CTE-ARM User's Guide

Barcelona Supercomputing Center

Copyright © 2017 BSC-CNS

December 30, 2020

Contents

1 Introduction **2**

2 System Overview **2**
- 2.1 Current basic software stack . . . 2

3 Connecting to CTE-ARM **2**
- 3.1 Password Management . . . 2
- 3.2 Transferring files . . . 3
- 3.3 Active Archive Management . . . 4
- 3.4 Repository management (GIT/SVN) . . . 5

4 File Systems **5**
- 4.1 Root Filesystem . . . 6
- 4.2 Parallel Filesystems . . . 6

5 Running Jobs **6**
- 5.1 Job Queues . . . 6
- 5.2 Submitting Jobs . . . 7
- 5.3 Interactive Sessions . . . 7
- 5.4 Job Directives . . . 7
- 5.5 Standard output/error management . . . 8
- 5.6 Examples . . . 8

6 Compiling Software **9**

7 Getting help **9**
- 7.1 Frequently Asked Questions (FAQ) . . . 9

8 Appendices **10**
- 8.1 SSH . . . 10
- 8.2 Transferring files . . . 11
- 8.3 Using X11 . . . 12
- 8.4 Requesting and installing a .X509 user certificate . . . 13

1 Introduction

This user's guide for the CTE-ARM cluster is intended to provide the minimum amount of information needed by a new user of this system. As such, it assumes that the user is familiar with many of the standard features of supercomputing as the Unix operating system.

Here you can find most of the information you need to use our computing resources and the technical documentation about the machine. Please read carefully this document and if any doubt arises do not hesitate to contact us (Getting help (chapter 7))

2 System Overview

CTE-ARM is a supercomputer based on ARM processors by Fujitsu (FX1000). It provides high performance, high scalability, and high reliability, as well as one of the world's highest levels of ultra-low power consumption. Its theoretical peak performance is 648.8TFLOPS (double precision) and its total amount of memory is 6TiB, distributed among nodes (32GB/node). There are 2 login nodes and 192 computing nodes. The login nodes have the CPU Intel(R) Xeon(R) Silver 4216 CPU @ 2.10GHz and 256 GB of main memory. Bear in mind that login nodes are x86_64 while computing nodes are ARM, therefore cross-compilation is needed. Each computing node has the following configuration:

- A64FX CPU (Armv8.2-A + SVE) @ 2.20GHz (4 sockets and 12 CPUs/socket, total 48 CPUs per node)
- 32GB of main memory HBM2
- Single Port Infiniband EDR
- TofuD network

It uses ARMv8.2-A Scalable Vector Extension (SVE) SIMD instruction set with 512-bit vector implementation.

2.1 Current basic software stack

The current software stack for computing nodes is: - Red Hat Enterprise Linux Server 8.1 (Ootpa) - GCC/8.3.1

3 Connecting to CTE-ARM

You can connect to CTE-ARM using two public login nodes. Please note that only incoming connections are allowed in the whole cluster. The logins are:

```
armlogin1.bsc.es
armlogin2.bsc.es
```

This will provide you with a shell in the login node. There you can compile and prepare your applications.

You must use Secure Shell (ssh) tools to login into or transfer files into the cluster. We do not accept incoming connections from protocols like telnet, ftp, rlogin, rcp, or rsh commands. Once you have logged into the cluster you cannot make outgoing connections for security reasons.

3.1 Password Management

In order to change the password, you have to login to a different machine (dt01.bsc.es). This connection must be established from your local machine.

```
% ssh -l username dt01.bsc.es

username@dtransfer1:~> passwd
Changing password for username.
Old Password:
New Password:
```

```
Reenter New Password:
Password changed.
```

Mind that the password change takes about 10 minutes to be effective.

3.2 Transferring files

There are two ways to copy files from/to the Cluster:

- Direct scp or sftp to the login nodes
- Using a Data transfer Machine which shares all the GPFS filesystem for transferring large files

Direct copy to the login nodes.

As said before no connections are allowed from inside the cluster to the outside world, so all scp and sftp commands have to be executed from your local machines and never from the cluster. The usage examples are in the next section.

On a Windows system, most of the secure shell clients come with a tool to make secure copies or secure ftp's. There are several tools that accomplish the requirements, please refer to the Appendices (chapter 8), where you will find the most common ones and examples of use.

Data Transfer Machine

We provide special machines for file transfer (required for large amounts of data). These machines are dedicated to Data Transfer and are accessible through ssh with the same account credentials as the cluster. They are:

- dt01.bsc.es
- dt02.bsc.es

These machines share the GPFS filesystem with all other BSC HPC machines. Besides scp and sftp, they allow some other useful transfer protocols:

- scp

```
localsystem$ scp localfile username@dt01.bsc.es:
username's password:

localsystem$ scp username@dt01.bsc.es:remotefile localdir
username's password:
```

- rsync

```
localsystem$ rsync -avzP localfile_or_localdir username@dt01.bsc.es:
username's password:

localsystem$ rsync -avzP username@dt01.bsc.es:remotefile_or_remotedir localdir
username's password:
```

- sftp

```
localsystem$ sftp username@dt01.bsc.es
username's password:
sftp> get remotefile

localsystem$ sftp username@dt01.bsc.es
username's password:
sftp> put localfile
```

- BBCP

```
bbcp -V -z <USER>@dt01.bsc.es:<FILE> <DEST>
bbcp -V <ORIG>  <USER>@dt01.bsc.es:<DEST>
```

- SSHFTP

```
globus-url-copy -help
globus-url-copy -tcp-bs 16M -bs 16M -v -vb your_file sshftp://your_user@dt01.bsc.es/~/
```

3.3 Active Archive Management

Active Archive (AA) is a mid-long term storage filesystem that provides 15 PB of total space. You can access AA from the Data Transfer Machine (section 3.2) (dt01.bsc.es and dt02.bsc.es) under /gpfs/archive/hpc/your_group.

NOTE: There is no backup of this filesystem. The user is responsible for adequately managing the data stored in it.

To move or copy from/to AA you have to use our special commands:

- dtcp, dtmv, dtrsync, dttar

These commands submit a job into a special class performing the selected command. Their syntax is the same than the shell command without 'dt' prefix (cp, mv, rsync, tar).

- dtq, dtcancel

dtq shows all the transfer jobs that belong to you. (works like mnq)
dtcancel works like mncancel (see below) for transfer jobs.

- *dttar:* submits a tar command to queues. Example: Taring data from /gpfs/to /gpfs/archive/hpc

```
% dttar -cvf  /gpfs/archive/hpc/usertest/outputs.tar ~/OUTPUTS
```

- *dtcp:* submits a cp command to queues. Remember to delete the data in the source filesystem once copied to AA to avoid duplicated data.

```
# Example: Copying data from /gpfs to /gpfs/archive/hpc
% dtcp -r  ~/OUTPUTS /gpfs/archive/hpc/usertest/
```

```
# Example: Copying data from /gpfs/archive/hpc to /gpfs
% dtcp -r  /gpfs/archive/hpc/usertest/OUTPUTS ~/
```

- *dtrsync:* submits a rsync command to queues. Remember to delete the data in the source filesystem once copied to AA to avoid duplicated data.

```
# Example: Copying data from /gpfs to /gpfs/archive/hpc
% dtrsync -avP  ~/OUTPUTS /gpfs/archive/hpc/usertest/
```

```
# Example: Copying data from /gpfs/archive/hpc to /gpfs
% dtrsync -avP  /gpfs/archive/hpc/usertest/OUTPUTS ~/
```

- *dtmv:* submits a mv command to queues.

```
# Example: Moving data from /gpfs to /gpfs/archive/hpc
% dtmv ~/OUTPUTS /gpfs/archive/hpc/usertest/
```

```
# Example: Moving data from /gpfs/archive/hpc to /gpfs
% dtmv /gpfs/archive/hpc/usertest/OUTPUTS ~/
```

Additionally, these commands accept the following options:

```
--blocking: Block any process from reading file at final destination until transfer completed.

--time: Set up new maximum transfer time (Default is 18h).
```

It is important to note that these kind of jobs can be submitted from both the 'login' nodes (automatic file management within a production job) and 'dt01.bsc.es' machine. AA is only mounted in Data Transfer Machine (section 3.2). Therefore if you wish to navigate through AA directory tree you have to login into dt01.bsc.es

3.4 Repository management (GIT/SVN)

There's no outgoing internet connection from the cluster, which prevents the use of external repositories directly from our machines. To circumvent that, you can use the "sshfs" command in your local machine.

Doing that, you can mount a desired directory from our GPFS filesystem in your local machine. That way, you can operate your GPFS files as if they were stored in your local computer. That includes the use of git, so you can clone, push or pull any desired repositories inside that mount point and the changes will transfer over to GPFS.

Setting up sshfs

- Create a directory inside your local machine that will be used as a mount point.
- Run the following command below, where the local directory is the directory you created earlier. Note that this command mounts your GPFS home directory by default.

```
sshfs -o workaround=rename <yourHPCUser>@dt01.bsc.es: <localDirectory>
```

- From now on, you can access that directory. If you access it, you should see your home directory of the GPFS filesystem. Any modifications that you do inside that directory will be replicated to the GPFS filesystem inside the HPC machines.
- Inside that directory, you can call "git clone", "git pull" or "git push" as you please.

4 File Systems

IMPORTANT: It is your responsibility as a user of our facilities to backup all your critical data. *We only guarantee a daily backup of user data under /gpfs/home. Any other backup should only be done exceptionally under demand of the interested user.*

Each user has several areas of disk space for storing files. These areas may have size or time limits, please read carefully all this section to know about the policy of usage of each of these filesystems. There are 3 different types of storage available in the cluster:

- *Root filesystem:* Is the filesystem where the operating system resides
- *FEFS:* FEFS is a distributed networked filesystem which can be accessed from all the nodes.
- *GPFS filesystem:* GPFS is a distributed networked filesystem which has two partitions in this machine gpfs_home and gpfs_projects. Both can be accessed from login nodes and Data Transfer Machine but only gpfs_home is mounted on computing nodes too.

4.1 Root Filesystem

The root file system, where the operating system is stored has its own partition.

There is a separate partition of the local hard drive mounted on /tmp that can be used for storing user data as you can read in [Local Hard Drive].

4.2 Parallel Filesystems

The Fujitsu Exabyte File System (FEFS) is a scalable cluster file system based on Lustre with high reliability and high availability for all nodes of the cluster. Besides, the IBM General Parallel File System (GPFS) is a high-performance shared-disk file system providing fast, reliable data access from all nodes of the cluster to a global filesystem.

These filesystems allow parallel applications simultaneous access to a set of files (even a single file) from any node that has the file system mounted while providing a high level of control over all file system operations.

The following filesystems are used in the cluster:

- */apps* (softlink to /fefs/apps): Over this filesystem will reside the applications and libraries that have already been installed on the machine. Take a look at the directories to know the applications available for general use.
- */home* (softlink to /gpfs/home): This filesystem has the home directories of all the users, and when you log in you start in your home directory by default. Every user will have their own home directory to store own developed sources and their personal data. A default quota will be enforced on all users to limit the amount of data stored there. Also, it is highly discouraged to run jobs from this filesystem. Please run your jobs on your group's /scratch instead.
- */gpfs/projects*: In addition to the home directory, there is a directory in /gpfs/projects for each group of users. For instance, the group bsc01 will have a /gpfs/projects/bsc01 directory ready to use. This space is intended to store data that needs to be shared between the users of the same group or project. A quota per group will be enforced depending on the space assigned by Access Committee. It is the project's manager responsibility to determine and coordinate the better use of this space, and how it is distributed or shared between their users. This filesystem is not mounted on computing nodes, you have to transfer you data and launch your jobs from /scratch//
- */scratch* (softlink to /fefs/scratch): It is the only filesystem intended for executions. There is a directory for every group inside it and a subdirectory for every user in the group directory. It is mounted on every node, login and computing nodes. You must transfer all your necessary scripts, input files and any other kind of data to this filesystem before launching a job.

5 Running Jobs

PJM is the utility used for batch processing support, so all jobs must be run through it. This section provides information for getting started with job execution at the Cluster.

5.1 Job Queues

There are several queues present in the machines and different users may access different queues. You can check anytime all queues you have access to using:

```
$ pjshowrsc --rg

[ CLST: compute ]
[ RSCUNIT: rscunit_ft02 ]
RSCGRP          NODE
              TOTAL  CNS FREE  ALLOC
small            24    24    0
middle           96    96    0
important       192   192    0
def_grp          96    96    0
large           192   192    0
```

5.2 Submitting Jobs

The method for submitting jobs is to use the PJM directives directly.

A job is the execution unit for PJM. A job is defined by a text file containing a set of directives describing the job's requirements and the commands to execute. These are the basic directives to submit jobs:

```
pjsub <job_script>
```

Submits a "job script" to the queue system, similar to sbatch in SLURM.

```
pjstat
```

Shows all the submitted jobs, similar to squeue in SLURM.

```
pjhold <job_id> / pjrls <job_id>
```

Holds and releases respectively a non-empty set of jobs with the given job id.

```
pjdel <job_id>
```

Deletes the job with the given <job_id>.

For a deep explanation of each command, please refer to their man pages.

Disclaimer About Job Submissions

If you are used to using our other HPC clusters, there's a big difference that you need to take into account when using CTE-ARM. In this cluster, in order to avoid potential issues when trying to write or access files at job execution time, **it is imperative** that the output files and the working directories are located inside the /fefs filesystem. This also includes the paths of the job output/error files.

Failing to do so can make your jobs to fail unexpectedly, so make sure to follow this general rule.

5.3 Interactive Sessions

Allocation of an interactive session has to be done through PJM:

```
pjsub --interact
```

5.4 Job Directives

A job must contain a series of directives to inform the batch system about the characteristics of the job. These directives appear as comments in the job script, here you may find the most common directives for both syntaxes:

```
#PJM -N <name>
```

Specify the name of the job

```
#PJM -j
```

Store both Standard Output and Standard Error to the same file, it will ignore -e directive if specified.

```
#PJM -X
```

Inherit environmental variables at batch job submission to the running environment of the job

```
#PJM -L rscgrp=<name>
```

Name of the resource group to submit the job, similar to qos in SLURM.

```
#PJM -L elapse=[[HH:]MM:]SS
```

The limit of wall clock time. you must set it to a value greater than real execution time for your application. Notice that your job will be killed after the time has passed.

```
#PJM -L node=<number>
```

The number of requested nodes.

```
#PJM --mpi "parameter[,...]"
```

This option specifies the parameters of an MPI job. These are more common parameters:

- proc= -> The number of processes to start.
- max-proc-per-node= -> The number of processes by node.

In order to use both options simultaneously, you have to use the following syntax:

```
#PJM --mpi "proc=<number1>,max-proc-per-node=<number2>"
```

This way, you can tune the mpi settings for your job.

```
#PJM -o filename
```

The name of the file to collect the standard output (stdout) of the job.

```
#PJM -e filename
```

The name of the file to collect the standard error output (stderr) of the job.

5.5 Standard output/error management

Standard output and standard error output are saved in files. If the output files were not specified, they will be created in the directory where the pjsub command was issued (%n is the job name, name of the jobscript if not specificed, and %j is the job id):

- *%n.%j.out*standard output
- *%n.%j.err*standard error output

5.6 Examples

Here you have an example for a sequential job:

```
#!/bin/bash
#------ pjsub option --------#
#PJM -L "rscgrp=small"
# Name of the resource group (= queue) to submit the job
#PJM -N serial
# Name of the job (optional)
#PJM -L node=1
# Specify the number of required nodes
#PJM -L elapse=00:05:00
# Specify the maximum duration of a job
#PJM -j
# Store stdout and stderr in the same file
#------- Program execution -------#
/usr/bin/hostname
```

The job would be submitted using "pjsub ". The output will be stored in the same directory as the file serial.*.out, where * is the job id.

In this case we have an example of a parallel job using MPI:

```
#!/bin/bash
#PJM -N parallel
#PJM -L rscgrp=small
#PJM -L node=2
#PJM -L elapse=0:30:00
#PJM --mpi "proc=6,max-proc-per-node=3"
# The number of MPI processes and the maximum of processes per node
#PJM -o job-%j.out
# File where standard output will be stored
#PJM -e job-%j.err
# File where standard errors will be stored

export PATH=/opt/FJSVxtclanga/tcsds-1.1.18/bin:$PATH
export LD_LIBRARY_PATH=/opt/FJSVxtclanga/tcsds-1.1.18/lib64:$LD_LIBRARY_PATH

mpirun -np 6 /fefs/apps/examples/test
```

This job will launch six MPI tasks distributed in two nodes

6 Compiling Software

There's one existing module for cross-compilation and MPI libraries from the login nodes:

```
module load fuji
```

You will be able to compile Fortran, C and Java with MPI:

```
mpifrtpx tesf.f -o test
mpifccpx test.c -o test
```

For longer compilations, you should request a node interactively or submit a jobscript and run the compilation on it.

7 Getting help

BSC provides users with excellent consulting assistance. User support consultants are available during normal business hours, Monday to Friday, 09 a.m. to 18 p.m. (CEST time).

User questions and support are handled at: support@bsc.es

If you need assistance, please supply us with the nature of the problem, the date and time that the problem occurred, and the location of any other relevant information, such as output files. Please contact BSC if you have any questions or comments regarding policies or procedures.

Our address is:

```
Barcelona Supercomputing Center - Centro Nacional de Supercomputación
C/ Jordi Girona, 31, Edificio Capilla 08034 Barcelona
```

7.1 Frequently Asked Questions (FAQ)

You can check the answers to most common questions at BSC's Support Knowledge Center[1]. There you will find online and updated versions of our documentation, including this guide, and a listing

[1] https://www.bsc.es/user-support/faq.php

with deeper answers to the most common questions we receive as well as advanced specific questions unfit for a general-purpose user guide.

8 Appendices

8.1 SSH

SSH is a program that enables secure logins over an insecure network. It encrypts all the data passing both ways, so that if it is intercepted it cannot be read. It also replaces the old an insecure tools like telnet, rlogin, rcp, ftp,etc. SSH is a client-server software. Both machines must have ssh installed for it to work.

We have already installed a ssh server in our machines. You must have installed an ssh client in your local machine. SSH is available without charge for almost all versions of UNIX (including Linux and MacOS X). For UNIX and derivatives, we recommend using the OpenSSH client, downloadable from http://www.openssh.org, and for Windows users we recommend using Putty, a free SSH client that can be downloaded from http://www.putty.org. Otherwise, any client compatible with SSH version 2 can be used.

This section describes installing, configuring and using the client on Windows machines. No matter your client, you will need to specify the following information:

- Select SSH as default protocol
- Select port 22
- Specify the remote machine and username

For example with putty client:

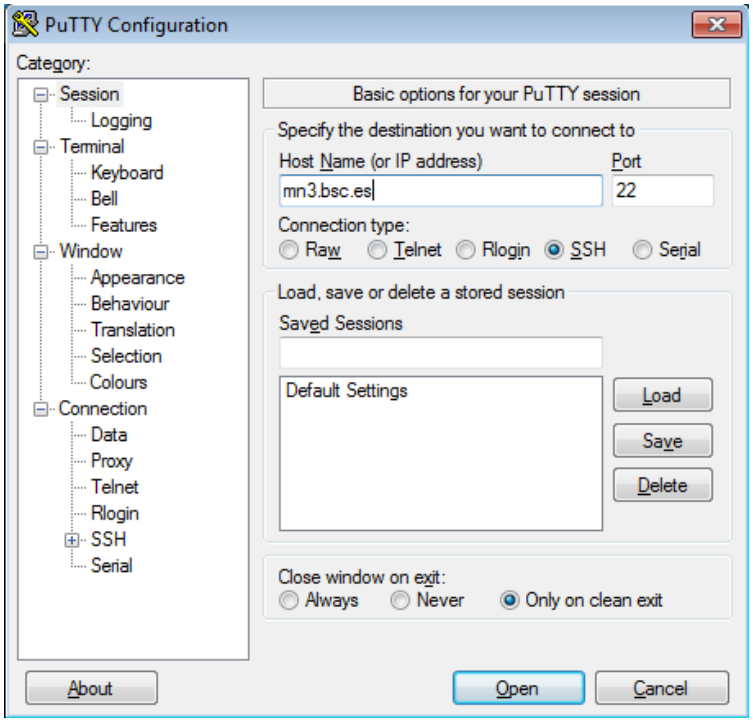

Figure 1: Putty client

This is the first window that you will see at putty startup. Once finished, press the **Open** button. If it is your first connection to the machine, your will get a *Warning* telling you that the host key from the server is unknown, and will ask you if you are agree to cache the new host key, press Yes.

IMPORTANT: If you see this warning another time and you haven't modified or reinstalled the ssh client, please do *not* log in, and contact us as soon as possible (see Getting Help (chapter 7)).

Finally, a new window will appear asking for your login and password:

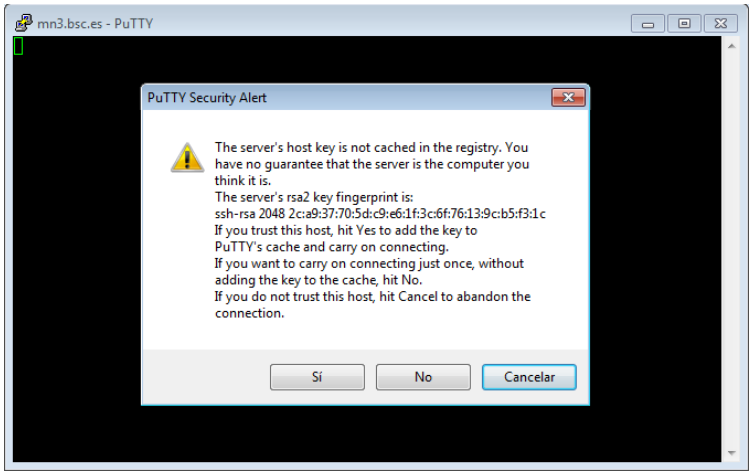

Figure 2: Putty certificate security alert

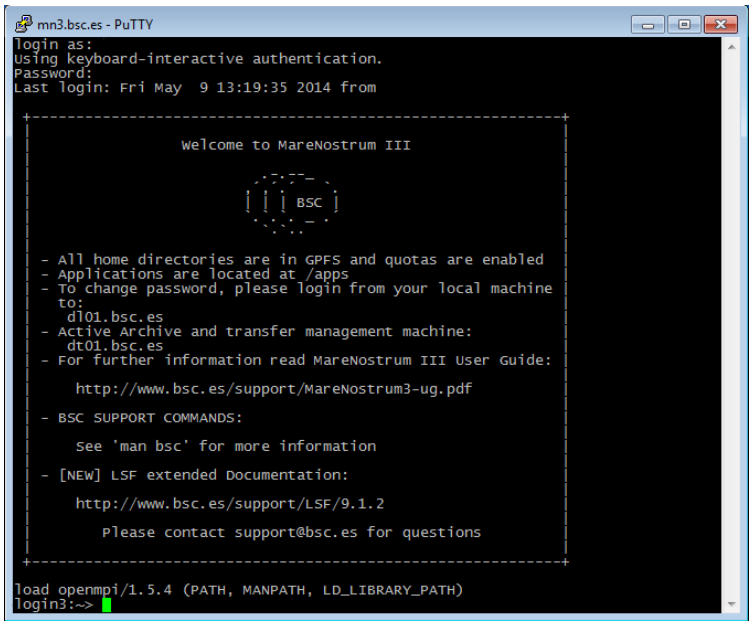

Figure 3: Cluster login

8.2 Transferring files

To transfer files to or from the cluster you need a secure ftp (sftp) o secure copy (scp) client. There are several different clients, but as previously mentioned, we recommend using of Putty clients for transferring files: **psftp** and **pscp**. You can find it at the same web page as Putty (http://www.putty.org[2]).

Some other possible tools for users requiring graphical file transfers could be:

- WinSCP: Freeware Sftp and Scp client for Windows (http://www.winscp.net)
- SSH: Not free. (http://www.ssh.org)

Using PSFTP

You will need a command window to execute psftp (press start button, click run and type cmd). The program first asks for the machine name (mn1.bsc.es), and then for the username and password. Once you are connected, it's like a Unix command line.

With command **help** you will obtain a list of all possible commands. But the most useful are:

[2] http://www.putty.org/

- get file_name : To transfer from the cluster to your local machine.
- put file_name : To transfer a file from your local machine to the cluster.
- cd directory : To change remote working directory.
- dir : To list contents of a remote directory.
- lcd directory : To change local working directory.
- !dir : To list contents of a local directory.

You will be able to copy files from your local machine to the cluster, and from the cluster to your local machine. The syntax is the same that cp command except that for remote files you need to specify the remote machine:

```
Copy a file from the cluster:
> pscp.exe username@mn1.bsc.es:remote_file local_file
Copy a file to the cluster:
> pscp.exe local_file username@mn1.bsc.es:remote_file
```

8.3 Using X11

In order to start remote X applications you need and X-Server running in your local machine. Here is a list of most common X-servers for windows:

- Cygwin/X: http://x.cygwin.com
- X-Win32 : http://www.starnet.com
- WinaXe : http://labf.com
- XconnectPro : http://www.labtam-inc.com
- Exceed : http://www.hummingbird.com

The only Open Source X-server listed here is Cygwin/X, you need to pay for the others.
Once the X-Server is running run putty with X11 forwarding enabled:

I tried running a X11 graphical application and got a GLX error, what can I do?

If you are running on a macOS/Linux system and, when you try to use some kind of graphical interface through remote SSH X11 remote forwarding, you get an error similar to this:

```
X Error of failed request: BadValue (integer parameter out of range for operation)
  Major opcode of failed request: 154 (GLX)
  Minor opcode of failed request: 3 (X_GLXCreateContext)
  Value in failed request: 0x0
  Serial number of failed request: 61
  Current serial number in output stream: 62
```

Try to do this fix:
macOS:

- Open a command shell, type, and execute:

```
$ defaults write org.macosforge.xquartz.X11 enable_iglx -bool true
```

- Reboot your computer.

Linux:

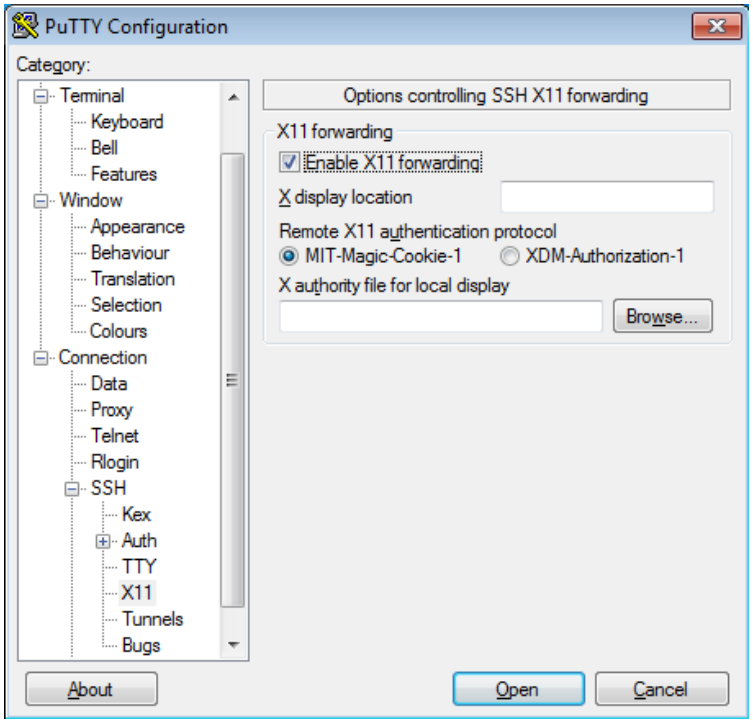

Figure 4: Putty X11 configuration

- Edit (as root) your Xorg config file and add this:

```
Section "ServerFlags"
    Option "AllowIndirectGLX" "on"
    Option "IndirectGLX" "on"
EndSection
```

- Reboot your computer.

This solves the error most of the time. The error is related to the fact that some OS versions have disabled indirect GLX by default, or disabled it at some point during an OS update.

8.4 Requesting and installing a .X509 user certificate

If you are a BSC employee (and you also have a PRACE account), you may be interested in obtaining and configuring a x.509 Grid certificate. If that is the case, you should follow this guide. First, you should obtain a certificate following the details of this guide (you must be logged in the BSC intranet):

- https://intranet.bsc.es/help-and-support/operations-services/personal-digital-certificates

Once you have finished requesting the certificate, you must download it in a ".p12" format. This procedure may be different depending on which browser you are using. For example, if you are using Mozilla Firefox, you should be able to do it following these steps:

- Go to "Preferences".
- Navigate to the "Privacy & Security" tab.
- Scroll down until you reach the "Certificates" section. Then, click on "View Certificates..."
- You should be able to select the certificate you generated earlier. Click on "Backup...".
- Save the certificate as "usercert.p12". Give it a password of your choice.

Once you have obtained the copy of your certificate, you must set up your environment in your HPC account. To acomplish that, follow these steps:

- Connect to dt02.bsc.es using your PRACE account.
- Go to the GPFS home directory of your HPC account and create a directory named ".globus".
- Upload the .p12 certificate you created earlier inside that directory.
- Once you are logged in, insert the following commands (insert the password you chose when needed):

```
module load prace globus
cd ~/.globus
openssl pkcs12 -nocerts -in usercert.p12 -out userkey.pem
chmod 0400 userkey.pem
openssl pkcs12 -clcerts -nokeys -in usercert.p12 -out usercert.pem
chmod 0444 usercert.pem
```

Once you have finished all the steps, your personal certificate should be fully installed.