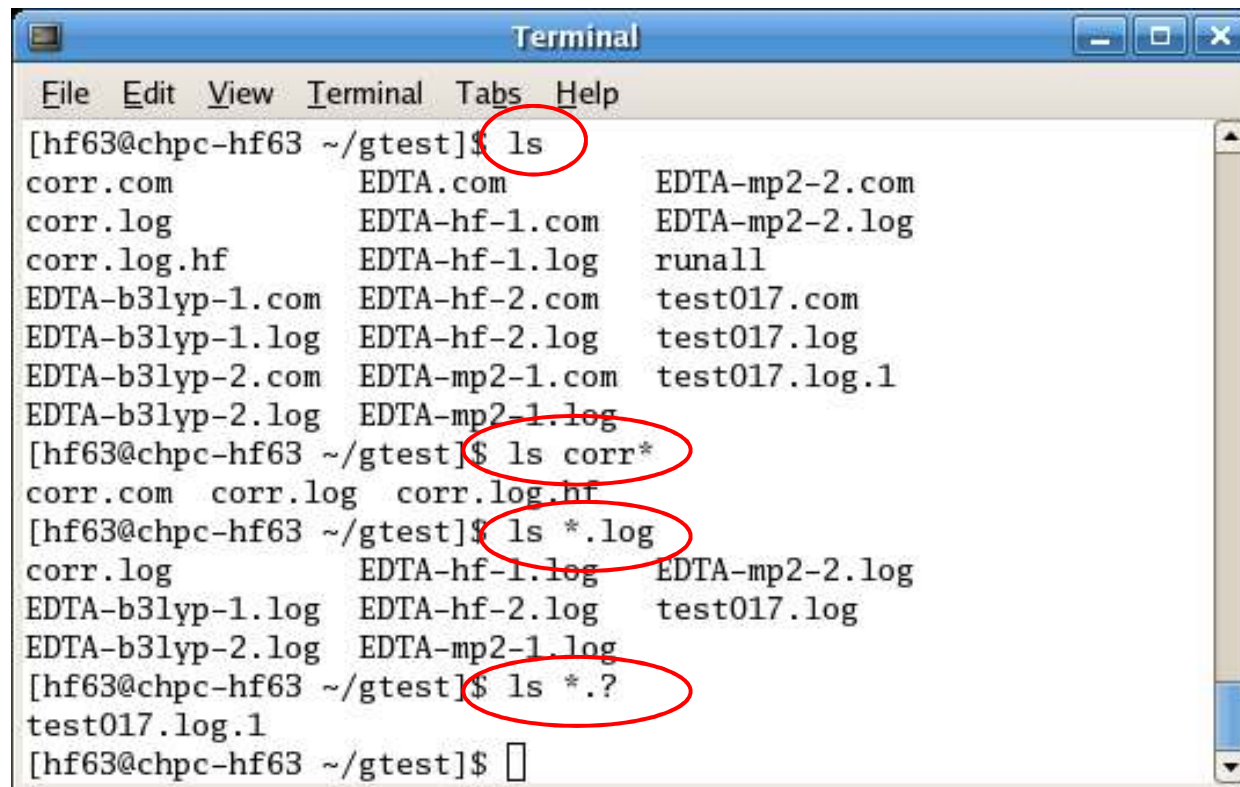


The image shows a terminal window titled "Terminal" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal displays the following commands and output:

```
[hf63@chpc-hf63 Mol1]$ mv mol1-+2.log mol2.log
[hf63@chpc-hf63 Mol1]$ grep 'Maximum Force' mol2.log
Maximum Force      0.022763      0.000450      NO
Maximum Force      0.016796      0.000450      NO
Maximum Force      0.059021      0.000450      NO
Maximum Force      0.018073      0.000450      NO
Maximum Force      0.009045      0.000450      NO
Maximum Force      0.003991      0.000450      NO
Maximum Force      0.009747      0.000450      NO
Maximum Force      0.005857      0.000450      NO
Maximum Force      0.002372      0.000450      NO
Maximum Force      0.001904      0.000450      NO
Maximum Force      0.002772      0.000450      NO
Maximum Force      0.000701      0.000450      NO
Maximum Force      0.000388      0.000450      YES
Maximum Force      0.000582      0.000450      NO
Maximum Force      0.000384      0.000450      YES
```

# Wildcards

- Most file/directory commands accept “wildcard characters”. Most commonly used ones are
- ? Any single character
- \* Any number of characters (including none)



A terminal window titled "Terminal" showing a series of file listing commands and their outputs. The prompt is [hf63@chpc-hf63 ~/gtest]. The first command is `ls`, which lists all files in the directory. The second command is `ls corr*`, which lists only files starting with "corr". The third command is `ls *.log`, which lists only files ending in ".log". The fourth command is `ls *.*`, which lists only files with at least one dot in the name. Red circles highlight the wildcards in the commands.

```
Terminal
File Edit View Terminal Tabs Help
[hf63@chpc-hf63 ~/gtest]$ ls
corr.com          EDTA.com          EDTA-mp2-2.com
corr.log          EDTA-hf-1.com     EDTA-mp2-2.log
corr.log.hf       EDTA-hf-1.log     runall
EDTA-b3lyp-1.com  EDTA-hf-2.com     test017.com
EDTA-b3lyp-1.log  EDTA-hf-2.log     test017.log
EDTA-b3lyp-2.com  EDTA-mp2-1.com    test017.log.1
EDTA-b3lyp-2.log  EDTA-mp2-1.log
[hf63@chpc-hf63 ~/gtest]$ ls corr*
corr.com corr.log corr.log.hf
[hf63@chpc-hf63 ~/gtest]$ ls *.log
corr.log          EDTA-hf-1.log     EDTA-mp2-2.log
EDTA-b3lyp-1.log  EDTA-hf-2.log     test017.log
EDTA-b3lyp-2.log  EDTA-mp2-1.log
[hf63@chpc-hf63 ~/gtest]$ ls *.*
test017.log.1
[hf63@chpc-hf63 ~/gtest]$
```

# Editing with Emacs

**emacs <file> &**

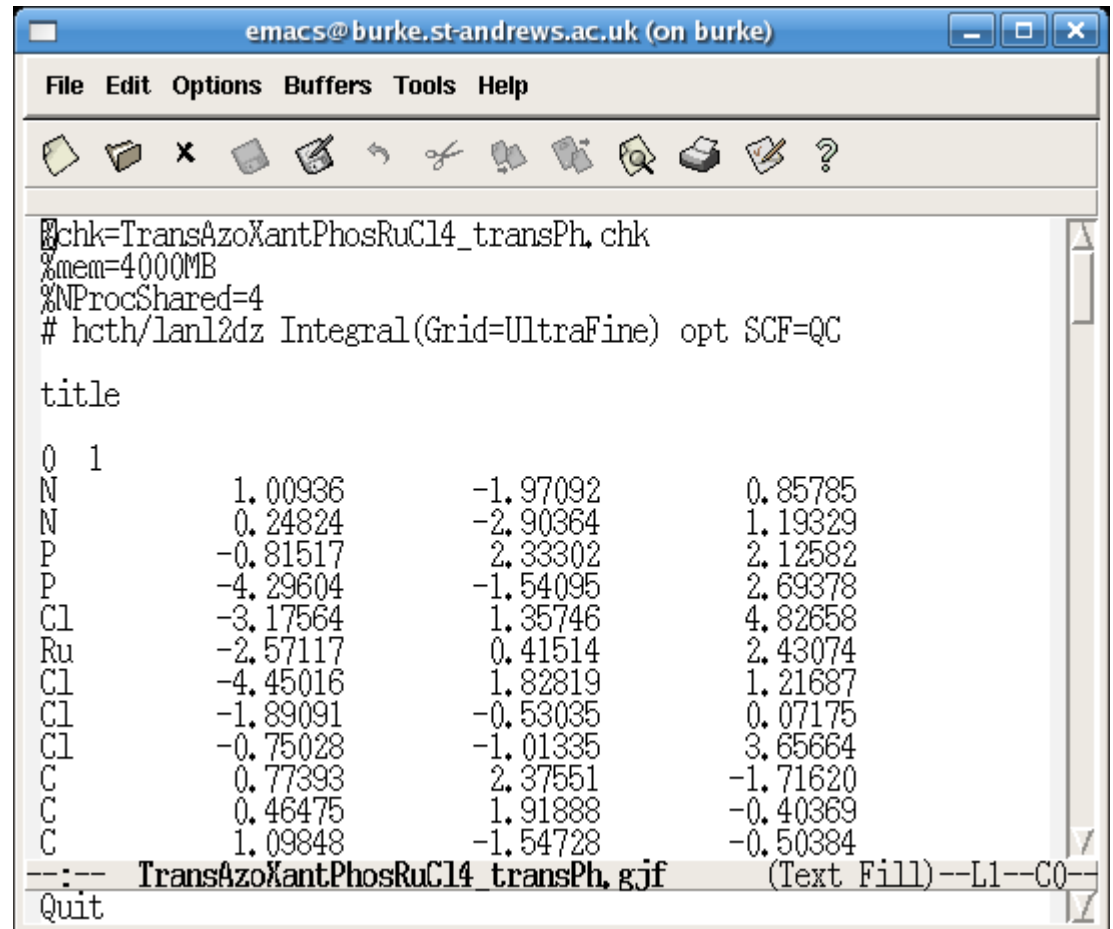
- Opens Emacs window on your desktop
- **&** executes (any) program in background, so you can go on working in the same window

Can be installed locally (Windows, Mac, Linux) and used to edit files remotely!

- Open “file”  
/ssh:<user>@<host>:<directory>/<file>  
e.g. /ssh:hf63@kennedy.st-andrews.ac.uk:testdir/test.txt
- Good if network is slow

# Using Emacs

- Menus are self-explanatory
- Try to remember some keyboard shortcuts, in case you need to use it without X client (emacs -nw)
  - Ctrl-x-s save
  - Ctrl-x-c exit
  - Ctrl-x-w save under different name



```
emacs@burke.st-andrews.ac.uk (on burke)
File Edit Options Buffers Tools Help
[Icons]
chk=TransAzoXantPhosRuCl4_transPh.chk
%mem=4000MB
%NProcShared=4
# hcth/lanl2dz Integral(Grid=UltraFine) opt SCF=QC
title
0 1
N      1.00936      -1.97092      0.85785
N      0.24824      -2.90364      1.19329
P     -0.81517      2.33302      2.12582
P     -4.29604     -1.54095      2.69378
Cl    -3.17564      1.35746      4.82658
Ru    -2.57117      0.41514      2.43074
Cl    -4.45016      1.82819      1.21687
Cl    -1.89091     -0.53035      0.07175
Cl    -0.75028     -1.01335      3.65664
C      0.77393      2.37551     -1.71620
C      0.46475      1.91888     -0.40369
C      1.09848     -1.54728     -0.50384
---:-- TransAzoXantPhosRuCl4_transPh.gjf (Text Fill)--L1--CO--
Quit
```



# Editing with vi

- Faster over a network
  - No separate window
  - A bit cryptic at first...
- start with  
`vi <file>`

## Two modes

### Command Mode: cursor keys and 1-2 letter commands

<b>x</b>	<b>delete character</b>	
<b>dd</b>	<b>delete line</b>	
<b>D</b>	<b>delete rest of line</b>	
<b>i</b>	<b>insert before cursor</b>	} enter input mode
<b>a</b>	<b>append after cursor</b>	
<b>o</b>	<b>new line below</b>	
<b>O</b>	<b>new line above</b>	
<b>:w</b>	<b>save</b>	
<b>:x</b>	<b>save and exit</b>	

### Input Mode

**Write text**

**Delete with backspace key**

**[ESC] key: back to command mode**

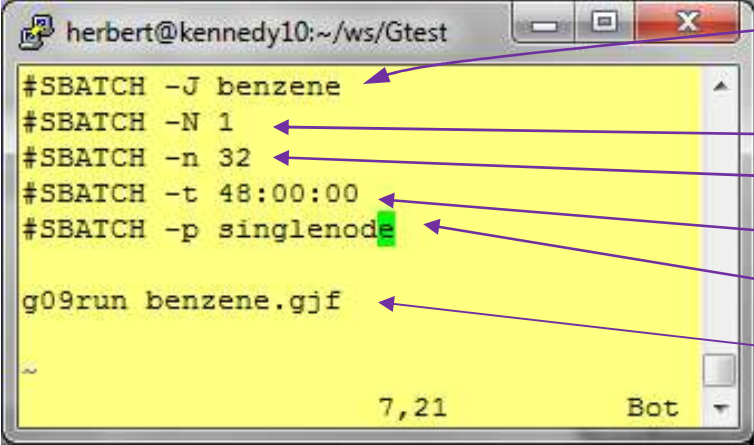
**Many tutorials and “cheat  
sheets” available**

**Google “vi editor”**

# Running SLURM Batch Jobs

Submit script to batch queue  
for execution when compute  
nodes are available

Job script with SLURM  
directives at the start:



The image shows a terminal window with a yellow background. The window title is 'herbert@kennedy10:~/ws/Gtest'. The content of the terminal is as follows:

```
#SBATCH -J benzene
#SBATCH -N 1
#SBATCH -n 32
#SBATCH -t 48:00:00
#SBATCH -p singlenode

g09run benzene.gjf

~
7,21 Bot
```

Annotations on the left side:

- SLURM directives {
- Commands to execute when job starts {

Annotations on the right side:

- Name of job
- Request one node
- Use 32 cores
- Run for 48h max
- Run in singlenode queue
- Script to run Gaussian

# Queues and Resources

Queues (“partitions” in SLURM):

- singlenode: requesting one node (32 cores) or less
- parallel: multiple nodes for parallel calculations
- gpu: one (or possibly both) of the gpu nodes
- Requesting more memory in parallel or singlenode queue:

```
#SBATCH --mem=196GB
```

(most nodes have 128GB, but a few have 196, 392 or 512GB;  
one very-large memory node has 1.5TB)

Requesting GPU nodes:

```
#SBATCH -p gpu
```

```
#SBATCH --gres=gpu:2
```

# Managing jobs

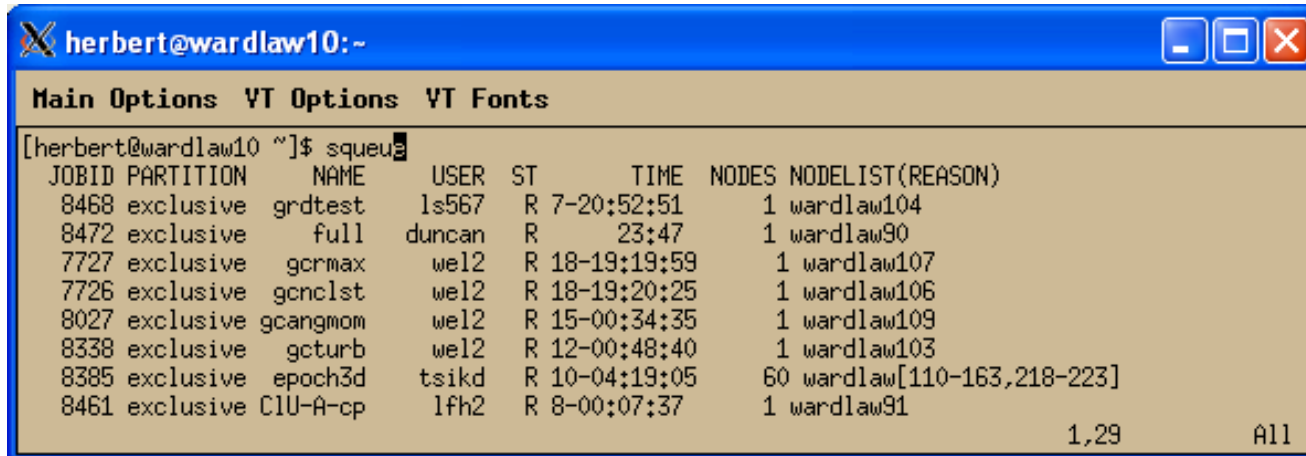
**sbatch <jobscript>**

submit job

**queue**

show running or queued jobs

**queue -u <user>** shows only your jobs



```
herbert@wardlaw10:~  
Main Options VT Options VT Fonts  
[herbert@wardlaw10 ~]$ queue  
JOBID PARTITION NAME USER ST TIME NODES NODELIST(REASON)  
8468 exclusive grdtest ls567 R 7-20:52:51 1 wardlaw104  
8472 exclusive full duncan R 23:47 1 wardlaw90  
7727 exclusive gcrmax we12 R 18-19:19:59 1 wardlaw107  
7726 exclusive gcnc1st we12 R 18-19:20:25 1 wardlaw106  
8027 exclusive gcangmom we12 R 15-00:34:35 1 wardlaw109  
8338 exclusive gcturb we12 R 12-00:48:40 1 wardlaw103  
8385 exclusive epoch3d tsikd R 10-04:19:05 60 wardlaw[110-163,218-223]  
8461 exclusive ClU-A-cp lfh2 R 8-00:07:37 1 wardlaw91  
1,29 All
```

**scancel <jobnumber>**

delete job

# Privacy and Data Protection

UNIX File permissions:

user (=owner)

group (same group as owner)

others (anybody else)

read

write

execute (file) or access (dir)

To change:

**chmod go-rwx <file>**

(remove all permissions from your group and others)

**chmod g+r <file>**

(give your group read permission)

**man chmod ...**

Root user can read (or write) anything!

```
hf63/bin> ls -l
total 10868
-rwxr-xr-x 1 hf63 frucht1 892082 2010-07-19 15:12 castep-spectrum
-rwxr-xr-x 1 hf63 frucht1 1213794 2010-06-14 00:38 cryst
-rwxr-xr-x 1 hf63 frucht1 4736220 2010-07-27 13:40 gdis
-rw-r--r-- 1 hf63 frucht1 40197 2010-07-27 13:41 gdis.elements
-rwxr-xr-x 1 hf63 users 11306 2006-11-06 15:04 make_p
-rwxr-xr-x 1 hf63 users 3468709 2007-02-27 17:53 MOPAC2007.exe
-rwxr-xr-x 1 hf63 users 220 2008-04-22 17:57 my_vasp_ether
-rwxr-xr-x 1 hf63 frucht1 51143 2010-02-04 15:35 NPtcp
-rwxr-xr-x 1 hf63 users 696472 2008-09-18 23:31 xyz-scan
hf63/bin>
```

# Security

## Passwords

- Don't share. Get your own account
- Make it difficult to guess (no names, dictionary words, birth dates; use special characters)
- Change if compromised

## ssh keys

- Keep private key secure

Don't run unknown programs

Don't log in from a PC that may have viruses

# Installing Software

## Compiling from source

- Intel and gnu compilers available
- See website or example batch jobs for path and environment settings

## CONDA

- Command **install-conda** will install individual setup in `/gpfs1/apps/conda/<user>/conda`
- Sometimes executable flags are not set in installation or update. If you encounter “command not found” errors, execute  
`chmod a+x /gpfs1/apps/conda/$USER/conda/bin/*`  
and, if using CONDA environments  
`chmod a+x /gpfs1/apps/conda/$USER/conda/envs/*/bin/*`

# How to be a good citizen

Do not flood the computer with jobs

If submitting large numbers of jobs, queue them behind each other:

**sbatch -d afterany:<jobid> <jobscrip>**

job will not start running before <jobid> finishes

Do not fill the home partition

no more than 20GB in home directory

run large calculations in /scratch/<group>/<user>

Or else...





# Further Information

Example batch jobs on kennedy in  
`/usr/local/examples`

The St Andrews HPC website  
<https://www.st-andrews.ac.uk/high-performance-computing/>

# How to get an Account or Advice

Contact me!

hf63@st-andrews.ac.uk

St Andrews HPC website

<https://www.st-andrews.ac.uk/high-performance-computing/>