TUGAS 4
PEMROGRAMAN JARINGAN - E

05111840000162 - Fransiskus Xaverius Kevin Koesnadi

Dosen Kelas : Royyana Muslim Ijtihadie, S. Kom., M. Kom., Ph. D.

Departemen Teknik Informatika

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

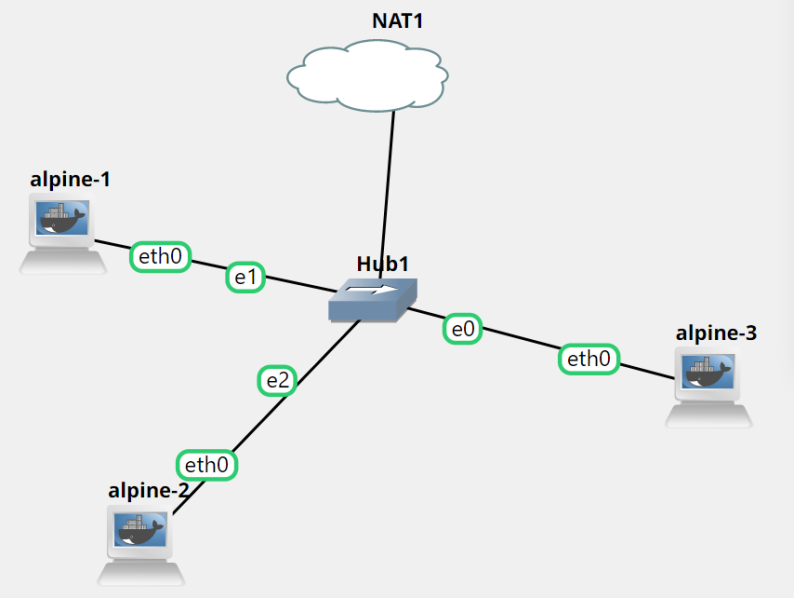
Surabaya

2021

Semua tugas berikut ini HARUS dijalankan di SIMULATOR

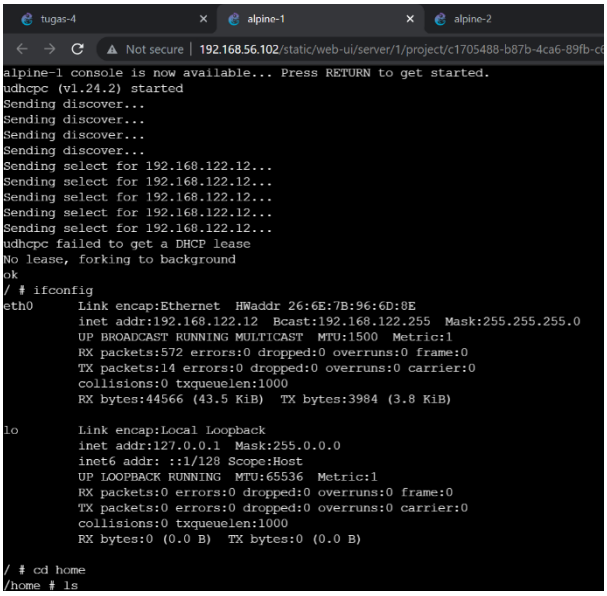
1. Jalankan program server seperti dalam pembahasan
2. Buatlah program client yang dapat melakukan 100 request get pada dalam satu saat untuk operasi get file "pokijan.jpg"
3. Capture dan submitlah poin 1 dan 2 dalam satu dokumen pdf. berikan DESKRIPSI dan PENJELASAN

Gambar arsitektur jaringan:

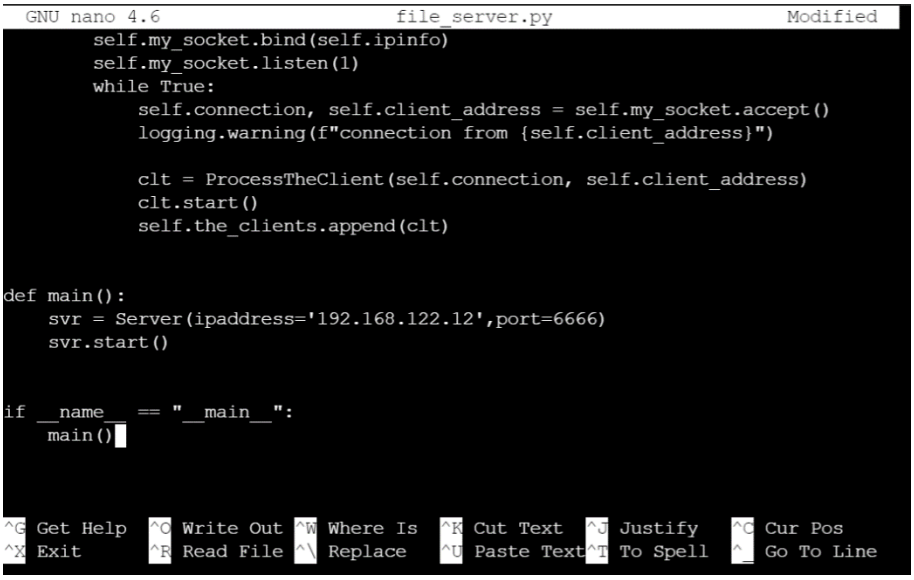


Penyelesaian:

1. Periksa *ip address* dari alpine-1 sebagai server dengan mengetikkan *command* `ifconfig`. Hal ini dilakukan karena alpine-1 akan menjadi server serta melihat port yang digunakan sebagai penguhubung antara *client* dan server.



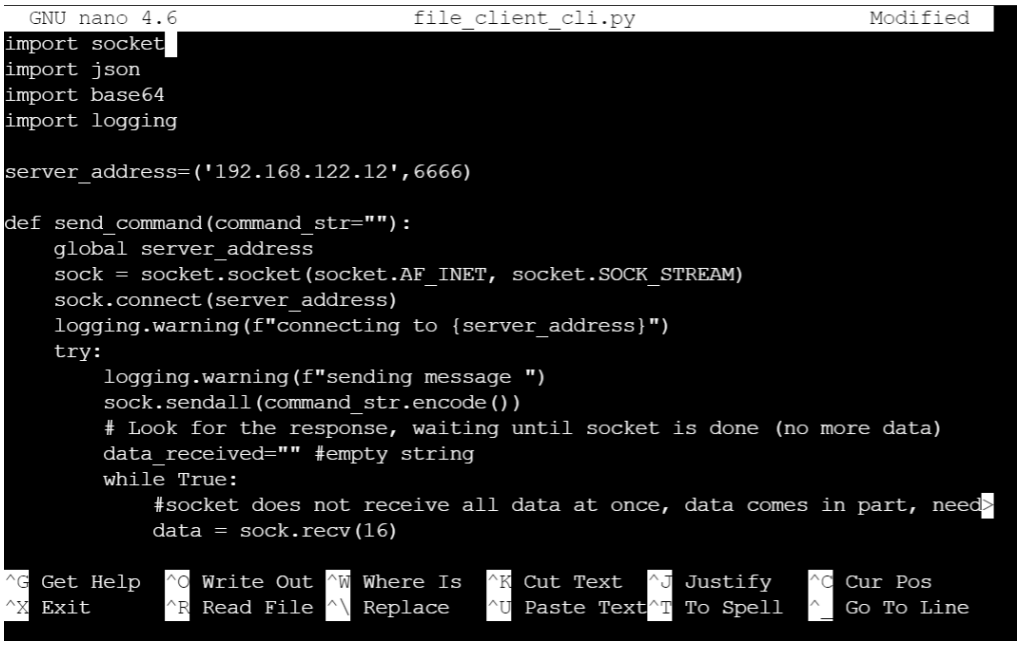
2. Lakukan pengubahan `ipaddres` di file_server.py berdasarkan hasil *ip address* yang ditemukan sebelumnya.



Ketika dijalankan akan ditemukan *ip address* dan *port* yang menginformasikan server sudah berjalan.

```
/home/Pemrograman_Jaringan_E/progjar4a/ans # python3 file_server.py
WARNING:root:server berjalan di ip address ('192.168.122.12', 6666)
```

3. Lakukan pengubahan `server address` di file_client_cli.py berdasarkan hasil *ip address* yang ditemukan sebelumnya.



4. Lakukan multi process dengan membuat fungsi baru. Dalam hal ini fungsi akan diletakkan pada file yang berbeda dan diletakkan pada multiprocess.py.

```python
from file_client_cli import remote_get
import time
import datetime
from multiprocessing import Process

def req100():
    texec = dict()
    catat_awal = datetime.datetime.now()
    filename = 'pokijan.jpg'
    for k in range(100):
        print(f"file - {k}")
        waktu = time.time()
        texec[k] = Process(target=remote_get, args=(filename,))
        texec[k].start()
    for k in range(100):
        texec[k].join()
    catat_akhir = datetime.datetime.now()
    selesai = catat_akhir - catat_awal
    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")

if __name__ == '__main__':
    req100()
```

Hal yang dilakukan adalah fungsi `req100` akan mengambil variabel `filename` sebagai *file* yang dimaksudkan lalu akan dimasukkan pada perulangan sebanyak 100 kali pokijan.jpg untuk pemanggilan fungsi `remote_get` dan diberikan *debugging* `print` untuk melihat iterasi.

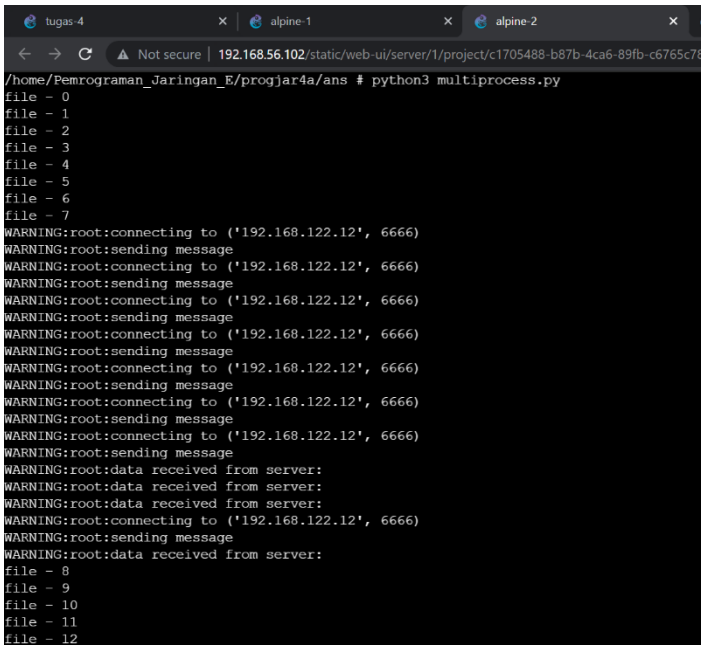
```python
filename = 'pokijan.jpg'
for k in range(100):
    print(f"file - {k}")
```

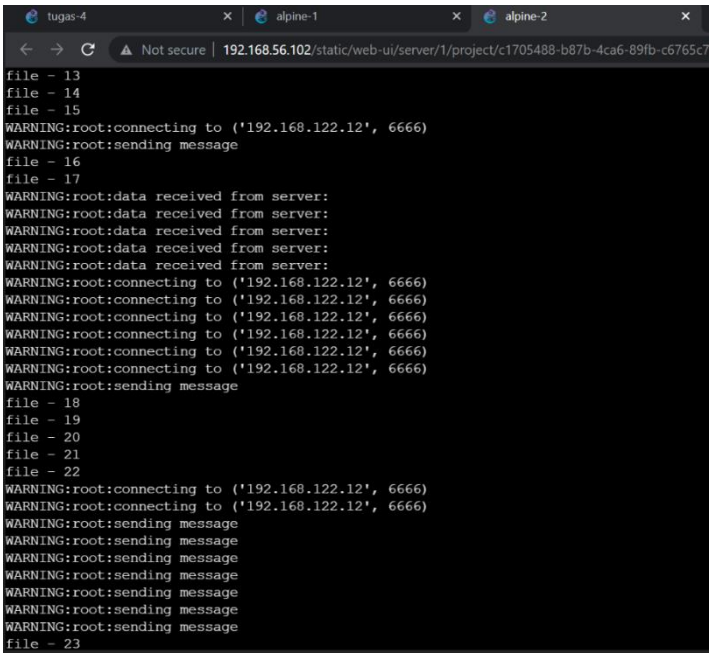
Dengan multiprocess diambil sebanyak 100 kali pada `Process(target=remote_get, args=(filename,))`

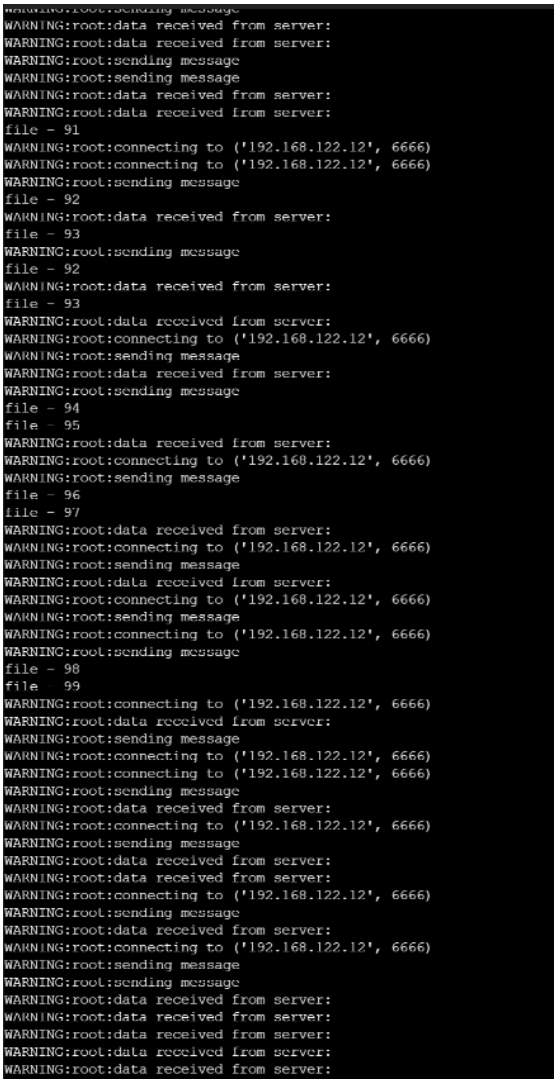
```python
    texec[k] = Process(target=remote_get, args=(filename,))
```

Pada file_client_cli.py akan dilakukan pemanggilan fungsi `remote_get` berdasarkan argumen `filename` melalui multi process.

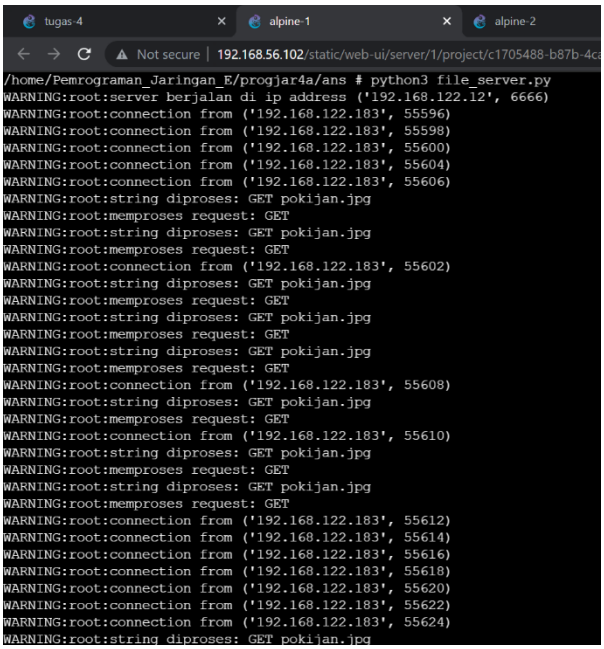
5. *Run* multiprocess.py pada alpine-2 sebagai *client* dan menghasilkan pemanggilan `remote_get` sebanyak 100 kali pokijan.jpg dan *run* file_server.py pada alpine-1 sebagai server.
    - alpine-2 (*client*)

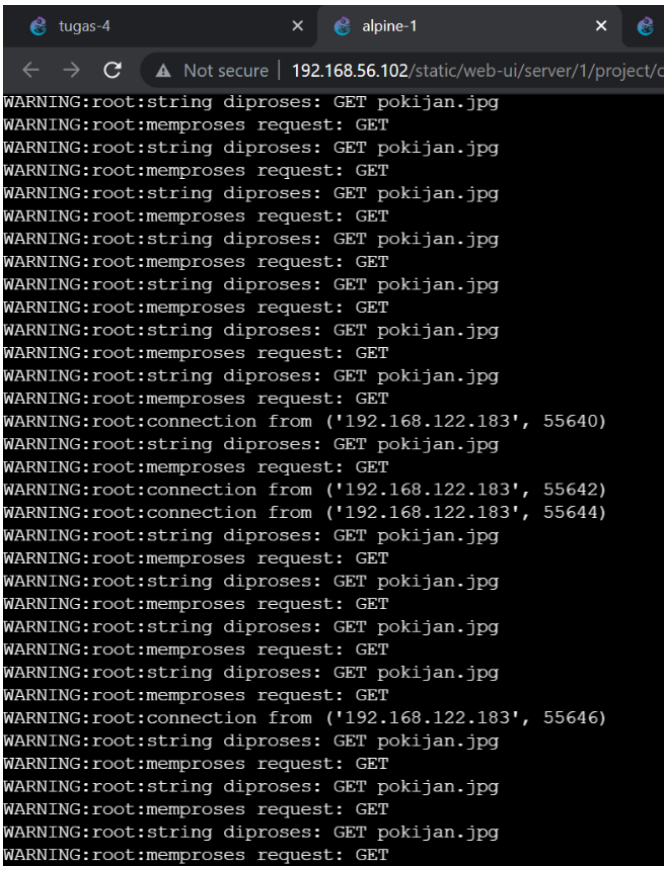






- alpine-1 (server)





Tautan GitHub: FXKevinK/Pemrograman_Jaringan_E at Tugas_4 (github.com)