南開大學

汇编语言与逆向技术课程实验报告

实验五：并行计算-HelloWorld

学　　院	网络空间安全学院
专　　业	信息安全
学　　号	2313546
姓　　名	蒋枘言
班　　级	信息安全班

一、实验目的

高性能计算（HPC, High Performance Computing）是计算机科学的一个分支，通过集群架构、并行算法和相关软件以并行计算/分布式计算的方式实现单台计算机无法达到的运算速度（每秒万亿次以上）。本实验的目的是帮助学生充分理解集群 MPI 并行计算的搭建、配置及运行，掌握在华为鲲鹏上如何运行。

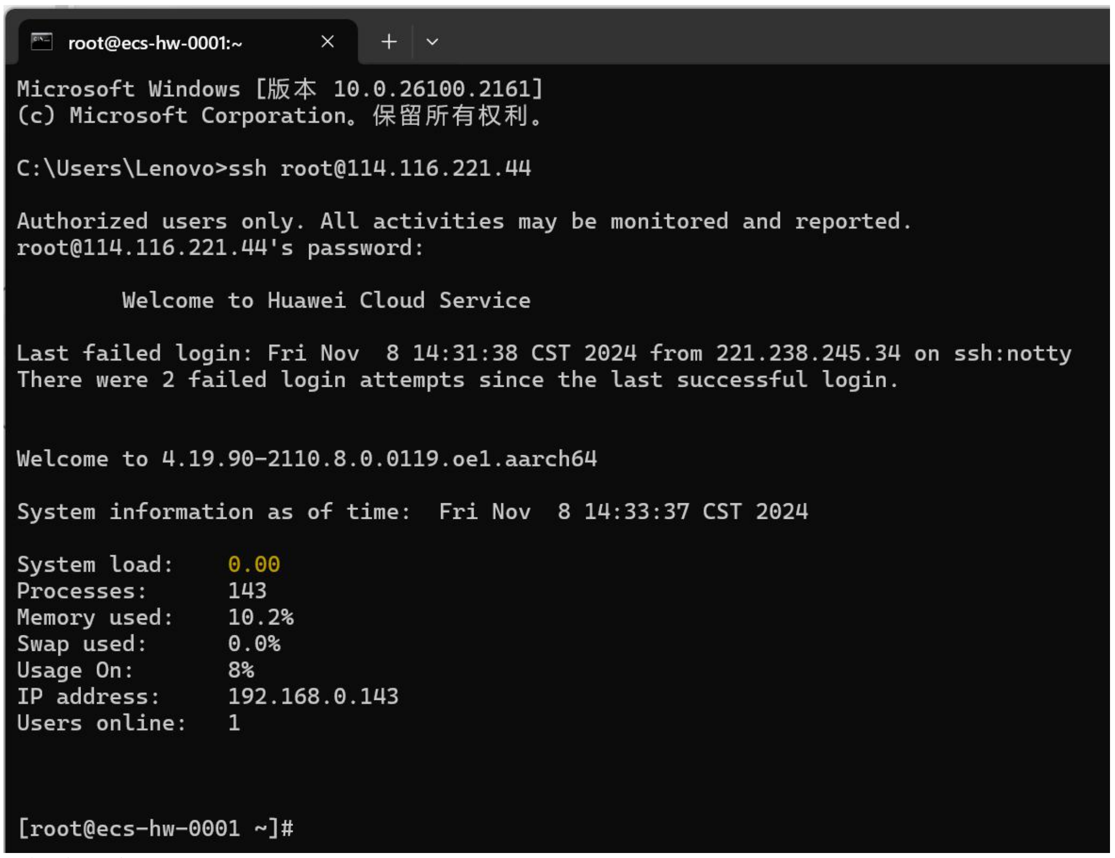
二、实验环境

华为鲲鹏云主机（openEuler20.03 操作系统） 4 台。

三、环境配置（四台主机上重复操作）

1.使用 ssh 工具登录 ecs（openssh）

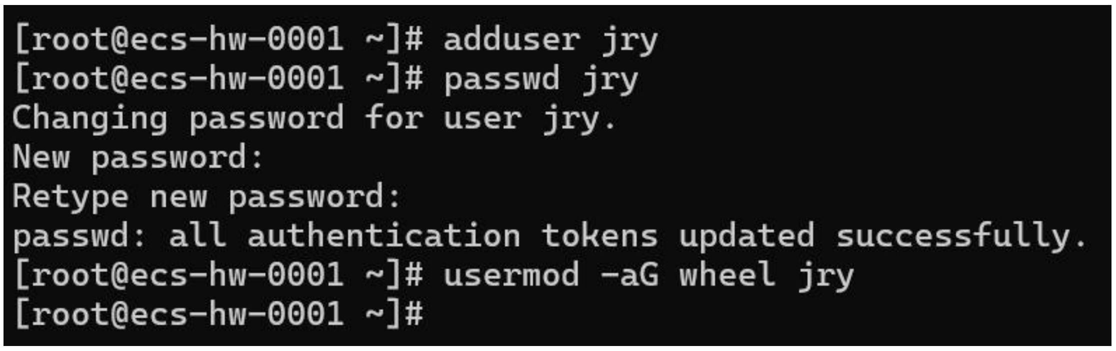
```
ssh root@114.116.221.44
```



```
Microsoft Windows [版本 10.0.26100.2161]
(c) Microsoft Corporation。保留所有权利。

C:\Users\Lenovo>ssh root@114.116.221.44

Authorized users only. All activities may be monitored and reported.
root@114.116.221.44's password:

        Welcome to Huawei Cloud Service

Last failed login: Fri Nov  8 14:31:38 CST 2024 from 221.238.245.34 on ssh:notty
There were 2 failed login attempts since the last successful login.


Welcome to 4.19.90-2110.8.0.0119.oe1.aarch64

System information as of time:  Fri Nov  8 14:33:37 CST 2024

System load:    0.00
Processes:      143
Memory used:    10.2%
Swap used:      0.0%
Usage On:       8%
IP address:     192.168.0.143
Users online:   1



[root@ecs-hw-0001 ~]#
```

2.创建用户

```
adduser jry
passwd jry
usermod -aG wheel jry
```



```
[root@ecs-hw-0001 ~]# adduser jry
[root@ecs-hw-0001 ~]# passwd jry
Changing password for user jry.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ecs-hw-0001 ~]# usermod -aG wheel jry
[root@ecs-hw-0001 ~]#
```

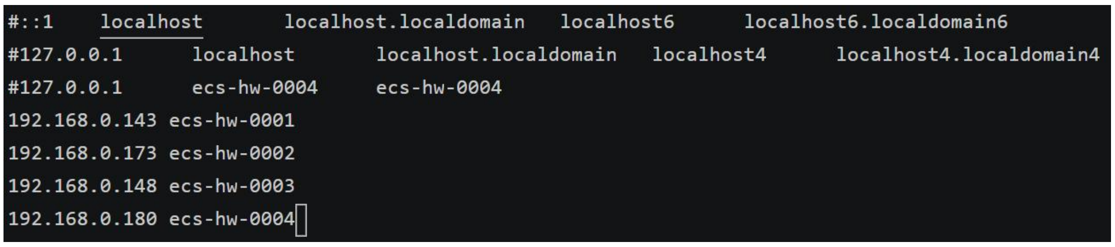
3.免密配置

（1）配置四台机器主机名和 ip 解析

```
vim /etc/hosts
```

注释文件原先本身的信息并添加以下信息（注意替换成四台主机的内网 ip）。

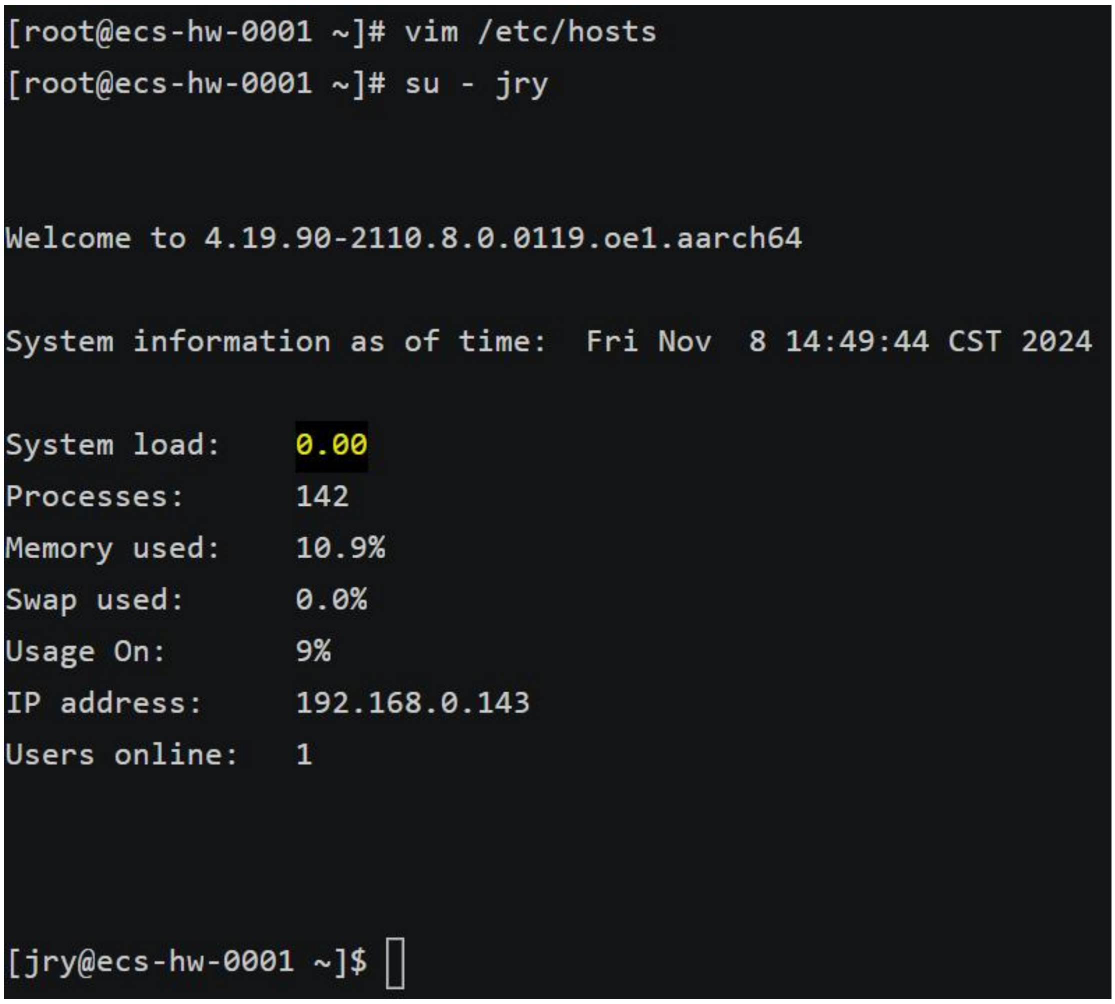
```
192.168.0.143 ecs-hw-0001
192.168.0.173 ecs-hw-0002
192.168.0.148 ecs-hw-0003
192.168.0.180 ecs-hw-0004
```



```
#::1      localhost        localhost.localdomain    localhost6       localhost6.localdomain6
#127.0.0.1       localhost        localhost.localdomain    localhost4       localhost4.localdomain4
#127.0.0.1       ecs-hw-0004      ecs-hw-0004
192.168.0.143 ecs-hw-0001
192.168.0.173 ecs-hw-0002
192.168.0.148 ecs-hw-0003
192.168.0.180 ecs-hw-0004
```

（2）退出 root 账户，重新登录到新建立的账户下

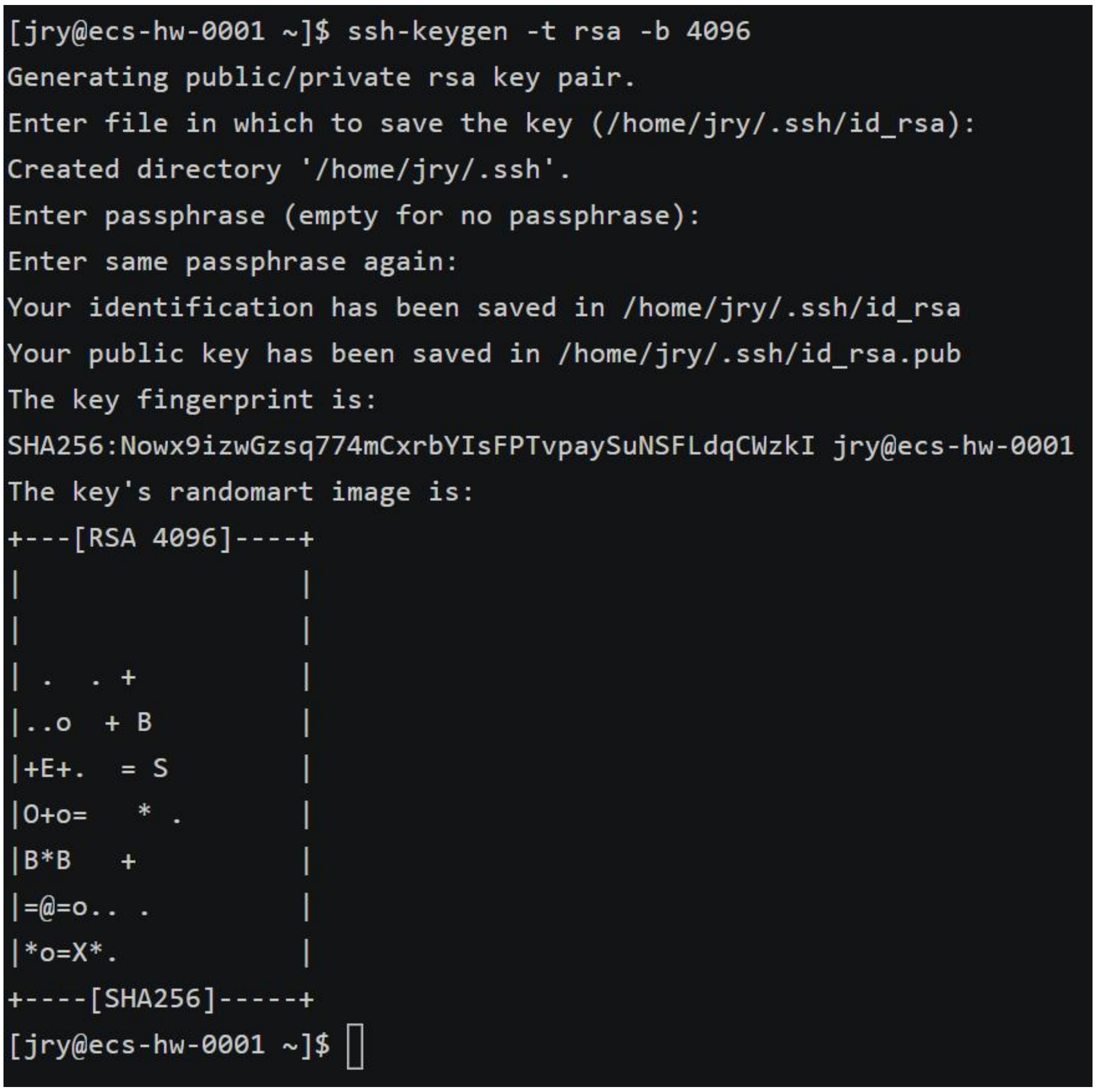
```
su - jry
```



```
[root@ecs-hw-0001 ~]# vim /etc/hosts
[root@ecs-hw-0001 ~]# su - jry


Welcome to 4.19.90-2110.8.0.0119.oe1.aarch64

System information as of time:  Fri Nov  8 14:49:44 CST 2024

System load:     0.00
Processes:       142
Memory used:     10.9%
Swap used:       0.0%
Usage On:        9%
IP address:      192.168.0.143
Users online:    1



[jry@ecs-hw-0001 ~]$
```

（3）本地生成密钥

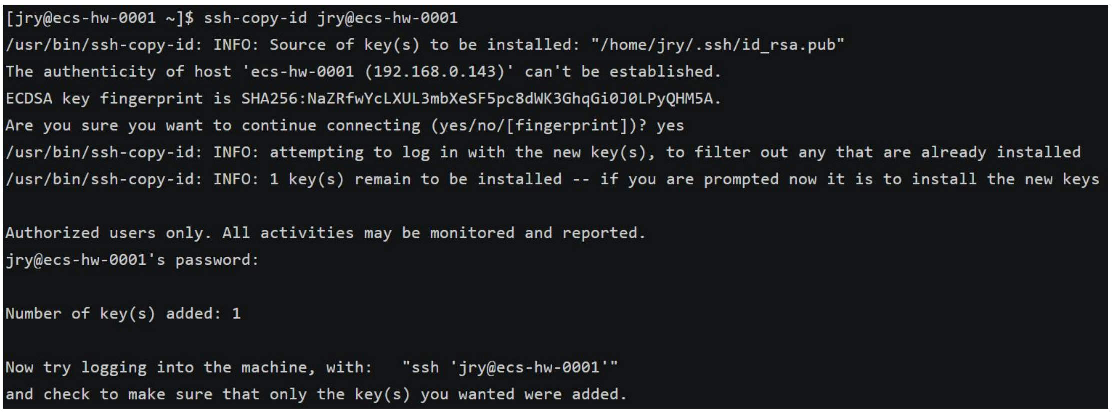
```
ssh-keygen -t rsa -b 4096
```



```
[jry@ecs-hw-0001 ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/jry/.ssh/id_rsa):
Created directory '/home/jry/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/jry/.ssh/id_rsa
Your public key has been saved in /home/jry/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Nowx9izwGzsq774mCxrbYIsFPTvpaySuNSFLdqCWzkI jry@ecs-hw-0001
The key's randomart image is:
+---[RSA 4096]----+
|                 |
|                 |
| .  . +          |
|..o  + B         |
|+E+.  = S        |
|O+o=   * .       |
|B*B   +          |
|=@=o.. .         |
|*o=X*.           |
+----[SHA256]-----+
[jry@ecs-hw-0001 ~]$ 
```

（4）添加公钥至所有主机

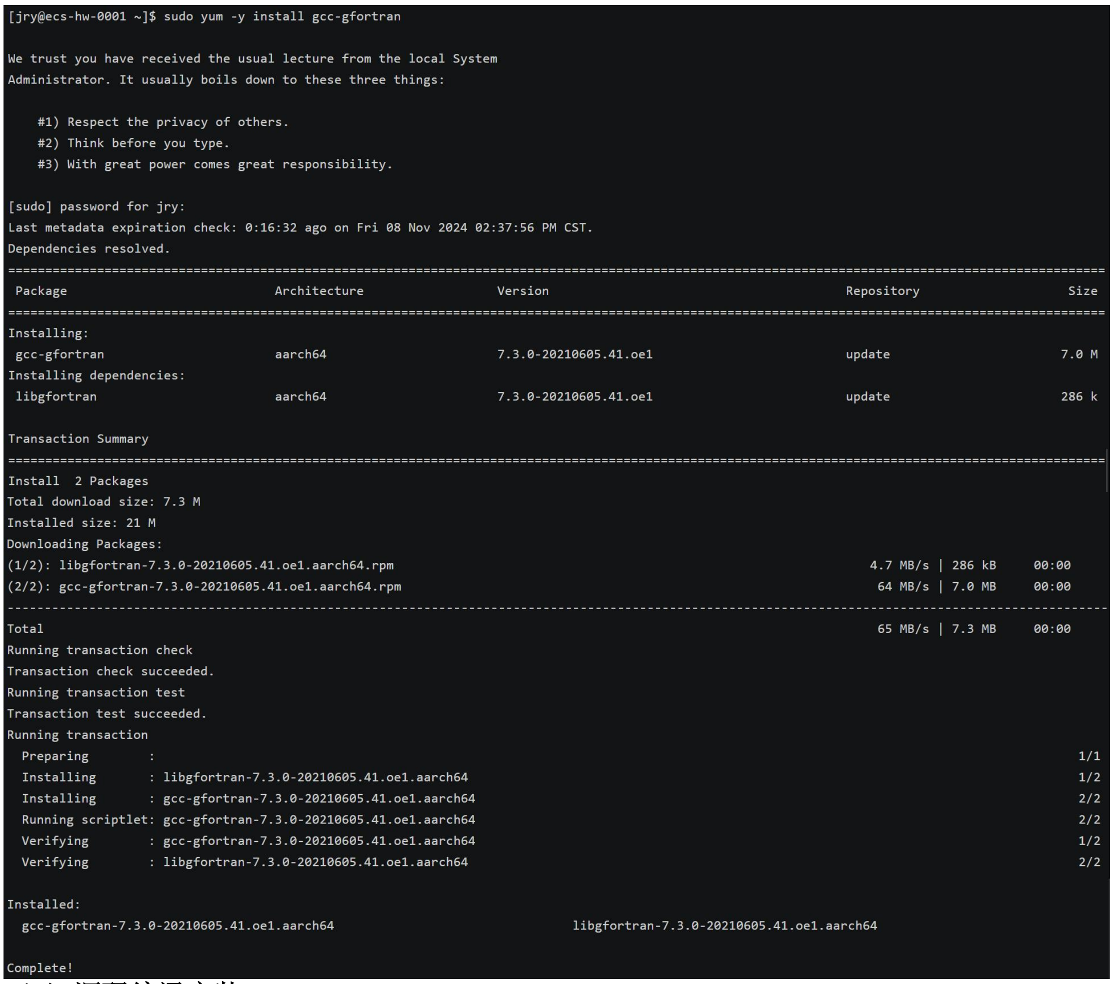
```
ssh-copy-id jry@ecs-hw-0001
ssh-copy-id jry@ecs-hw-0002
ssh-copy-id jry@ecs-hw-0003
ssh-copy-id jry@ecs-hw-0004
```



```
[jry@ecs-hw-0001 ~]$ ssh-copy-id jry@ecs-hw-0001
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/jry/.ssh/id_rsa.pub"
The authenticity of host 'ecs-hw-0001 (192.168.0.143)' can't be established.
ECDSA key fingerprint is SHA256:NaZRfwYcLXUL3mbXeSF5pc8dWK3GhqGi0J0LPyQHM5A.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys

Authorized users only. All activities may be monitored and reported.
jry@ecs-hw-0001's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'jry@ecs-hw-0001'"
and check to make sure that only the key(s) you wanted were added.
```

（5）安装依赖包

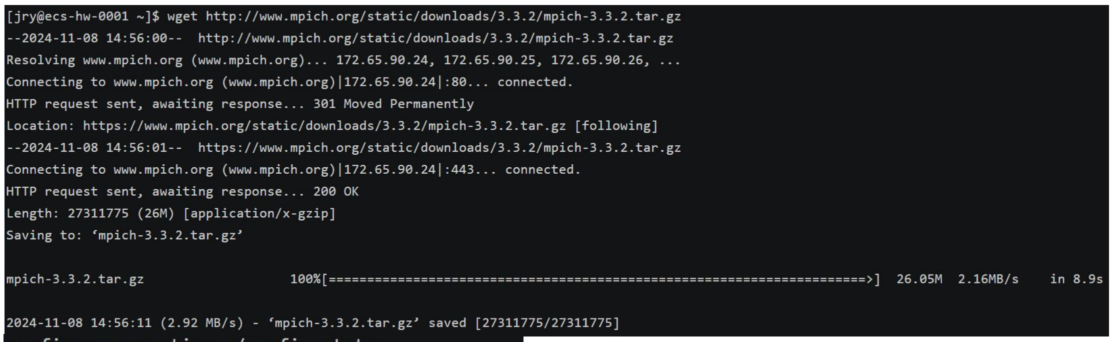
```
sudo yum -y install gcc-gfortran
sudo yum -y install gcc-c++
```



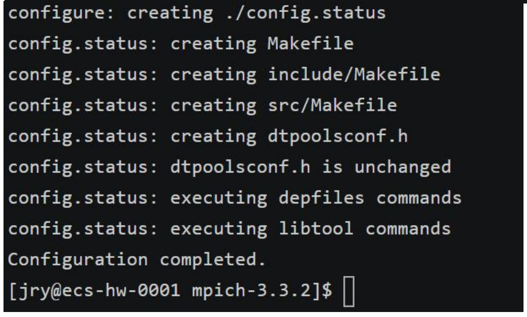
```
[jry@ecs-hw-0001 ~]$ sudo yum -y install gcc-gfortran

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for jry:
Last metadata expiration check: 0:16:32 ago on Fri 08 Nov 2024 02:37:56 PM CST.
Dependencies resolved.
======================================================================================================================================================
 Package                            Architecture                  Version                                        Repository                      Size
======================================================================================================================================================
Installing:
 gcc-gfortran                       aarch64                       7.3.0-20210605.41.oe1                          update                         7.0 M
Installing dependencies:
 libgfortran                        aarch64                       7.3.0-20210605.41.oe1                          update                         286 k

Transaction Summary
======================================================================================================================================================
Install  2 Packages
Total download size: 7.3 M
Installed size: 21 M
Downloading Packages:
(1/2): libgfortran-7.3.0-20210605.41.oe1.aarch64.rpm                                                                 4.7 MB/s | 286 kB     00:00
(2/2): gcc-gfortran-7.3.0-20210605.41.oe1.aarch64.rpm                                                                 64 MB/s | 7.0 MB     00:00
------------------------------------------------------------------------------------------------------------------------------------------------------
Total                                                                                                                 65 MB/s | 7.3 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                                                          1/1
  Installing       : libgfortran-7.3.0-20210605.41.oe1.aarch64                                                                                1/2
  Installing       : gcc-gfortran-7.3.0-20210605.41.oe1.aarch64                                                                               2/2
  Running scriptlet: gcc-gfortran-7.3.0-20210605.41.oe1.aarch64                                                                               2/2
  Verifying        : gcc-gfortran-7.3.0-20210605.41.oe1.aarch64                                                                               1/2
  Verifying        : libgfortran-7.3.0-20210605.41.oe1.aarch64                                                                                2/2

Installed:
  gcc-gfortran-7.3.0-20210605.41.oe1.aarch64                                 libgfortran-7.3.0-20210605.41.oe1.aarch64

Complete!
```

（6）源码编译安装 mpi

```
wget http://www.mpich.org/static/downloads/3.3.2/mpich-3.3.2.tar.gz
tar -zxvf mpich-3.3.2.tar.gz
cd mpich-3.3.2
./configure
sudo make && sudo make install
```



```
[jry@ecs-hw-0001 ~]$ wget http://www.mpich.org/static/downloads/3.3.2/mpich-3.3.2.tar.gz
--2024-11-08 14:56:00--  http://www.mpich.org/static/downloads/3.3.2/mpich-3.3.2.tar.gz
Resolving www.mpich.org (www.mpich.org)... 172.65.90.24, 172.65.90.25, 172.65.90.26, ...
Connecting to www.mpich.org (www.mpich.org)|172.65.90.24|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://www.mpich.org/static/downloads/3.3.2/mpich-3.3.2.tar.gz [following]
--2024-11-08 14:56:01--  https://www.mpich.org/static/downloads/3.3.2/mpich-3.3.2.tar.gz
Connecting to www.mpich.org (www.mpich.org)|172.65.90.24|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 27311775 (26M) [application/x-gzip]
Saving to: ‘mpich-3.3.2.tar.gz’

mpich-3.3.2.tar.gz                 100%[======================================================================>]  26.05M  2.16MB/s    in 8.9s

2024-11-08 14:56:11 (2.92 MB/s) - ‘mpich-3.3.2.tar.gz’ saved [27311775/27311775]
```



```
configure: creating ./config.status
config.status: creating Makefile
config.status: creating include/Makefile
config.status: creating src/Makefile
config.status: creating dtpoolsconf.h
config.status: dtpoolsconf.h is unchanged
config.status: executing depfiles commands
config.status: executing libtool commands
Configuration completed.
[jry@ecs-hw-0001 mpich-3.3.2]$
```

四、实验内容

1.创建示例程序源码（四台主机上重复操作，以下示例步骤均在 ecs-hw-0001 上执行）

执行以下命令，创建 hello 目录存放该程序的所有文件, 并进入 hello 目录，创建示例程序源码 mpi_hello_world.c。

```
mkdir /home/jry/hello
cd /home/jry/hello
vim mpi_hello_world.c
```

代码内容如下：

```
#include <mpi.h>
#include <stdio.h>
int main(int argc, char** argv) {
MPI_Init(NULL, NULL);
int world_size;
MPI_Comm_size(MPI_COMM_WORLD, &world_size);
int world_rank;
MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);
char processor_name[MPI_MAX_PROCESSOR_NAME];
int name_len;
MPI_Get_processor_name(processor_name, &name_len);
printf("Hello world from processor %s, rank %d out of %d processors\n",processor_name,
world_rank, world_size);
MPI_Finalize();
}
```

2.创建 makefile（四台主机上重复操作）

执行以下命令，创建 makefile。

```
vim makefile
```

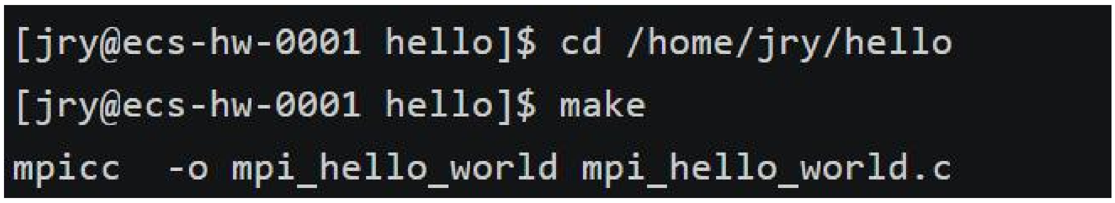
代码内容如下：

```
EXECS=mpi_hello_world
MPICC?=mpicc
all: ${EXECS}
mpi_hello_world: mpi_hello_world.c
	${MPICC} -o mpi_hello_world mpi_hello_world.c
clean:
	rm -f ${EXECS}
```

3.进行编译（四台主机上重复操作）

执行以下命令，进行编译。

```
cd /home/jry/hello
make
```



```
[jry@ecs-hw-0001 hello]$ cd /home/jry/hello
[jry@ecs-hw-0001 hello]$ make
mpicc  -o mpi_hello_world mpi_hello_world.c
```

4.建立主机配置文件（四台主机上重复操作）

执行以下命令，建立主机配置文件。

```
vim /home/jry/hello/config
```

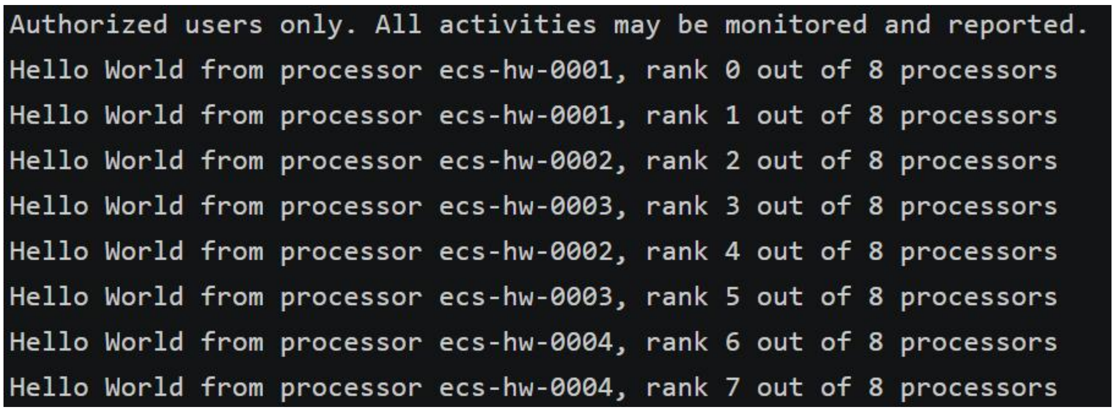
添加内容如下：

```
ecs-hw-0001:2
ecs-hw-0002:2
ecs-hw-0003:2
ecs-hw-0004:2
```

5.运行监测（只需要在 ecs-hw-0001 上执行）

执行以下命令，查看运行结果

```
mpiexec -n 8 -f /home/jry/hello/config /home/jry/hello/mpi_hello_world
```



```
Authorized users only. All activities may be monitored and reported.
Hello World from processor ecs-hw-0001, rank 0 out of 8 processors
Hello World from processor ecs-hw-0001, rank 1 out of 8 processors
Hello World from processor ecs-hw-0002, rank 2 out of 8 processors
Hello World from processor ecs-hw-0003, rank 3 out of 8 processors
Hello World from processor ecs-hw-0002, rank 4 out of 8 processors
Hello World from processor ecs-hw-0003, rank 5 out of 8 processors
Hello World from processor ecs-hw-0004, rank 6 out of 8 processors
Hello World from processor ecs-hw-0004, rank 7 out of 8 processors
```

五、思考题

1.集群之间如果彼此不配置信任秘钥，程序能否正常运行？

答：不能。①信任秘钥的配置是为了确保各个节点之间的通信安全，防止未经授权的访问和数据泄露。②在没有配置信任秘钥的情况下，节点之间的通信可能会因为缺乏身份验证而被拒绝，导致程序无法正确执行并行计算任务。