

Laporan Tugas 3

Nama : Joseph Eric Amadeo Seloatmodjo

NRP : 05111840000077

Kelas : Progar E

Soal:

Buatlah program yang mengimplementasikan

1. Multi Process
2. Multi Thread
3. Multi Process Asynchronous
4. Multi Thread Asynchronous

dengan menggunakan protokol transport UDP. kasus dapat didefinisikan sendiri dan Buatlah arsitektur jaringan anda sendiri di simulator GNS3.

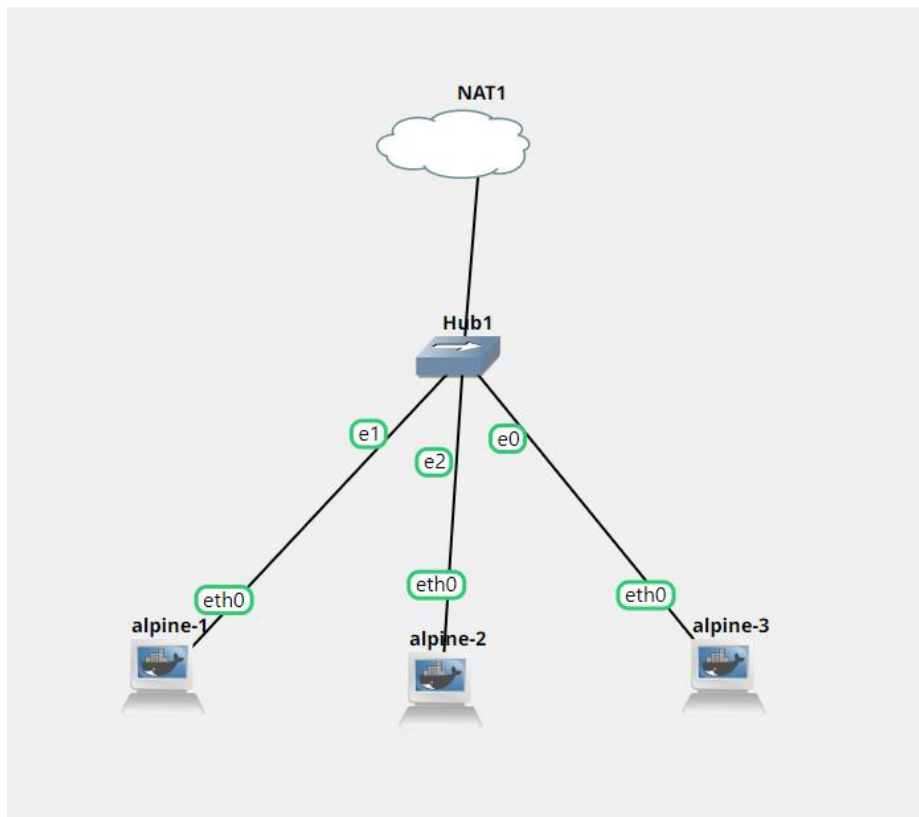
Buatlah laporan dalam bentuk PDF yang berisikan screenshot dari

1. Deskripsi kasus yang dibuat
2. Gambar arsitektur jaringan (dalam simulator gns3)
3. Program yang dibuat (1-4)
4. Hasil outputnya

1. Kasus

Terdapat 2 client dan 1 server. Kedua client akan mendownload beberapa file gambar. Setelah itu kedua client mengirimkan file tersebut ke server. Kedua client ini akan menggunakan metode pengiriman yang berbeda, yaitu multiprocess, multithreaded, asynchronous multiprocess, dan asynchronous multithreaded. Setelah itu kita akan melihat perbedaan performa dari kedua client dan cara pengiriman yang berbeda ini.

2. Gambar Arsitektur Jaringan



3. Program yang Dibuat

Dapat dilihat di [https://github.com/josepheric/Pemrograman Jaringan E/tree/Tugas 3](https://github.com/josepheric/Pemrograman_Jaringan_E/tree/Tugas_3)

Library.py

```

import logging
import requests
import socket
import os
import time
import datetime

def get_url_list():
    urls = dict()
    urls['kompas'] = 'https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsfbSpec=/0x0:998x665/740x500/data/photo/2020/03/01/5e5b52f4db896.jpg'
    urls['cat'] = 'https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg'
    return urls

def download_gambar(url=None, tuliskefile='image'):
    waktu_awal = datetime.datetime.now()
    if (url is None):
        return False
    ff = requests.get(url)

```

```

tipe = dict()
tipe['image/png']='png'
tipe['image/jpg']='jpg'
tipe['image/gif']='gif'
tipe['image/jpeg']='jpg'
tipe['application/zip']='jpg'
tipe['video/quicktime']='mov'
#     time.sleep(2) #untuk simulasi, diberi tambahan delay 2 detik

content_type = ff.headers['Content-Type']
logging.warning(content_type)
if (content_type in list(tipe.keys())):
    namafile = os.path.basename(url)
    ekstensi = tipe[content_type]
    if (tuliskefile):
        fp = open(f"{tuliskefile}.{ekstensi}", "wb")
        fp.write(ff.content)
        fp.close()

        waktu_process = datetime.datetime.now() - waktu_awal
        waktu_akhir =datetime.datetime.now()
        logging.warning(f"writing {tuliskefile}.{ekstensi} dalam waktu {waktu_
process} {waktu_awal} s/d {waktu_akhir}")
        return waktu_process
    else:
        return False

def send_file(IP_ADDRESS, PORT, filename):
    print(IP_ADDRESS, PORT, filename)
    clientSock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

    fp=open(filename,'rb')
    k=fp.read()
    ter kirim=0
    for x in k:
        k_bytes=bytes([x])
        clientSock.sendto(k_bytes,(IP_ADDRESS,PORT))
        ter kirim=ter kirim+1

if __name__=='__main__':
    #check fungsi
    k = download_gambar('https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsf
bSpec=/0x0:998x665/740x500/data/photo/2020/03/01/5e5b52f4db896.jpg')
    print(k)

```

multi_process_async.py

```

from library import download_gambar, get_url_list, send_file
import time
import datetime
from multiprocessing import Process, Pool

IP_SERVER = "192.168.122.244"
PORT = 5050

def send_all():
    texec = dict()
    urls = get_url_list()
    status_task = dict()
    task_pool = Pool(processes=20) #2 task yang dapat dikerjakan secara simultan, dapat diset sesuai jumlah core
    catat_awal = datetime.datetime.now()
    for k in urls:
        download_gambar(urls[k],k)
        print(f"mendownload {urls[k]}")
        texec[k] = task_pool.apply_async(func=send_file, args=(IP_SERVER, PORT, f"{k}.jpg"))
        print('send to server succesful')

    #setelah menyelesaikan tugasnya, dikembalikan ke main process dengan mengambil hasilnya dengan get
    for k in urls:
        status_task[k]=texec[k].get(timeout=10)

    catat_akhir = datetime.datetime.now()
    selesai = catat_akhir - catat_awal
    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")
    print("status TASK")
    print(status_task)

#fungsi download_gambar akan dijalankan secara multi process

if __name__=='__main__':
    send_all()

```

multi_process.py

```

from library import download_gambar, get_url_list, send_file
import time
import datetime
from multiprocessing import Process

```

```

IP_SERVER = "192.168.122.244"
PORT = 5050

def send_all():
    texec = dict()
    urls = get_url_list()
    catat_awal = datetime.datetime.now()
    for k in urls:
        download_gambar(urls[k],k)
        print(f"mendownload {urls[k]}")
        waktu = time.time()
        #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi down
load gambar secara multiprocessing
        texec[k] = Process(target=send_file, args=(IP_SERVER,PORT,f"{k}.jpg"))
        print('send to server succesfully')
        texec[k].start()
        #setelah menyelesaikan tugasnya, dikembalikan ke main process dengan join
    for k in urls:
        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 {cata
t_akhir}")
#fungsi download_gambar akan dijalankan secara multi process
if __name__=='__main__':
    send_all()

```

multi_thread_async.py

```

from library import download_gambar,get_url_list, send_file
import time
import datetime
import concurrent.futures

IP_SERVER = "192.168.122.244"
PORT = 5050

def send_all():
    texec = dict()
    urls = get_url_list()
    status_task = dict()
    task = concurrent.futures.ThreadPoolExecutor(max_workers=4)
    catat_awal = datetime.datetime.now()
    for k in urls:
        download_gambar(urls[k], k)
        print(f"mendownload {urls[k]}")
        texec[k] = task.submit(send_file,IP_SERVER, PORT, f"{k}.jpg")

```

```

        print('send file succesfully')

    #setelah menyelesaikan tugasnya, dikembalikan ke main thread dengan memang
gil result
    for k in urls:
        status_task[k]=texec[k].result()

    catat_akhir = datetime.datetime.now()
    selesai = catat_akhir - catat_awal
    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {cata
t_akhir}")
    print("hasil task yang dijalankan")
    print(status_task)

#fungsi download_gambar akan dijalankan secara multithreading

if __name__=='__main__':
    send_all()

```

multi_thread.py

```

from library import download_gambar,get_url_list,send_file
import time
import datetime
import threading

IP_SERVER = "192.168.122.244"
PORT = 5050

def send_all():
    texec = dict()
    urls = get_url_list()
    catat_awal = datetime.datetime.now()
    for k in urls:
        download_gambar(urls[k], k)
        print(f"mendownload {urls[k]}")
        #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi down
load gambar secara multithread
        texec[k] = threading.Thread(target=send_file, args=(IP_SERVER, PORT, f
"{k}.jpg"))
        print('send file succesfully')
        texec[k].start()

    #setelah menyelesaikan tugasnya, dikembalikan ke main thread dengan join
    for k in urls:
        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 {cata
t_akhir}")

#fungsi download_gambar akan dijalankan secara multithreading

if __name__=='__main__':
    send_all()

```

server.py

```

# Server
import socket

IP_ADDRESS = '192.168.122.244'
PORT = 5050

serverSock = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
serverSock.bind((IP_ADDRESS,PORT))
filename='server1.jpg'
fp = open(filename,'wb+')
ditulis=0
counter=0
while True:
    data, addr = serverSock.recvfrom(1024)
    counter=counter+len(data)
    print(addr," blok ", counter,"panjang : ",len(data), data)
    fp.write(data)

# while True:
#     data,addr = serverSock.recvfrom(1024)
#     print("Message: ", data.decode())

```

single_thread

```

from library import download_gambar, get_url_list, send_file
import time
import datetime

IP_SERVER = "192.168.122.244"
PORT = 5050

def send_all():

```

```

urls = get_url_list()
catat = datetime.datetime.now()
for k in urls:
    download_gambar(urls[k], k)
    print(f"mendownload {urls[k]}")
    waktu_proses = download_gambar(urls[k])
    print(f"completed {waktu_proses} detik")

    send_file(IP_SERVER, PORT, f"{k}.jpg")
    print('send to server succesfully')
selesai = datetime.datetime.now() - catat
print(f"Waktu TOTAL yang dibutuhkan {selesai} detik")

#fungsi download_gambar akan dijalankan secara berurutan

if __name__ == '__main__':
    send_all()

```

4. Output

Server (alpine-1), Client Multi process Async (alpine-2) vs Client multi process (alpine-3)

server.py

< role	alpine-1	×	alpine-2	×	alpine-3	>	—	×
('192.168.122.214', 35133)	blok	5637	panjang :	1	b'\xbe'			
('192.168.122.214', 35133)	blok	5638	panjang :	1	b'U'			
('192.168.122.214', 35133)	blok	5639	panjang :	1	b'\xae'			
('192.168.122.214', 35133)	blok	5640	panjang :	1	b'j'			
('192.168.122.214', 35133)	blok	5641	panjang :	1	b'w'			
('192.168.122.214', 35133)	blok	5642	panjang :	1	b'\x16'			
('192.168.122.214', 35133)	blok	5643	panjang :	1	b'\xc9'			
('192.168.122.214', 35133)	blok	5644	panjang :	1	b'\x1b'			
('192.168.122.214', 35133)	blok	5645	panjang :	1	b')'			
('192.168.122.214', 35133)	blok	5646	panjang :	1	b'\xd9'			
('192.168.122.214', 35133)	blok	5647	panjang :	1	b'\xff'			
('192.168.122.214', 35133)	blok	5648	panjang :	1	b'\x00'			
('192.168.122.214', 35133)	blok	5649	panjang :	1	b'\xb8'			
('192.168.122.214', 35133)	blok	5650	panjang :	1	b'\x7f'			
('192.168.122.214', 35133)	blok	5651	panjang :	1	b'\xa5'			
('192.168.122.214', 35133)	blok	5652	panjang :	1	b'f'			
('192.168.122.214', 35133)	blok	5653	panjang :	1	b'\xbd'			
('192.168.122.214', 35133)	blok	5654	panjang :	1	b'f'			
('192.168.122.214', 35133)	blok	5655	panjang :	1	b'm'			
('192.168.122.214', 35133)	blok	5656	panjang :	1	b'\x0c'			
('192.168.122.214', 35133)	blok	5657	panjang :	1	b'n'			
('192.168.122.214', 35133)	blok	5658	panjang :	1	b'\xcd'			
('192.168.122.214', 35133)	blok	5659	panjang :	1	b'\xe2'			
('192.168.122.214', 35133)	blok	5660	panjang :	1	b'\xc4'			

Client 1 (multiprocess_async.py)


```
< |ole      alpine-1  X      alpine-2  X      alpine-3  X      >  -  X
From https://github.com/josepheric/Pemrograman_Jaringan_E
4339a5f..290d246  Tugas_3    -> origin/Tugas_3
Updating 4339a5f..290d246
Fast-forward
 progjar3/Tugas_3/multi_process.py      | 4 ++--
 progjar3/Tugas_3/multi_process_async.py | 4 ++--
 progjar3/Tugas_3/multi_thread.py       | 4 ++--
 progjar3/Tugas_3/multi_thread_async.py | 4 ++--
 progjar3/Tugas_3/single_thread.py      | 4 ++--
 5 files changed, 10 insertions(+), 10 deletions(-)
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # python3 multi_process_async.py
WARNING:root:image/jpeg
WARNING:root:writing Kompas.jpg dalam waktu 0:00:00.191781 2021-06-17 14:42:59.3
mendownload https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsfbSpec=/0x0:998x
192.168.122.244 5050 Kompas.jpg
send to server sukses
WARNING:root:image/jpeg
WARNING:root:writing cat.jpg dalam waktu 0:00:02.676790 2021-06-17 14:42:59.5292
mendownload https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg
192.168.122.244 5050 cat.jpg
send to server sukses
Waktu TOTAL yang dibutuhkan 0:00:03.030533 detik 2021-06-17 14:42:59.329031 s/d
status TASK
{'Kompas': None, 'cat': None}
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```

Client 2 (multi_process.py)

```
< alpine-1  X      alpine-2  X      alpine-3  X      >  -  X
Downloading chardet-4.0.0-py2.py3-none-any.whl (178 kB)
| 178 kB 3.3 MB/s
Collecting certifi>=2017.4.17
  Downloading certifi-2021.5.30-py2.py3-none-any.whl (145 kB)
| 145 kB 3.6 MB/s
Collecting idna<3,>=2.5
  Downloading idna-2.10-py2.py3-none-any.whl (58 kB)
| 58 kB 4.4 MB/s
Installing collected packages: urllib3, idna, chardet, certifi, requests
Successfully installed certifi-2021.5.30 chardet-4.0.0 idna-2.10 requests-2.25.1
WARNING: You are using pip version 21.0.1; however, version 21.1.2 is available.
You should consider upgrading via the '/usr/bin/python3.8 -m pip install --upgra
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # python3 multi_process.py
WARNING:root:image/jpeg
WARNING:root:writing Kompas.jpg dalam waktu 0:00:00.275700 2021-06-17 14:44:31.5
mendownload https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsfbSpec=/0x0:998x
send to server sukses
192.168.122.244 5050 Kompas.jpg
WARNING:root:image/jpeg
WARNING:root:writing cat.jpg dalam waktu 0:00:01.164374 2021-06-17 14:44:31.8171
mendownload https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg
send to server sukses
192.168.122.244 5050 cat.jpg
Waktu TOTAL yang dibutuhkan 0:00:01.566834 detik 2021-06-17 14:44:31.539135 s/d
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```

Server (alpine-1), Client Multi Thread Async (alpine-2) vs Client Multi Thread (alpine-3)

Server

```

< role alpine-1 X alpine-2 X alpine-3 X > - X
('192.168.122.214', 52367) blok 18033 panjang : 1 b'\xb0'
('192.168.122.214', 52367) blok 18034 panjang : 1 b'\xb3'
('192.168.122.214', 52367) blok 18035 panjang : 1 b'{'
('192.168.122.214', 52367) blok 18036 panjang : 1 b'|'
('192.168.122.214', 52367) blok 18037 panjang : 1 b'$'
('192.168.122.214', 52367) blok 18038 panjang : 1 b'\xce'
('192.168.122.214', 52367) blok 18039 panjang : 1 b'\x0e'
('192.168.122.214', 52367) blok 18040 panjang : 1 b'h'
('192.168.122.214', 52367) blok 18041 panjang : 1 b'<'
('192.168.122.214', 52367) blok 18042 panjang : 1 b'\xf2'
('192.168.122.214', 52367) blok 18043 panjang : 1 b'\x07'
('192.168.122.214', 52367) blok 18044 panjang : 1 b'z'
('192.168.122.214', 52367) blok 18045 panjang : 1 b'\xe0'
('192.168.122.214', 52367) blok 18046 panjang : 1 b'\xb3'
('192.168.122.214', 52367) blok 18047 panjang : 1 b'\x11'
('192.168.122.214', 52367) blok 18048 panjang : 1 b'\xce'
('192.168.122.214', 52367) blok 18049 panjang : 1 b'j'
('192.168.122.214', 52367) blok 18050 panjang : 1 b'\xab'
('192.168.122.214', 52367) blok 18051 panjang : 1 b'l'
('192.168.122.214', 52367) blok 18052 panjang : 1 b'\x0f'
('192.168.122.214', 52367) blok 18053 panjang : 1 b'j'
('192.168.122.214', 52367) blok 18054 panjang : 1 b's'
('192.168.122.214', 52367) blok 18055 panjang : 1 b'8'
('192.168.122.214', 52367) blok 18056 panjang : 1 b'I'

```

Client 1 (multi_thread_async.py)

```

< alpine-1 X alpine-2 X alpine-3 X > - X
192.168.122.244 5050 kompas.jpg
send to server succesful
WARNING:root:image/jpeg
WARNING:root:writing cat.jpg dalam waktu 0:00:02.676790 2021-06-17 14:42:59.5292
mendownload https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg
192.168.122.244 5050 cat.jpg
send to server succesful
Waktu TOTAL yang dibutuhkan 0:00:03.030533 detik 2021-06-17 14:42:59.329031 s/d
status TASK
{'kompas': None, 'cat': None}
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # python3 multi_thread_async.py
WARNING:root:image/jpeg
WARNING:root:writing kompas.jpg dalam waktu 0:00:00.334303 2021-06-17 14:47:03.8
mendownload https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsfbSpec=/0x0:998x
192.168.122.244 5050 kompas.jpg
send file succesfully
WARNING:root:image/jpeg
WARNING:root:writing cat.jpg dalam waktu 0:00:01.059514 2021-06-17 14:47:04.1470
mendownload https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg
send file succesfully
192.168.122.244 5050 cat.jpg
Waktu TOTAL yang dibutuhkan 0:00:01.505727 detik 2021-06-17 14:47:03.812136 s/d
hasil task yang dijalankan
{'kompas': None, 'cat': None}
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #

```

Client 2 (multi_thread.py)

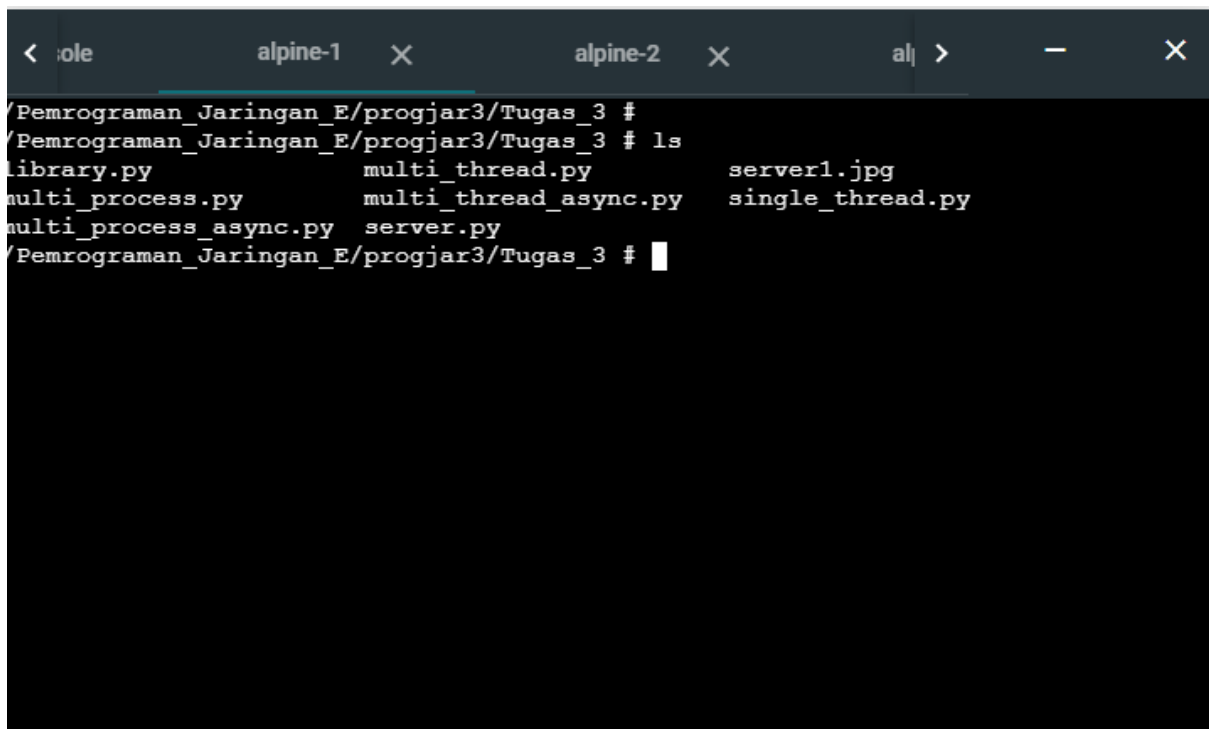
```
alpine-1  X      alpine-2  X      alpine-3  X  >  -  X
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # python3 multi_thread.py
WARNING:root:image/jpeg
WARNING:root:writing kompas.jpg dalam waktu 0:00:00.157945 2021-06-17 14:49:00.7
mendownload https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsfbSpec=/0x0:998x
send file succesfully
192.168.122.244 5050 kompas.jpg
WARNING:root:image/jpeg
WARNING:root:writing cat.jpg dalam waktu 0:00:01.206913 2021-06-17 14:49:00.8963
mendownload https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg
send file succesfully
192.168.122.244 5050 cat.jpg
Waktu TOTAL yang dibutuhkan 0:00:01.495779 detik 2021-06-17 14:49:00.738127 s/d
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```

Singlethread.py:

```
alpine-1  X      alpine-2  X      alpine-3  X  >  -  X
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # python3 single_thread.py
WARNING:root:image/jpeg
WARNING:root:writing kompas.jpg dalam waktu 0:00:00.223541 2021-06-17 14:51:04.2
mendownload https://asset.kompas.com/crops/qz_jJxyaZgGgboomdCEXsfbSpec=/0x0:998x
WARNING:root:image/jpeg
WARNING:root:writing image.jpg dalam waktu 0:00:00.199909 2021-06-17 14:51:04.51
completed 0:00:00.199909 detik
192.168.122.244 5050 kompas.jpg
send to server succesfully
WARNING:root:image/jpeg
WARNING:root:writing cat.jpg dalam waktu 0:00:02.787926 2021-06-17 14:51:05.1963
mendownload https://avatarfiles.alphacoders.com/121/thumb-1920-121594.jpg
WARNING:root:image/jpeg
WARNING:root:writing image.jpg dalam waktu 0:00:00.176159 2021-06-17 14:51:07.9
completed 0:00:00.176159 detik
192.168.122.244 5050 cat.jpg
send to server succesfully
Waktu TOTAL yang dibutuhkan 0:00:03.991070 detik
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```

HASIL FILES:

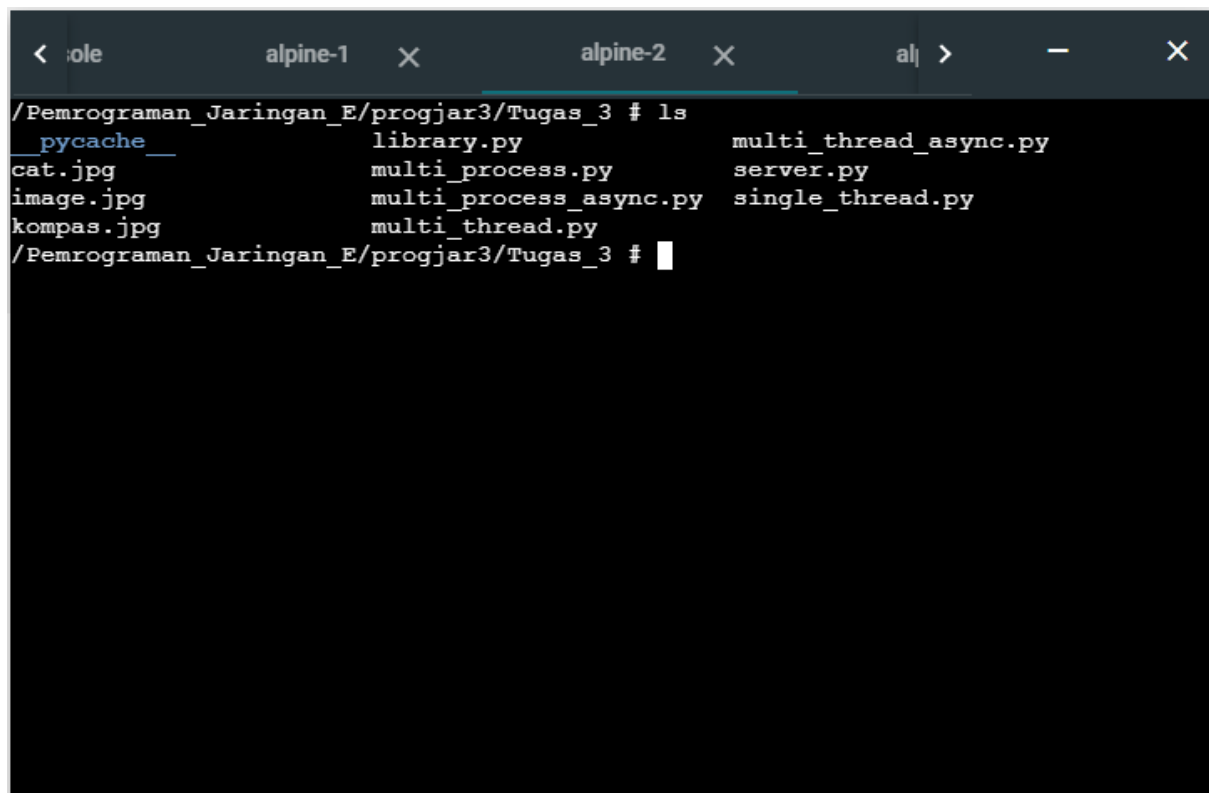
Server:



A terminal window with a dark background and light gray text. The window has a title bar with three tabs: 'role', 'alpine-1', and 'alpine-2'. The 'alpine-1' tab is selected. The terminal shows the following commands and output:

```
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #  
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # ls  
library.py          multi_thread.py      server1.jpg  
multi_process.py    multi_thread_async.py single_thread.py  
multi_process_async.py server.py  
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```

Clients:



A terminal window with a dark background and light gray text. The window has a title bar with three tabs: 'role', 'alpine-1', and 'alpine-2'. The 'alpine-2' tab is selected. The terminal shows the following commands and output:

```
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # ls  
__pycache__         library.py            multi_thread_async.py  
cat.jpg             multi_process.py     server.py  
image.jpg           multi_process_async.py single_thread.py  
kompas.jpg          multi_thread.py  
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```

```
< alpine-1 x alpine-2 x alpine-3 x > - x
/Pemrograman_Jaringan_E/progjar3/Tugas_3 # ls
__pycache__      multi_process.py      server.py
cat.jpg          multi_process_async.py single_thread.py
kompas.jpg       multi_thread.py
library.py       multi_thread_async.py
/Pemrograman_Jaringan_E/progjar3/Tugas_3 #
```