

# **LAPORAN PEMROSESAN PARALEL**

(Numerik Python Menggunakan MPI Secara Paralel)



Disusun Oleh:

Pratama Arjan Rangkuti (09011182227006)

Dosen Pengampu:

Adi Hermansyah, S.Kom., M.T

**FAKULTAS ILMU KOMPUTER  
PROGRAM STUDI SISTEM KOMPUTER  
UNIVERSITAS SRIWIJAYA**

**2023**

## 1. Upgrade OS

Menggunakan command '*sudo apt update && sudo apt upgrade*' untuk memperbarui OS pada setiap device

```
worker3@master:~$ sudo apt update && sudo apt upgrade
```

## 2. Install net-tools

Melakukan penginstalan net-tools dengan menggunakan command '*sudo apt install net-tools*' untuk mengecek IP

```
worker3@master:~$ sudo apt instal net-tools
```

## 3. Cek ip

Menggunakan command '*ifconfig*' atau '*hostname -I*' untuk mengecek IP masing masing device

```
worker3@master:~$ hostname -I
192.168.18.193
```

## 4. Konfigurasi file

Membuka file /etc/hosts menggunakan command '*sudo nano /etc/hosts*'

```
worker3@master:~$ sudo nano /etc/hosts
```

Lalu mengedit file dengan menambahkan IP dan peran.

```
GNU nano 6.2 /etc/hosts
192.168.199.71 master localhost
127.0.1.1 worker3-VirtualBox

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

192.168.199.138 slave1
192.168.199.172 slave2
```

## 5. User bersama

Membuat user baru bersama dengan command '*sudo adduser <nama user>*'

```
worker3@master:~$ sudo adduser pempar2
```

### ○ User root

Memberi akses root ke user yang baru ditambahkan dengan command '*sudo usermod -aG sudo <nama user>*'

```
worker3@master:~$ sudo usermod -aG sudo pempar2
```

### ○ Login user

Masuk ke user dengan command '*su - <nama user>*'

```
worker3@master:~$ su - pempar2
```

## 6. Konfigurasi SSH

Melakukan konfigurasi SSH, SSH(Secure Shell) digunakan untuk otentikasi dan pertukaran data aman antara node dalam cluster MPI.

### ○ Install SSH

Melakukan penginstallan SSH dengan command '*sudo apt install openssh-server*'

```
pempar2@master:~$ sudo apt install openssh-server
```

Dapat dilakukan pengecekan SSH untuk menghubungkan master ke klien, dengan command '*ssh <nama user>@<host>*'

### ○ Generate keygen

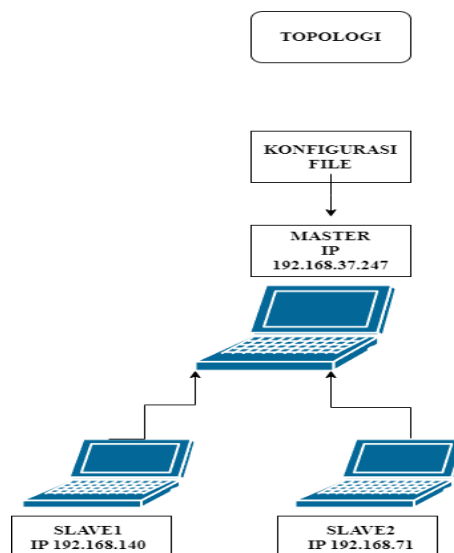
Dilakukan di master, menggunakan command '*ssh-keygen -t rsa*'

```
pempar2@master:~$ ssh-keygen -t rsa
```

### ○ Input key publik ke klien

Dilakukan di master, membuat isi dari file *id\_rsa.pub* disalin ke file *authorized\_keys* menggunakan command '*ssh-copy-id <nama user>@<host>*'

```
pempar2@master:~$ ssh-copy-id pempar2@slave2
```



## 7. Konfigurasi NFS

Konfigurasi NFS (Network File System) merupakan proses mengatur dan mengkonfigurasi sistem berkas yang memungkinkan berbagi sistem berkas antara komputer dalam jaringan.

### ○ Shared folder

Membuat folder bersama menggunakan command '*mkdir <nama folder>*'

```
pempar2@master:~$ mkdir sk3
```

### ○ Install NFS Server

Menginstall NFS pada master dengan command '*sudo apt install nfs-kernel-server*'

```
pempar2@master:~$ sudo apt install nfs-kernel-server
```

### ○ Konfigurasi file

Dilakukan di master, buka file */etc/exports* dengan command '*sudo nano /etc/exports*'

```
pempar2@master:~$ sudo nano /etc/exports
```

Dan dilakukan pengeditan pada file, dengan menambahkan *<lokasi shared folder>*

*\*(rw, sync, no\_root\_squash, no\_subtree\_check)*

```
GNU nano 6.2 /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw, sync, no_subtree_check) hostname2(ro, sync, no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw, sync, fsid=0, crossmnt, no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw, sync, no_subtree_check)
#
/home/pempar2/sk3 *(rw, sync, no_root_squash, no_subtree_check)

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^N Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Lalu, untuk menyimpan ulang daftar direktori setelah mengedit file dapat menggunakan command '***sudo exportfs -a***' dan untuk memulai ulang layanan pada server dapat menggunakan command '***sudo systemctl restart nfs-kernel-server***'

```
pempar2@master:~$ sudo exportfs -a
pempar2@master:~$ sudo systemctl restart nfs-kernel-server
```

### ○ Install NFS Klient

Melakukan instalasi NFS pada worker dengan command '***sudo apt install nfscommon***'

```
pempar2@slave2-VirtualBox:~$ sudo apt install nfs-common
```

```
pempar2@alddiff-VirtualBox:~$ sudo apt install nfs-common
```

### ○ Mounting

Dilakukan pada worker dengan menggunakan command '***sudo mount <server host>:<lokasi shared folder di server> <lokasi shared folder di client>***'

```
pempar2@alddiff-VirtualBox:~$ sudo mount master:/home/pempar2/sk3 /home/pempar2/sk3
```

```
pempar2@slave2-VirtualBox:~$ sudo mount master:/home/pempar2/sk3 /home/pempar2/sk3
```

## 8. MPI

MPI adalah singkatan dari "Message Passing Interface." Ini adalah standar komunikasi yang digunakan dalam pemrograman paralel, terutama dalam pemrograman terdistribusi untuk sistem berbasis kluster atau superkomputer.

### ○ Install MPI

Melakukan instalasi MPI dengan command '***sudo apt install openmpi-bin libopenmpi-dev***'

```
pempar2@alddiff-VirtualBox:~$ sudo apt install openmpi-bin libopenmpi-dev
```

### ○ Testing

Dilakukan di master, membuat file python di folder sebelumnya, dengan command '***touch <nama file>.py***' Lalu, dapat melakukan pengeditan dalam file dengan menggunakan command '***nano <nama file>.py***'

```
pempar2@master:~$ cd /home/pempar2/sk3/  
pempar2@master:~/sk3$ touch tes.py  
pempar2@master:~/sk3$ nano tes.py
```

## 9. Konfigurasi python

### ○ Install python

Melakukan instalasi python versi 3 dengan menggunakan command '*sudo apt install python3-pip*'

```
pempar2@master:~$ sudo apt install python3-pip
```

### ○ Install pustaka MPI4

Melakukan instalasi pustaka MPI4 yang menyediakan dukungan untuk komunikasi dan pemrograman paralel menggunakan MPI (Message Passing Interface) dapat dilakukan dengan command '*pip install mpi4py*'

```
pempar2@master:~$ pip install mpi4py
```

## 10. Numerik

Numerik adalah sebuah simbol yang merepresentasikan sebuah nomor (angka) atau bilangan. Di dalam sistem numerik ini, penulisan angka berdasarkan posisi dan basis. Posisi suatu angka dalam sistem ini menentukan nilai dari bilangan yang diwakilinya. Maka notasi yang digunakan disebut notasi posisional.

Python adalah bahasa pemrograman yang banyak digunakan dalam aplikasi web, pengembangan perangkat lunak, ilmu data, dan machine learning (ML). Developer menggunakan Python karena efisien dan mudah dipelajari serta dapat dijalankan di berbagai platform.

Dalam bahasa pemrograman python, kita dapat menggunakan sebuah metode untuk menghitung suatu permasalahan matematis dan metode tersebut disebut dengan metode numerik. Metode numerik merupakan sebuah metode yang digunakan untuk menyelesaikan berbagai kasus dan masalah matematis melalui komputasi. Metode numerik pada umumnya melibatkan jumlah komputasi yang sangat banyak, dengan melibatkan proses iterasi hitungan. Oleh karena itu, komputasi metode numerik dilakukan dengan menggunakan bantuan komputer. Metode numerik saat ini juga banyak diaplikasikan dalam berbagai bidang, salah satunya dibidang engineering.

### ○ Menjalankan Numerik

Menjalankan file yang telah diedit dengan command '*mpirun -n <jumlah prosesor> host <daftar host> python3 <nama file>.py*'

Dari proses yang telah berjalan, maka akan menghasilkan output berupa pengurutan



dari data yang telah diinput pada codingan.

```
king@master:~$ mpirun -n 3 -host master,slave1,slave2 python3 /home/queen/yuk.py
Iterasi : 1, 7.416666666666667, Waktu per Iterasi: 3.361701965332031e-05 detik
Iterasi : 2, 5.056647940074907, Waktu per Iterasi: 0.0005714893341064453 detik
Iterasi : 3, 4.505918637197902, Waktu per Iterasi: 0.0009796619415283203 detik
Iterasi : 4, 4.472262596169605, Waktu per Iterasi: 0.0016503334045410156 detik
Iterasi : 5, 4.47213595679263, Waktu per Iterasi: 0.002307415008544922 detik

Jumlah iterasi : 6
Hasil akhir : 4.47213595679263
None
Iterasi : 1, 7.416666666666667, Waktu per Iterasi: 2.4080276489257812e-05 detik
Iterasi : 2, 5.056647940074907, Waktu per Iterasi: 0.00016689300537109375 detik
Iterasi : 3, 4.505918637197902, Waktu per Iterasi: 0.0002319812774658203 detik
Iterasi : 4, 4.472262596169605, Waktu per Iterasi: 0.0002923011779785156 detik
Iterasi : 5, 4.47213595679263, Waktu per Iterasi: 0.00035119056701660156 detik

Jumlah iterasi : 6
Hasil akhir : 4.47213595679263
None
Iterasi : 1, 7.416666666666667, Waktu per Iterasi: 3.552436828613281e-05 detik
Iterasi : 2, 5.056647940074907, Waktu per Iterasi: 0.0004830360412597656 detik
Iterasi : 3, 4.505918637197902, Waktu per Iterasi: 0.0011119842529296875 detik
Iterasi : 4, 4.472262596169605, Waktu per Iterasi: 0.0014526844024658203 detik
Iterasi : 5, 4.47213595679263, Waktu per Iterasi: 0.0018298625946044922 detik

Jumlah iterasi : 6
Hasil akhir : 4.47213595679263
None
```

Selain itu, kita dapat pula menggunakan pyhton untuk menjalankan file yang dituju dengan sebuah command ***'python3 <nama file>.py'***

```
king@master:~$ python3 /home/queen/yuk.py
Iterasi : 1, 7.416666666666667, Waktu per Iterasi: 2.4557113647460938e-05 detik
Iterasi : 2, 5.056647940074907, Waktu per Iterasi: 0.0001239776611328125 detik
Iterasi : 3, 4.505918637197902, Waktu per Iterasi: 0.00016069412231445312 detik
Iterasi : 4, 4.472262596169605, Waktu per Iterasi: 0.00019502639770507812 detik
Iterasi : 5, 4.47213595679263, Waktu per Iterasi: 0.00022864341735839844 detik

Jumlah iterasi : 6
Hasil akhir : 4.47213595679263
None
```