This workshop will help you get started using HPC on the University of Exeter flagship HPC system.

You will learn

* how to browse the installed software
* how to submit work to the HPC
* how software is designed for HPC

Contents

1 Logging in

1.1 Before you start

- You must be on the campus network via Wifi[1], LAN[2], or VPN[3].
- You will need to be a member of a *research project* to complete this workshop. Group sessions will use **Research_Project-IscaTraining** but individuals using this document outside of group sessions will receive a research project from their project instigator or supervisor which they should use instead.
- If you have further questions please refer to our Wiki[4].

[1] `https://as.exeter.ac.uk/it/network/wifi/`
[2] `https://as.exeter.ac.uk/it/equipmentandsoftware/`
[3] `https://as.exeter.ac.uk/it/network/vpn/`
[4] `http://login.isca.ex.ac.uk:8080/wiki`

1.2 Linux

If you are using a Linux or Apple Mac computer then you almost certainly have an SSH client available. To access ISCA, open a terminal and type the command below, replacing *myUserName* with your actual University username.

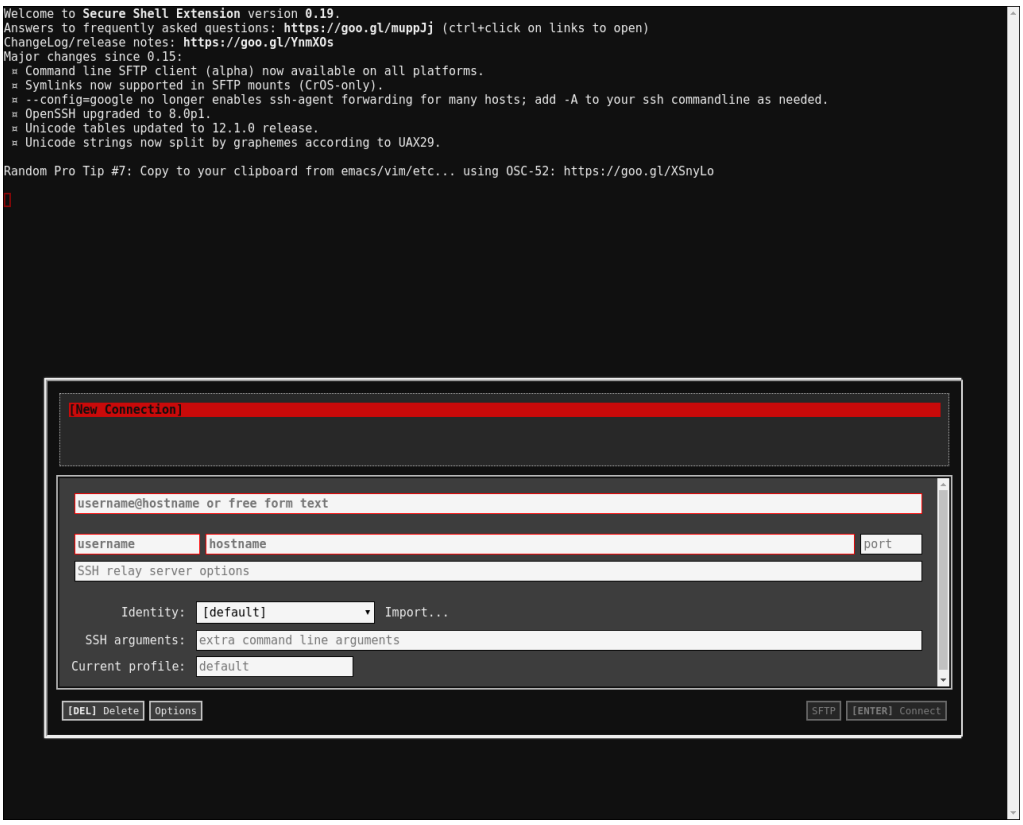
Type

```
ssh myUserName@login.isca.ex.ac.uk
```

It will prompt you for your password. For security, the characters will not be shown on the screen while you type. When you are finished typing your password, press [Enter] to log in to Isca HPC.

1.3 Windows

1. Go to https://www.google.com/chrome/
2. Click *Extensions*
3. Search for **Secure Shell Extension, Offered by: Google Secure Shell Developers**
4. Click *Add to Chrome*
5. Click the extension icon in the menu bar
6. Fill in the dialog below with your University username and the hostname **login.isca.ex.ac.uk**



Click connect then it will prompt you for your password. For security, the characters will not be shown on the screen while you type. When you are finished typing your password, press [Enter] to log in to Isca HPC.

2 Message-of-the-day

As soon as you are connected you will see the message-of-the-day which may include important information about current ISCA events. usually it will look like the message below.

```
*************************************************************************
*                 UNIVERSITY OF EXETER ISCA HPC SERVICE                 *
*                                                                       *
* This computer system is operated on behalf of the University of Exeter. *
* Only authorised users are entitled to connect and/or login to this    *
* computer system. If you are not sure whether you are authorised, then *
* you are not and should DISCONNECT IMMEDIATELY.                        *
*                                                                       *
* For user documentation please visit                                   *
*    http://login.isca.ex.ac.uk:8080/wiki                               *
* Email isca-support@exeter.ac.uk with any direct support issues        *
*                                                                       *
*              CURRENT TIMEOUT FOR INACTIVITY IS 24 HOURS               *
*                                                                       *
*************************************************************************
```

Who is allowed to connect or login to the HPC?

Where is the documentation?

Who do you call?

How long can you stay logged in for?

3 Command prompt

1. **pwd** tells you what directory you are currently in

   What directory are you in?

2. **cd** changes your current directory

   Type

   ```
   cd /tmp
   ```

   Type

   ```
   pwd
   ```

3. **ls** lists the files in your current directory

   Type

   ```
   ls
   ```

   Type

   ```
   ls -l
   ```

4. **nano** is a text file editor recommended for beginners

   Type

   ```
   nano myfile.txt
   ```

   Type some text

   Press

   [Ctrl]+o, [Enter], [Ctrl]+x

5. **What happens if many people edit the same file?**

6. **less** views text files

   Type

   ```
   less myfile.txt
   ```

7. **mkdir** creates directories

   Type

   ```
   mkdir $USER
   ```

   What directory was created? (Tip: use the **ls** command)

8. **cp** copies files and directories

   Type

   ```
   cp myfile.txt $USER.txt
   ```

   What has this command done? (Tip: use the **ls** command)

   Type

   ```
   cp myfile.txt $USER
   ```

   What has this command done?

   Choose another user file and type

   ```
   cp [other-file] $USER
   ```

   What has this command done?

9. **mv** moves or renames files and directories

   Type

   ```
   mv $USER.txt $USER-renamed.txt
   ```

   Type

   ```
   mv $USER-renamed.txt $USER
   ```

   Choose another user file and type

   ```
   mv [other-file] $USER
   ```

   Why could you not move the other user file?

10. **help** gives you some basic assistance

    Type

    ```
    help variables
    ```

    Which directories are searched when looking for commands?

11. **man** lets you read the manual

    Type

    ```
    man man
    ```

    What would you type to find information about the **ls** command?

4 Installed software

1. To see the complete list of available and installed modules please refer to http://login.isca.ex.ac.uk:8080/modules.txt

2. If you know the name of the software then you can search on the command line

   Type

   ```
   module avail Workshop
   ```

3. Software is organised into groups called **toolchains** to ensure compatibility. See http://login.isca.ex.ac.uk:8080/toolchains.txt for a description of what software each toolchain comprises.

   How many toolchains is **Workshop** available in?

4. If the software you wish to use is not on the list, or the version for your desired toolchain is not available, please go to the full list of HPC software modules at http://login.isca.ex.ac.uk:8080/modulesdescriptions.html and click the software title to send an email to ISCA support.

5. To load a module use the **module load** command.

   Type

   ```
   module load Workshop/1.0.0-foss-2019b
   ```

6. To see which modules are currently loaded run the **module list** command

7. You can switch between different modules using **module switch**

   Type

   ```
   module switch Workshop/2.0.0-foss-2019b
   ```

8. To unload a module run the **module unload** command.

   Type

   ```
   module unload Workshop/2.0.0-foss-2019b
   ```

5 Submitting work

Work is allocated to HPC resources under the control of a scheduling system called **Moab** which tries to make sure that everyone gets a fair share of the resources while also ensuring the HPC is used to the maximum of its ability. When a suitable timeslot is decided by **Moab**, the resource manager **Torque** receives the allocated work from **Moab** and runs it.

1. Obtain an interactive session on the serial test queue

   Type

   ```
   msub -I -l 60:0 -q stq -A Research_Project-IscaTraining
   ```

   What is the job id?

2. Wait for the job to start

   Where is the job running?

3. Show the queue

   Type

   ```
   showq -u $USER
   ```

   Why is the job id different in showq?

4. Check your job

   Type

   ```
   checkjob [job id]
   ```

   How long do you have this node reserved for?

6 A simple program

6.1 R

Type

```
nano myscript.R
```

Enter the following text:

```
C<-2
A<-3
while(A<91999){
  B<-3
  while(B<A/2) if(A%%B) B<-B+2 else break
  if(B>=A/2) C<-C+1
  A<-A+2
}
print(C)
```

Press

[Ctrl]+[o], [Enter], [Ctrl]+x

Type

```
module load R
```

Type

```
time R -f myscript.R
```

How long did the calculation take?

Type

```
nano mympiscript.R
```

Enter the following text:

```
library("Rmpi")
D<-function(){
  P<-mpi.comm.size()-1
  R<-mpi.comm.rank()-1
  C<-0
  A<-3+2*R
  while(A<91999){
    B<-2
    while(B<A/2) {
      if(A%%B) B=B+1 else break
    }
    if(B>=A/2) C=C+1
    A<-A+2*P
  }
  return(C)
}
mpi.spawn.Rslaves(nslaves=4,needlog=FALSE)
mpi.bcast.Robj2slave(D)
mpi.bcast.cmd(C<-D())
mpi.bcast.cmd(mpi.gather.Robj(C))
C<-2
x<-mpi.gather.Robj(C)
print(sum(x))
mpi.close.Rslaves()
mpi.exit()
```

Press

[Ctrl]+o, [Enter], [Ctrl]+x

How much quicker do you think it will be?

Type

```
time R -f mympiscript.R
```

How much quicker was it?

6.2 Python

Type

```
nano myscript.py
```

Enter the following text:

```
(C,A)=(2,3)
while A<91999:
    B=3
    while B<A/2:
        if A%B:
            B=B+2
        else:
            break
    if B>=A/2:
        C=C+1
    A=A+2
print(C)
```

Press

[Ctrl]+o, [Enter], [Ctrl]+x

Type

```
module load Python/3.6.6-foss-2018b
```

Type

```
time python myscript.py
```

How long did the calculation take?

Type

```
nano mympiscript.py
```

Enter the following text:

```
from mpi4py import MPI
M = MPI.COMM_WORLD
P = M.Get_size()
R = M.Get_rank()
(C,A)=(0,3+2*R)
while A<91999:
    B=3
    while B<A/2:
        if A%B:
            B=B+2
        else:
            break
    if B>=A/2:
        C=C+1
    A=A+2*P
x=M.gather(C,root=0)
if(0==R):
    print(2+sum(x))
```

Press

[Ctrl]+o, [Enter], [Ctrl]+x

How much quicker do you think it will be?

Type

```
time mpirun -np 4 mympiscript.py
```

How much quicker was it?

6.3 C

Type

```
nano myprogram.c
```

Enter the following text:

```
#include <stdio.h>
void main() {
  int A,B,C=2;
  for(A=3;A<91999;(C+=(B>=(A/2)),A+=2)){
    for(B=3;B<(A/2);B+=2) if(!(A%B)) break;
  }
  printf("%d\n",C);
}
```

Press

[Ctrl]+o, [Enter], [Ctrl]+x

Type

```
module load foss/2019a
```

Type

```
gcc myprogram.c
```

Type

```
time ./a.out
```

How long did the calculation take?

Type

```
nano mympiprogram.c
```

Enter the following text:

```
#include <mpi.h>
#include <stdio.h>
int main (){
  int A,B,C=0,D,P,R;
  MPI_Init(0,0);
  MPI_Comm_size(MPI_COMM_WORLD,&P);
  MPI_Comm_rank(MPI_COMM_WORLD,&R);
  for(A=3+(2*R);A<91999;(C+=(B>=(A/2)),A+=(2*P))){
    for(B=3;B<(A/2);B+=2) if(!(A%B)) break;
  }
  MPI_Reduce(&C,&D,1,MPI_INT,MPI_SUM,0,MPI_COMM_WORLD);
  if(R==0) printf("%d\n",2+D);
  MPI_Finalize();
  return 0;
}
```

Press

[Ctrl]+o, [Enter], [Ctrl]+x

How much quicker do you think it will be?

Type

```
gcc mympiprogram.c -lmpi
```

Type

```
time mpirun -np 4 ./a.out
```

How much quicker was it?

7 Appendix

7.1 Glossary

- CLI: Command Line Interface. Interactivity with the computer on a text basis (no graphics).
- HPC: High Performance Computing. Normally provided by a cluster of cooperative computers or a purpose built supercomputer.
- ISCA: The University's flagship HPC facility. It is comprised of over 400 cooperative computers.
- SSH: Secure SHell. It is the present industry standard for remote CLI sessions and it can also support X11 windows.
- VNC: Virtual Network Computing. Allows display of a remote desktop over a trusted network.
- X11: The standard linux windowing system. Incompatible with standard Windows OS for which you should use VNC instead.