

中国科学院理论物理研究所高性能计算集群

# 使用手册

Introductions for HPC Cluster of ITP-CAS

科学计算与信息平台办公室

SCIENTIFIC COMPUTING AND INFORMATION OFFICE

## 目录

声明 .....	2
1 系统资源 .....	3
1.1 硬件概况 .....	3
1.2 队列列表 .....	错误!未定义书签。
1.3 已安装软件列表 .....	4
1.4 Linux 环境变量 .....	5
1.5 软件环境变量 .....	5
2 登录集群 .....	6
2.1 系统登陆 .....	6
2.1.1 登录节点 IP 地址 .....	6
2.1.2 更换密钥 .....	6
2.1.3 采用 SSH 密钥登录 .....	7
2.1.4 文件上传下载 .....	10
3 PBS 作业调度 .....	10
3.1 作业调度系统 .....	10
3.2 PBS 运行参数 .....	11
3.3 PBS 环境变量 .....	11
3.4 PBS 脚本举例 .....	12
3.5 常用命令 .....	13
4 应用举例 .....	14
4.1 Mathematica .....	14
4.2 调用 lapack 库 .....	14
4.3 TensorFlow 的使用 .....	15
5 版本说明 .....	16

## 声明

中国科学院理论物理研究所高性能计算集群 (HPC Cluster of ITP-CAS) 作为所级平台, 为研究所的科研和教学提供科学计算与数值模拟服务。凡在集群上完成的相关成果, 请在成果中给予标注, 并向研究所图书馆提供相应的文件以便于存档。致谢可参考如下范例, “The results described in this paper are supported by **HPC Cluster of ITP-CAS**”。

本文档仅供内部交流学习使用, 请勿用作其他用途。如您对集群使用有疑问或建议, 欢迎加入 QQ 群 (群号: 646926958) 在线交流。

# 1 系统资源

## 1.1 硬件概况

- 登录节点：1 台(login2017.itp.ac.cn)，提供用户提交作业等服务。登录后默认是新的 home 目录，旧集群的 home 目录在/public\_old/home/目录下。
- 计算队列：o\_cpu(已下线)、ncpu、gpu、gpu4、fat1t
- 网络系统：100Gb/s Infiniband EDR 网络。
- 存储系统：集群存储采用并行存储系统，目前分新旧两套存储系统。
  - 新存储系统 IB 网络采用 100Gb/s，容量为 221TB，挂载点为/public。**PS: 该存储按照课题组进行配额，每课题组默认配额为 4TB。**存储目录结构如下：
    - 用户目录： /public/home/
    - 软件目录： /public/software/
    - 源代码目录： /public/sourcecode/
    - 实例目录： /public/example/
    - 测试目录： /public/test/
  - 原存储节点：旧存储系统 IB 网络采用 56Gb/s，容量为 181TB。该存储可从登陆节点访问，挂载点为/public\_old；也可从旧登陆节点使用原来的用户名和密码登陆访问，挂载点为/public/。

**PS: 因该存储已服务较长时间，存在数据安全风险，将在 2018 年元旦停止服务。**  
**请用户在年底前下载备份原有文件，用户目录位置为/public\_old/home/。因 home 目录的用户权限问题，建议新集群用户都从新集群的登陆节点访问。**没有新集群账号的应从旧登陆节点（159.226.161.101/ cluster.itp.ac.cn）访问。
- 原登陆节点：1 台（cluster.itp.ac.cn），为只有旧集群账号的用户提供数据备份，不再提供作业提交服务。

## 1.2 队列列表

根据计算集群的不同硬件环境，计算节点划分为 o\_cpu、ncpu、gpu、gpu4、fat1t 等队列。

队列	单节点配置					系统指标				队列运行状况
	CPU 型号	主频	内存 GB	核数	GPU	台数	核数	IB 网络性能	峰值 Tflops 实测/理论	
o_cpu	AMD Opteron 6132HE	2.2G	128	32	0	47	1504	56Gb/s	10.5/13.3	下线
ncpu	AMD Opteron 6276	2.3G	64G	32	0	90	2880	56Gb/s	18.6/26.5	open
gpu	Intel Xeon E5-2640	2.5G	32G	12	2* K20	40	480	56Gb/s	76.3/103	open
gpu4	Intel Xeon E5-2640v2	2.0G	64G	16	4*K20	36	576	56Gb/s	124.4/177.7	open
fat1t	Intel Xeon E704850v3	2.2	1TB	56	0	16	896	100Gb/s	18.0/31.5	open

## 1.3 已安装软件列表

集群基础软件部署			
类型	软件名	版本号	安装位置
编译器 &库	GNU 编译器	4.9.1	/public/software/compiler/gnu/4.9.1
		5.4.0	/public/software/compiler/gnu/5.4.0
		6.4.0	/public/software/compiler/gnu/6.4.0
		7.1.0	/public/software/compiler/gnu/7.1.0
	Intel 编译器	2015.2.164	/public/software/compiler/intel/composer_xe_2015.2.164
		2016.3.210	/public/software/compiler/intel/composer_xe_2016.3.210
		2017.2.174	/public/software/compiler/intel/composer_xe_2017.2.174
	ocaml	4.04.1	/public/software/compiler/ocaml/4.04.1
	glibc	2.17	/public/software/compiler/glibc/2.17
数学库	MKL	2015.2.164	/public/software/compiler/intel/composer_xe_2015.2.164/mkl
	MKL	2016.3.210	/public/software/compiler/intel/composer_xe_2016.3.210/mkl
	fftw-float	2.1.5	/public/software/mathlib/fftw/2.1.5/float
		3.3.4	/public/software/mathlib/fftw/3.3.4/float
	fftw-double	2.1.5	/public/software/mathlib/fftw/2.1.5/double
		3.3.4	/public/software/mathlib/fftw/3.3.4/double
	lapack-gnu	3.4.2	/public/software/mathlib/lapack/3.4.2/gnu
	lapack-intel	3.4.2	/public/software/mathlib/lapack/3.4.2/intel
	lapack++	1.1	/public/software/mathlib/lapack++
MPI	petsc	3.4.3	/public/software/mathlib/petsc
	intelmpi	5.0.2.044	/public/software/mpi/intelmpi/5.0.2.044
	intelmpi	5.1.3.210	/public/software/mpi/intelmpi/5.1.3.210
	openmpi	1.6.5	/public/software/mpi/openmpi/1.6.5/intel
	openmpi	1.8.7	/public/software/mpi/openmpi/1.8.7/intel
	openmpi	2.0.1-gnu	/public/software/mpi/openmpi/2.0.1/gnu
	openmpi	2.0.1-intel	/public/software/mpi/openmpi/2.0.1/intel
	openmpi	2.1.1-intel	/public/software/mpi/openmpi/2.1.1/intel
apps	R	3.2.1	/public/software/apps/R/3.2.1
	Abinit	7.10.4	/public/software/apps/abinit/7.10.4/openmpi
	cosmomc	Nov2016	/public/software/apps/CosmoMC/Nov2016/intelmpi
	cp2k	2.6.1	/public/software/apps/cp2k/2.6.1/intelmpi
	gaussian	g09	/public/software/apps/gaussian/g09
	gromacs	5.1.3	/public/software/apps/gromacs/5.1.3/openmpi
	lammps	14May16	/public/software/apps/lammps/14May16/intelmpi
	namd	2.10	/public/software/apps/namd/2.10/openmpi
	python	2.7.12	/public/software/apps/python/2.7.12
		3.6.1	/public/software/apps/python/3.6.1
	vasp	5.4.4	/public/software/apps/vasp/5.4.4/intelmpi
	xmds	2.2.2	/public/software/apps/python/2.7.12/bin/xmds2
	ocaml	4.04.1	/public/software/compiler/ocaml/4.04.1
	Mathematica	8.0.4	/public/software/apps/mathematica/8.0.4
	trng	1.1	/public/software/apps/trng
	pythia	8.2	/public/software/apps/pythia8226
	root	6.10.02	/public/software/apps/singularity/bin/root_v6.10.02.img
	TensorFlow		/public/software/apps/singularity/bin/tensorflow.img

## 1.4 Linux 环境变量

Linux 是一个多用户的操作系统，每个用户登录系统后都有自己专用的运行环境。这个环境是由一组变量所定义，这组变量被称为环境变量，环境变量可以被当前用户所运行的所有程序所使用。Linux 系统的环境变量和 Shell 紧密相关的，它是通过 Shell 命令来设置的。

### ■ 全局环境变量

/etc/profile、/etc/profile.d/\*.sh 这些文件中设置的环境变量对所有用户都起作用，登录时自动生效，称为全局环境变量。全局环境变量可用来设定一些默认的应用环境，如指定编译器、MPI 并行库等。

### ■ 用户环境变量

为了避免版本冲突，应用程序后相关的环境参数通常都没有加入的全局环境变量，集群中在/public/software/profile.d/目录下创建相应的 env 文件，方便用户在 ~/.bashrc 或 PBS 脚本中自由选择 source，配置自己专属的环境变量。

### ■ 常见的环境变量命令

echo: 显示环境变量，例如 echo \$HOME

export: 设置一个新的环境变量，例如 export NAME="Raid Cheng"

unset: 清除环境变量，例如 unset NAME

常见的环境变量，如

HOME 当前用户的主目录

PATH 决定 shell 到哪些目录中寻找命令或可执行程序

LD\_LIBRARY\_PATH 设定在哪些目录寻找动态链接库

INCLUDE：编译程序时设定在哪些目录下寻找头文件

## 1.5 软件环境变量

集群软件采用 module 模块管理，module 是环境变量模块化管理工具（可参考 <http://modules.sourceforge.net/>），可以自动处理编译器及函数库的依赖。

环境变量文件存放在/public/software/modules，module avail 查看所有可用模块：

```
$ module avail
[root@admin1001 ~]# module avail

----- /public/software/modules -----
apps/R/3.2.1                benchmark/hpl/2.2/intel      mathlib/fftw/3.3.4/float
apps/abinit/7.10.4/openmpi  benchmark/i7z/0.28/gnu     mathlib/lapack/3.4.2/gnu
apps/cosmomc/Nov2016/intelmpi benchmark/iozone/3.430/gnu  mathlib/lapack/3.4.2/intel
apps/cp2k/2.6.1/intelmpi    benchmark/stream/5.10/intel mathlib/petsc/3.4.3/intel
apps/gaussian/g09           compiler/glibc/2.17         mpi/intelmpi/5.0.2.044
apps/gromacs/5.1.3/openmpi  compiler/gnu/4.9.1         mpi/intelmpi/5.1.3.210
apps/lammps/14May16         compiler/intel/composer_xe_2015.2.164 mpi/openmpi/1.6.5/intel
apps/namd/2.10/openmpi      compiler/intel/composer_xe_2016.3.210 mpi/openmpi/1.8.7/intel
apps/python/2.7.12          compiler/ocaml/4.04.1       mpi/openmpi/2.0.1/gnu
apps/python/3.6.1           mathlib/fftw/2.1.5/double   mpi/openmpi/2.0.1/intel
apps/vasp/5.4.4             mathlib/fftw/2.1.5/float
benchmark/hpl/2.1/intel     mathlib/fftw/3.3.4/double
[root@admin1001 ~]#
```

使用 module load 加载环境变量，例如要使用 openmpi-2.0.1 intel 版本，可用如下命令加载：

```
$ module load mpi/openmpi/2.0.1/intel
```

module 常用命令：

module avail	# 查看可用环境变量
module load	# xxx 加载某环境变量
module list	# 查看已加载环境变量
module unload	# xxx 卸载某环境变量
module purge	# 清除所有环境变量

## 2 登录集群

### 2.1 系统登陆

#### 2.1.1 登录节点 IP 地址

集群新登录节点地址是 [login2017.itp.ac.cn](http://login2017.itp.ac.cn), 端口 22, 出于安全考虑该节点只允许采用 ssh 密钥登录方式。该登陆节点可访问队列为：ncup、gpu、gpu4、fat1t 队列。

因集群存在不同的硬件环境 AMD CPU 节点、GPU 节点、Intel Xeon E7 CPU 节点，不同的硬件构架下对指令集的兼容性也存在差异。在 login2017 登录节点上编译的程序，存在无法在其他队列如 ncpu\gpu 等计算节点上正确运行。因此，建议提交到 ncpu 队列的作业可从 login2017 节点 ssh 登入 login1 节点上编译后再提交；提交到 gpu 或 gpu4 队列的作业可从 login2017 节点 ssh 登入 glogin1 或 glogin2 节点上编译后再提交。

旧登陆节点为 login2 (IP:159.226.161.102)，仅供没有新集群账号的、原集群用户提供文件下载、文件整理，无法提交作业。

#### 2.1.2 更换密钥

密钥是用户登录集群系统的凭证文件，请用户妥善保管，一旦密钥泄露，用户账号安全、用户数据安全将无法保证，同时也会对集群系统的安全带来严重危害。

如果用户需要更换密钥，可与系统管理员联系重新生产新的密钥对。或者登入集群系统后，通过以下命令，生成密钥对并下载保存新的密钥。

- 1) 生成新的密钥文件 id\_rsa 和 public 密钥文件 id\_rsa.pub

```
# ssh-keygen -t rsa
```

```
$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/public/home/.../.ssh/id_rsa):
/public/home/.../.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /public/home/.../.ssh/id_rsa.
Your public key has been saved in /public/home/.../.ssh/id_rsa.pub.
The key fingerprint is:
...
The key's randomart image is:
```

PS: 在生成密钥对的过程，可以选择输入 **passphrase** 作为安全选项。一旦输入了 passphrase，

则要牢记，在导入 Xshell 中 id\_rsa 文件、登录系统时候需要输入 **passphrase**，否则无法登录。

- 2) 如果没有更新 home 中隐藏文件夹.ssh 下 authorized\_keys 文件内容，则需要按照下面方式更新。(系统在生成密钥对的时候默认会更新 authorized\_keys 文件)

```
# cat id_rsa.pub >>authorized_keys
```

- 3) 下载 home 中隐藏文件夹.ssh 下密钥文件 id\_rsa 到本地。

**备注：**请不要修改 home 目录权限，以免影响 SSH 的安全审计，从而登录失败。

## 2.1.3 采用 SSH 密钥登录

### 2.1.3.1 Linux 系统及 MacOSX 系统用户

- 1) 登录登陆节点

下载私钥文件到指定目录，修改密钥文件权限

```
#chmod 600 username_id_rsa
```

打开终端在命令行下执行如下命令：

```
#ssh username@IPAddress -i username_id_rsa
```

- 2) 文件上传或下载

使用使用参数“-i”通过 scp 命令在 Linux 与 MacOSX 主机与登陆节点之间上传下载文件，

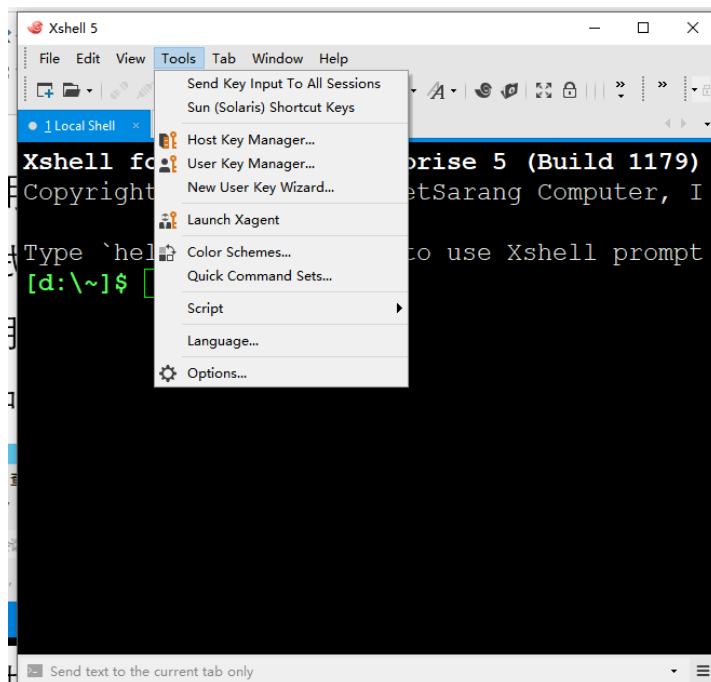
```
# scp -i user_id_rsa user@host1:sourcefile user@host2:targerfiles
```

### 2.1.3.2 Windows 系统用户

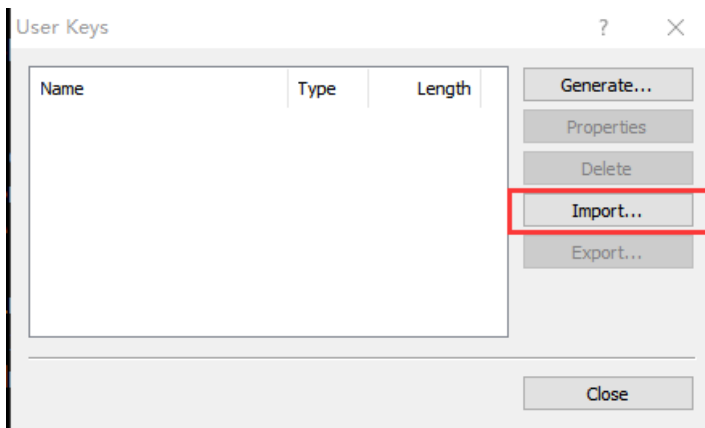
Windows 用户可以用 SSH Secure Shell Client/PuTTY/SecureCRT/xShell 等 SSH 客户端软件登录，推荐使用 xshell 或 PuTTY 登入。

- 1) Xshell 使用密钥登入

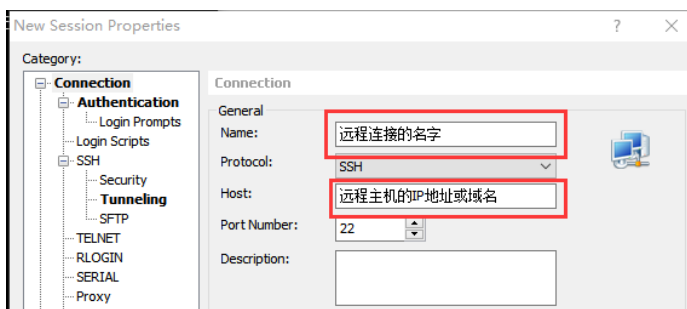
- a) 打开 xshell 添加用户私钥到xshell，选项tool-> User Key Manager...



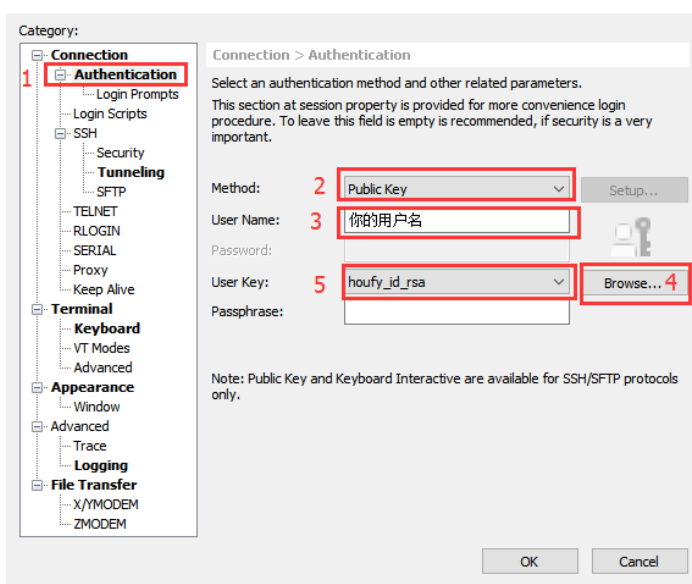
- b) 在下面选项中导入密钥文件，选项Import



c) 新建ssh 远程主机会话



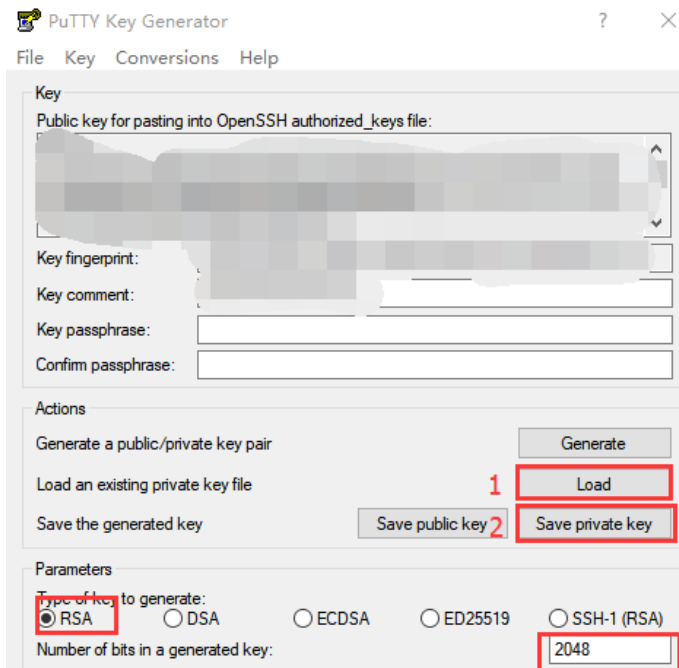
d) 选择 **Authentication** 选项卡，在认证方式中选择 **Public Key**，在用户名地方填写你申请的集群用户名，选择**Browse** 按钮找到你刚导入的私钥，点击**OK** 建立会话。



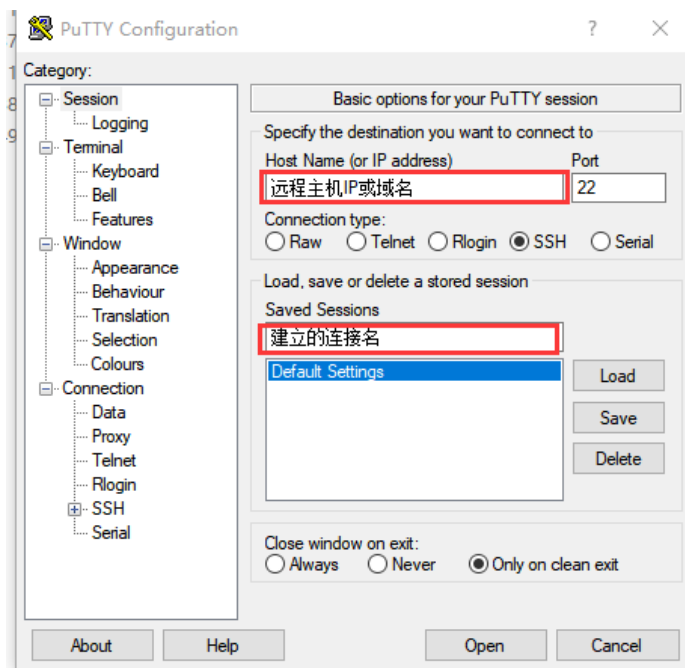
## 2) PuTTY 用户使用密钥登入

a) 安装PuTTY 后，因PuTTY 使用的密钥格式与ssh-keygen 生成的密钥格式不同，需要打开PuTTYgen,导入私钥，另存为ppk 格式密钥。

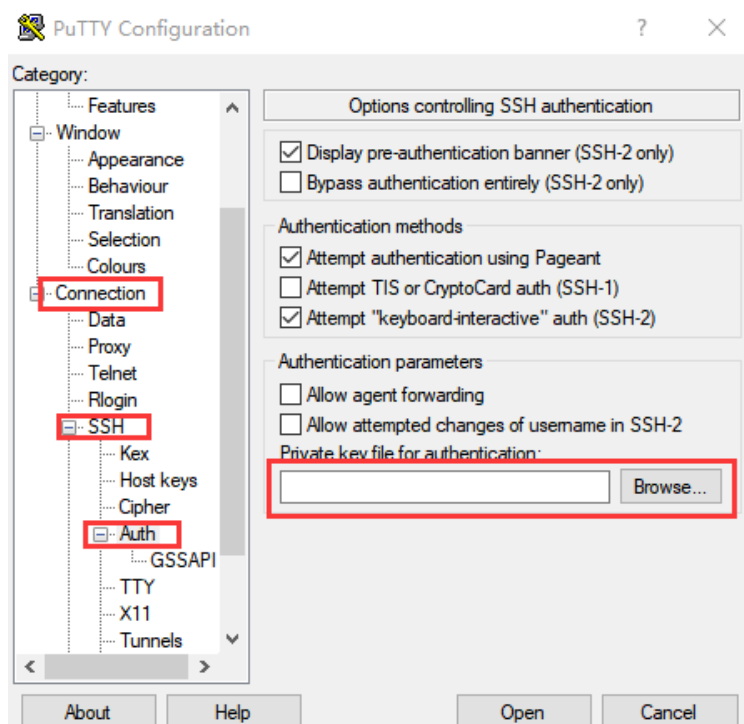




b) 打开PuTTY，输入远程主机IP 等信息



c) 在左侧Connection 选项卡，选择SSH->Auth 导入刚刚另存的PPK 密钥文件,然后  
点击 下面的Open 按钮。



D) 输入你的用户名即可登入系统。



## 2.1.4 文件上传下载

使用 xftp 工具进行文件上传下载，系统登录成功后，点击按钮启动 xftp：



初次使用需要进行身份验证，点击浏览->文件，选择刚刚的密钥文件 id\_ras，点击确定即可，通过拖拽方式上传下载文件。

## 3 PBS 作业调度

### 3.1 作业调度系统

研究所集群采用 PBS 作业调度系统，用户编写由运行程序的命令及 PBS 环境变量设置组成的 PBS 脚本，用户使用 qsub 命令提交 PBS 脚本到作业调度系统，运行作业。

## 3.2 PBS 运行参数

运行参数	参数说明
-a <作业开始运行的时间>	向 PBS 系统指定作业运行的开始时间。 作业运行时间格式为： [[[[CC]YY]MM]DD]hhmm[.SS]
-A <用户名>	使用不同的用户来提交作业，缺省使用当前用户名
-o <标准输出文件的路径> -e <标准错误输出的路径>	该参数指定标准错误输出的位置，缺省的情况下，PBS 系统把标准输出和标准错误输出放在用户 qsub 命令提交作业的目录下。  标准错误输出：<作业名>.o<作业号> 标准错误输出：<作业名>.e<作业号>  路径使用如下格式标准： [<节点名>:]<路径名>
-N <作业名>	指定提交的作业名
-q <目标队列>	指定作业提交的目标队列，其中目标队列可以是目标队列、目标节点名或者是目标节点上的队列。如果目标队列是一个路由队列，那么服务器可能把作业路由到新的队列中。如果该参数没有指定，命令 qsub 会把作业脚本提交到缺省的队列中。
-l <申请资源列表>	该参数指定作业脚本申请的 PBS 系统资源列表。 申请资源列表使用如下格式：  <资源名>=[<数量>][,资源名=[<数量>]], ...]  例如作业希望申请在双路节点上申请 5 个 CPU 资源的情况，则可以在脚本中如下：  #PBS -l nodes=2:ppn=2+1:ppn=1

## 3.3 PBS 环境变量

变 量 名	说 明
登陆 SHELL 继承来的变量	包括\$HOME, \$LANG, \$LOGNAME, \$PATH, \$MAIL, \$SHELL 和\$TZ。
\$PBS_O_HOST	qsub 提交的节点名称
\$PBS_O_QUEUE	qsub 提交的作业的最初队列名称

<b>\$PBS_O_WORKDIR</b>	qsub 提交的作业的绝对路径
<b>\$PBS_JOBID</b>	作业被 PBS 系统指定的作业号
<b>\$PBS_JOBNAME</b>	用户指定的作业名，可以在作业提交的时候用 qsub -N <作业名> 指定，或者在 PBS 脚本中加入 #PBS -N <作业名>。
<b>\$PBS_NODEFILE</b>	PBS 系统指定的作业运行的节点名。该变量在并行机和机群中使用。当在 PBS 脚本中用 #PBS -l nodes=2:ppn=2 指定程序运行的节点数时，可以使用 \$PBS_NODEFILE 在脚本中引用 PBS 系统指定的作业运行的节点名。比如： #PBS -l nodes=2:ppn=2 mpirun -np 4 -machinefile \$PBS_NODEFILE <程序名>
<b>\$PBS_QUEUE</b>	PBS 脚本在执行时的队列名

### 3.4 PBS 脚本举例

PBS 脚本本质上是 Linux 的 Shell 脚本，注释以“#”开头，PBS 运行参数以“#PBS”开头，在 PBS 脚本里可以直接调用 Shell 命令和系统命令。

#### ➤ 串行脚本举例：

```
#PBS -N test
#PBS -l nodes=1:ppn=1
#PBS -l walltime=12:00:00
#PBS -q fat1t

cd /home/user/work
./test.exe > $HOME/result/test.result
```

#### ➤ 并行脚本举例：

```
#PBS -N vasp_job
#PBS -l nodes=2:ppn=8
#PBS -q low
echo This jobs is $PBS_JOBID@$PBS_QUEUE
cd $PBS_O_WORKDIR
module load mpi/openmpi/2.0.1/intel
mpirun -np 16 -machinefile $PBS_NODEFILE ./vasp
```

#### ➤ 指定当前作业节点独占

```
#PBS -W x=NACCESSPOLICY:SINGLEJOB
```

Allows **only one job** on a node

```
#PBS -W x=NACCESSPOLICY:SINGLEUSER
```

Allows more than one job on a node but **only by a single user**

## 3.5 常用命令

### 1) 提交作业

```
qsub job.pbs
```

### 2) 删除作业，普通用户只能删除自己的作业；如作业无法删除请与管理员联系。

```
qdel jobID
```

### 3) 查看作业状态

```
qstat -a
```

(作业状态说明：E 退出；Q 排队；H 挂起；R 运行；C 结束)

-f 列出指定作业所有状态信息

-au 列出指定用户的作业

-n 列出作业运行在哪些节点

-a 显示已提交的作业状态信息

-i 列出未在运行的作业

-r 正在运行的作业

-s 所有作业状态

-Q 列出队列的限制信息

-q 已配置的所有队列状态信息

-B 服务器信息

### 4) 查看节点状态

```
pestat
```

excl: 所有 CPU 资源已被占用；

busy CPU 已接近满负荷运行

free 全部或部分 CPU 空闲

offl 该节点离线

### 5) 把作业移到另一队列

```
qmove qmove destination job_identifier
```

正在运行的、正在退出的无法移动队列

更多 PBS 的命令可参考网站 <http://www.pbsworks.com.cn> 相关文档。

## 4 应用举例

### 4.1 Mathematica

在计算集群上运行 Mathematica 程序，**请勿直接在登录节点运行**。需要 Mathematica 程序写成交互式格式 [可在 notebook 中创建后另存为.m 的格式]，比如创建一个 random 生成的程序，存为 random.m。PBS 脚本可写作：

```
#!/bin/bash
#PBS -N random
#PBS -l nodes=1:ppn=1
#PBS -l walltime=01:00:00
#PBS -j oe
#PBS -q debug
cd $PBS_O_WORKDIR
/public/software/Mathematica/MathKernel -script random.m >
outfile
exit 0
```

或者

1) 在 random.m 程序头增加一句：

```
#!/public/software/apps/mathematica/8.0.4/MathematicaScript -
script
```

2) 修改 random.m 的执行权限

```
#chmod +x random.m
```

3) 修改 PBS 脚步为

```
#!/bin/bash
#PBS -N random
#PBS -l nodes=1:ppn=1
#PBS -l walltime=01:00:00
#PBS -j oe
#PBS -q debug
cd $PBS_O_WORKDIR
./random.m > outfile
exit 0
```

然后在登陆节点上用 qsub 提交作业即可。

**小提示：** 提交作业前，请先 monitotlm 命令查看下集群上 Mathematica 剩余授权数。

### 4.2 调用 lapack 库

Lapack 库提供了线性方程求解、特征向量计算、矩阵分解等数值计算中常用的工具函

数，使用 Lapack 库有助于加快代码开发速度和提高代码质量。(感谢周海军老师提供了本示例中的测试应用程序。)

#### 1) 串行程序编译

加载 mkl 库环境

```
module load compiler/intel/composer_xe_2016.3.210
```

编译程序

```
icpc -mkl SVD.cpp -o SVD.exe
```

编写 PBS 脚本

```
#PBS -N test
#PBS -l nodes=1:ppn=1
#PBS -l walltime=24:00:00
#PBS -q fat1t

cd /home/user
./SVD.exe
```

提交作业

```
qsub example.pbs
```

#### 2) 并行程序编译

加载 mkl 环境变量

```
module load mpi/intelmpi/5.1.3.210
```

编译

```
mpiicc -mkl xxx.cpp -o xxx.exe
```

编写 PBS 作业脚本

```
#PBS -N svd
#PBS -l nodes=1:ppn=10
#PBS -l walltime=24:00:00
#PBS -q fat1t

cd /public/home/user
module load mpi/intelmpi/5.1.3.210
mpirun -np 10 -machinefile $PBS_NODEFILE ./xxx.exe
```

提交作业

```
qsub example.pbs
```

## 4.3 TensorFlow 的使用

因 TensorFlow 依赖于较新的 c 函数库等 (例如: glibc>=2.17), 为了避免升级集群环境中相应的依赖库而影响集群系统的稳定运行, 集群采用 singularity 容器化技术虚拟 (技术细节可以参考其官网: <http://singularity.lbl.gov/>), 建立一个支持相应 C 函数库的虚拟沙盒运行

环境，供用户挂载后，通过加载相应的运行环境而使用虚拟沙盒中的 TensorFlow 软件。

作为 python 模块，TensorFlow 安装在 python 模块标准位置。使用时，

- 1) 首先加载该环境变量

```
module load apps/tensorflow/1.3
```

- 2) 启动 python 环境（编写好的 python 脚本可以直接加载此命令后面执行）

```
run_tensorflow_python
```

**\*特别注意**，沙盒环境是一个虚拟环境，**并不包含其他软件的环境变量**，只能运行相应的软件，每次使用完**必须**退出该虚拟环境，退出命令为 **exit**。

## 5 版本说明

本文档版将不定期进行修订，版本编号规则为 YYYYMMXX，其中 YYYY 为年份、YY 为月份，XX 为当月第 XX 次修订。

### ■ Ver20170701 版

- 修改其中关于四路服务器 CPU 参数的错误，
- 增加对不同队列的硬件参数的介绍。

### ■ Ver20170702 版

- 增加文档版本说明，并修订部分使用说明
- 增加 GNU 编译器 5.4.0\6.4.0\7.1.0
- 增加 Intel 编译器 2017.2.174
- 安装应用软件 xmds
- 修正 GNU 编译器对 mpc 动态库的加载错误

### ■ Ver20170703 版

- 配置登录节点域名
- 修订致谢名称为 HPC Cluster of ITP-CAS
- 修订 pam\_unix 配置，排除类 unix 用户密钥登陆问题
- 安装 ocaml、Mathematica，重装 xmds
- 增加 Mathematica 应用说明

### ■ Ver20170801 版

- 安装 lapack++、trng、pythia、TensorFlow 等
- 限制普通用户直接登陆计算节点



- 修订了文档的部分说明、增加了相应的应用示例。

#### ■ Ver20170901 版

- 增加用户自己生成密钥的操作说明
- 更新计算集群管理软件
- 增加 PBS 作业独占参数使用说明
- 升级原刀片节点、GPU 节点操作系统、软件环境，开放 ncpu、gpu、gpu4 队列
- 修复并挂载旧存储，挂载位置/public\_old 目录，登陆集群后用户可拷贝旧存储数据到新集群存储目录/public/home/
- 原登陆节点 (login1\login2) 开放所内访问、下载旧集群存储文件功能
- 增加集群存储的硬件说明

#### ■ Ver20170902 版

- 修正新旧登陆节点之间的用户、home 目录的混乱。新登陆节点采用新用户、默认新的 home 目录，旧登陆节点采用旧用户、默认旧的 home 目录。

#### ■ Ver20170903 版

- 新集群用户须从新登陆节点访问旧存储的 home 目录，如果从旧登陆节点登陆存在用户目录权限问题。

#### ■ Ver20171001 版

- 因不同队列硬件构架不同，可能存在指令集的不兼容性。增加相应的登录节点，方便用户编译并提交作业到相应队列。
- 开放旧集群用户从所外登入旧登录节点，进行文件下载、备份。
- 安装 openmpi2.1.1。