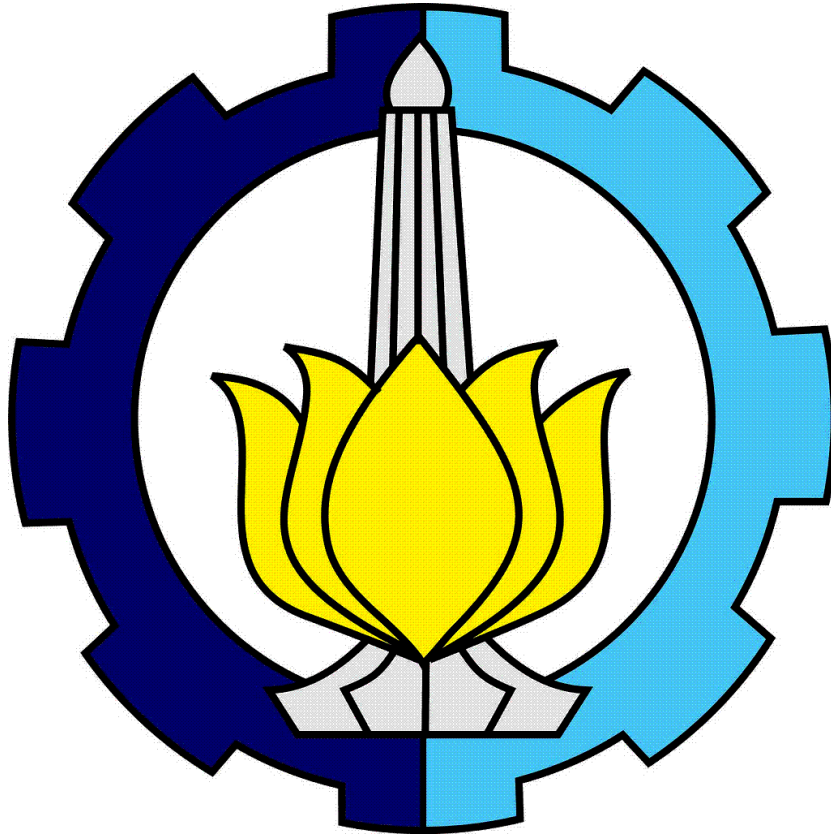


Dokumentasi
Tugas Implementasi Program Server



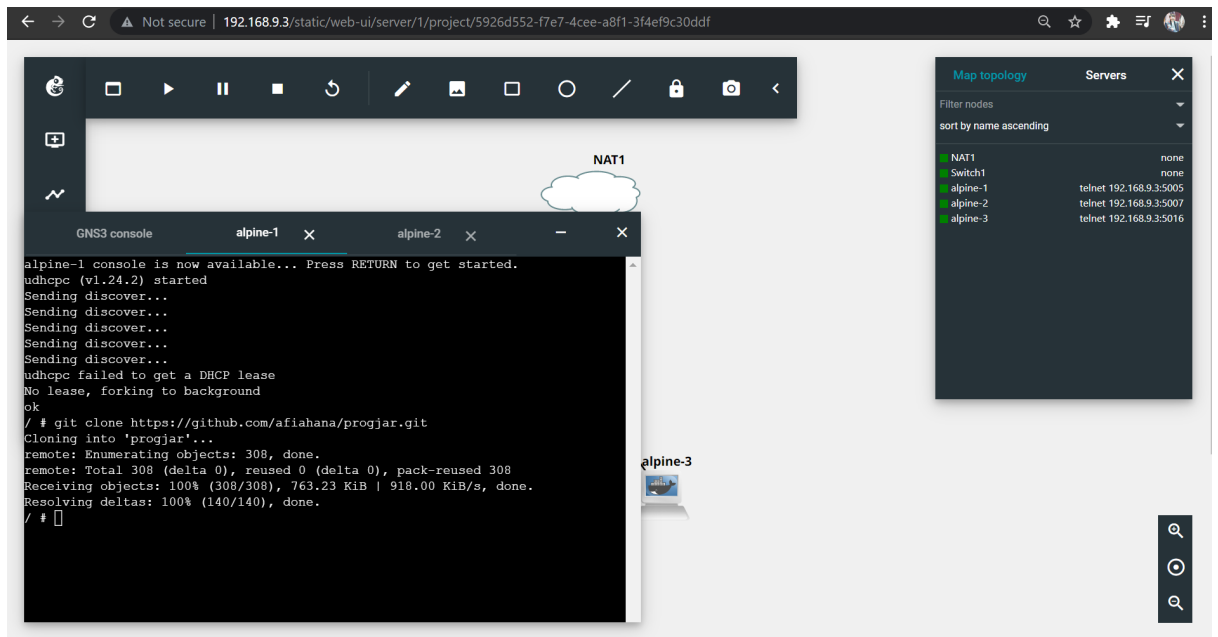
Pemrograman Jaringan
Kelas E

Afia Hana Yusriya

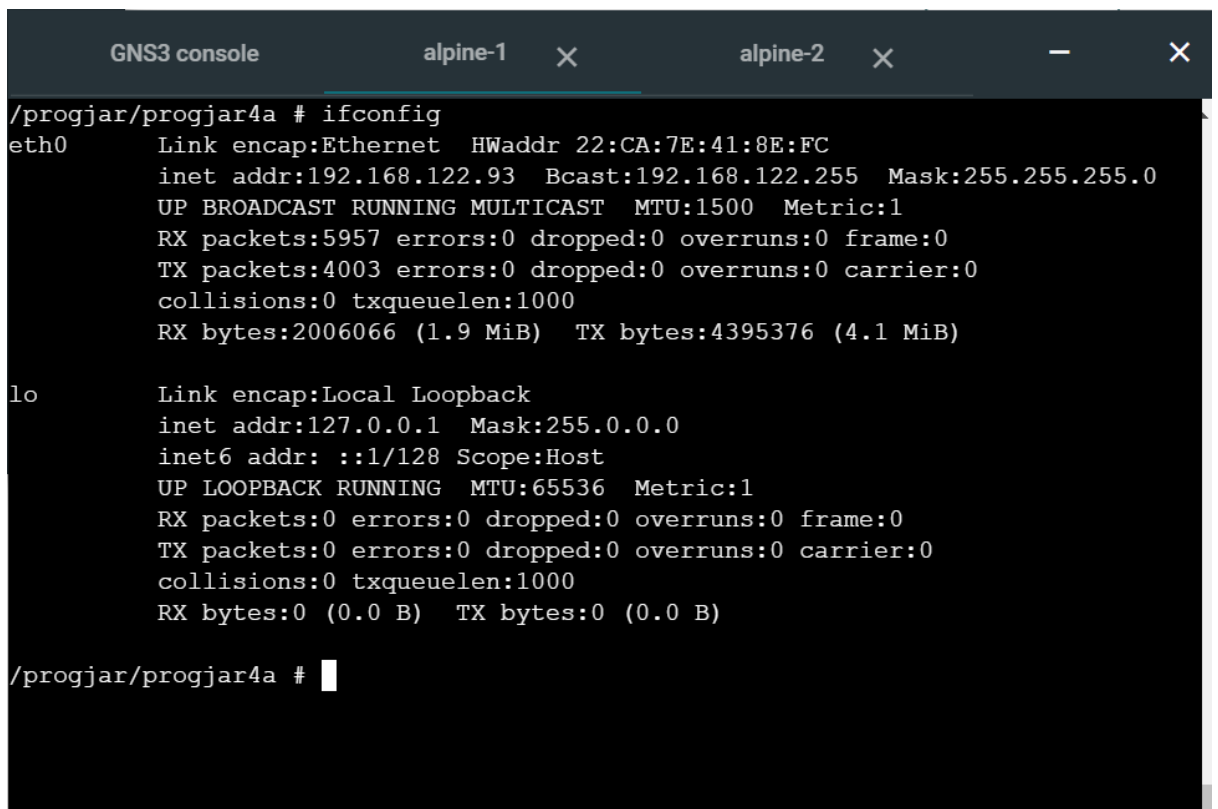
05111840000111

Institut Teknologi Sepuluh Nopember
Surabaya
2021

1. Clone repository progjar



2. Cek IP address alpine-1 menggunakan command **ifconfig**



3. Mengganti IP address pada file_server.py menjadi IP address alpine-1

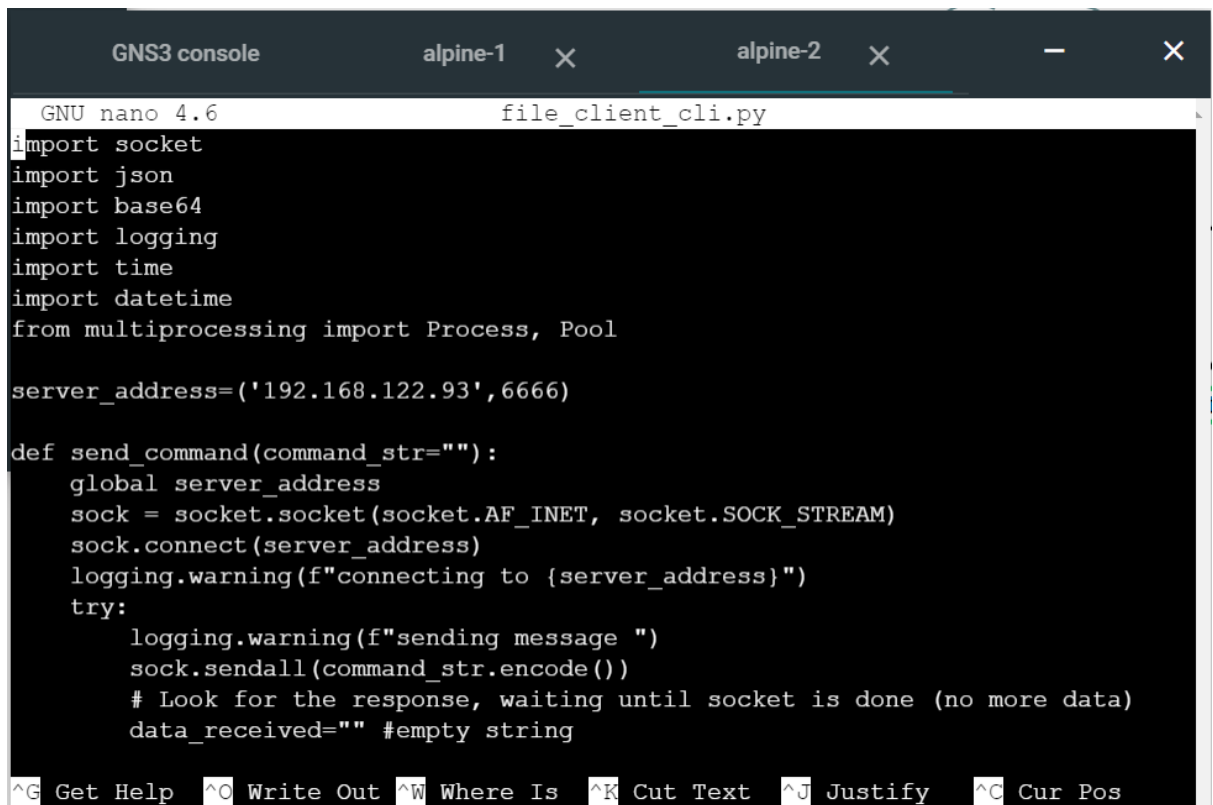
```
GNU nano 4.6 file_server.py
    if data:
        d = data.decode()
        hasil = fp.proses_string(d)
        hasil=hasil+"\r\n\r\n"
        self.connection.sendall(hasil.encode())
    else:
        break
self.connection.close()

class Server(threading.Thread):
    def __init__(self,ipaddress='192.168.122.93',port=6666):
        self.ipinfo=(ipaddress,port)
        self.the_clients = []
        self.my_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        self.my_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
        threading.Thread.__init__(self)

    def run(self):
        logging.warning(f"server berjalan di ip address {self.ipinfo}")

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
```

4. Pada alpine-1, buka folder progjar4a lalu jalankan file file_server.py dengan command **python3 file_server.py**
5. Pada alpine-2, buka folder progjar4a lalu buka file file_client_cli.py
6. Ubah server address pada file_client_cli.py menjadi IP address alpine-1



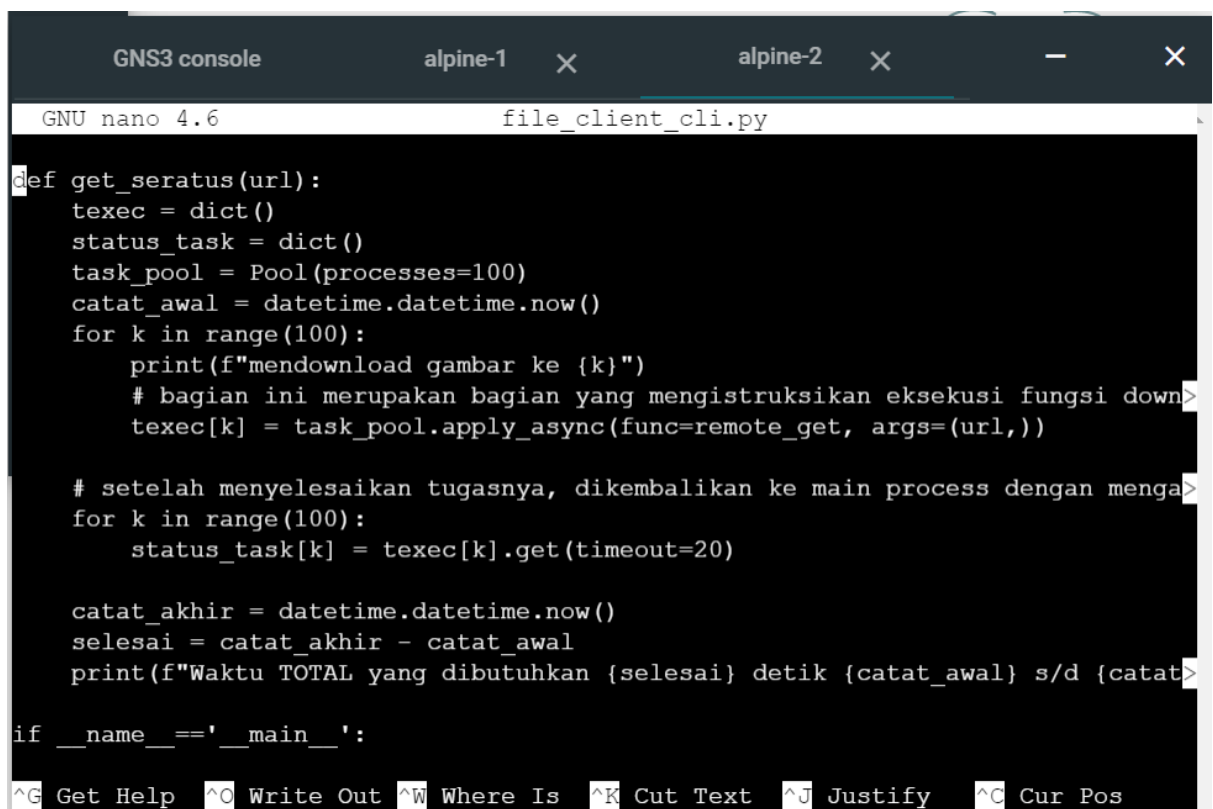
The screenshot shows a GNS3 console window with two tabs: 'alpine-1' and 'alpine-2'. The 'alpine-2' tab is active, displaying a nano editor window for 'file_client_cli.py'. The code in the editor includes imports for socket, json, base64, logging, time, and datetime, and from multiprocessing import Process, Pool. It defines a server_address and a send_command function that connects to the server and sends a message. The nano editor's status bar at the bottom shows shortcuts: ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, and ^C Cur Pos.

```
GNU nano 4.6 file_client_cli.py
import socket
import json
import base64
import logging
import time
import datetime
from multiprocessing import Process, Pool

server_address=('192.168.122.93',6666)

def send_command(command_str=""):
    global server_address
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.connect(server_address)
    logging.warning(f"connecting to {server_address}")
    try:
        logging.warning(f"sending message ")
        sock.sendall(command_str.encode())
        # Look for the response, waiting until socket is done (no more data)
        data_received="" #empty string
```

7. Tambahkan fungsi baru untuk multithreading (beracuan pada multithread udp pada tugas sebelumnya dengan beberapa modifikasi) yaitu



The screenshot shows the same GNS3 console window as before, but the code in the nano editor has been updated. A new function 'get_seratus' is added, which uses a multiprocessing Pool to execute 100 tasks in parallel. Each task calls 'remote_get' with a given URL. The function also calculates the total time taken for all tasks to complete. The nano editor's status bar remains the same.

```
GNU nano 4.6 file_client_cli.py

def get_seratus(url):
    texec = dict()
    status_task = dict()
    task_pool = Pool(processes=100)
    catat_awal = datetime.datetime.now()
    for k in range(100):
        print(f"mendownload gambar ke {k}")
        # bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi down
        texec[k] = task_pool.apply_async(func=remote_get, args=(url,))

    # setelah menyelesaikan tugasnya, dikembalikan ke main process dengan menga
    for k in range(100):
        status_task[k] = texec[k].get(timeout=20)

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

if __name__=='__main__':
```

8. Ubah isi dari fungsi main menjadi seperti gambar berikut

```
GNS3 console      alpine-1  X      alpine-2  X      -      X
GNU nano 4.6      file_client_cli.py
catat_awal = datetime.datetime.now()
for k in range(100):
    print(f"mendownload gambar ke {k}")
    # bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi down
    texec[k] = task_pool.apply_async(func=remote_get, args=(url,))

# setelah menyelesaikan tugasnya, dikembalikan ke main process dengan menga
for k in range(100):
    status_task[k] = texec[k].get(timeout=20)

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

if __name__ == '__main__':
    server_address=('192.168.122.93',6666)
    remote_list()
    #remote_get('pokijan.jpg')
    get_seratus('pokijan.jpg')

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
```

9. Jalankan file_client_cli.py di alpine-2

```
GNS3 console      alpine-1  X      alpine-2  X      -      X
Add a link
mendownload gambar ke 86
mendownload gambar ke 87
mendownload gambar ke 88
mendownload gambar ke 89
mendownload gambar ke 90
mendownload gambar ke 91
mendownload gambar ke 92
mendownload gambar ke 93
mendownload gambar ke 94
mendownload gambar ke 95
WARNING:root:sending message
mendownload gambar ke 96
mendownload gambar ke 97
mendownload gambar ke 98
mendownload gambar ke 99
WARNING:root:connecting to ('192.168.122.93', 6666)
WARNING:root:connecting to ('192.168.122.93', 6666)
WARNING:root:connecting to ('192.168.122.93', 6666)
WARNING:root:connecting to ('192.168.122.93', 6666)
WARNING:root:sending message
WARNING:root:connecting to ('192.168.122.93', 6666)
WARNING:root:sending message
WARNING:root:connecting to ('192.168.122.93', 6666)
```

```
GNS3 console    alpine-1  X    alpine-2  X    -    X
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
WARNING:root:data received from server:
Waktu TOTAL yang dibutuhkan 0:00:04.055139 detik 2021-07-10 14:18:12.859056 s/d
2021-07-10 14:18:16.914195
```

10. Hasil di alpine-1 (server)

```
GNS3 console    alpine-1  X    alpine-2  X    -    X
WARNING:root:connection from ('192.168.122.215', 34782)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.215', 34838)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.215', 34796)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.215', 34814)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.215', 34812)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.215', 34774)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
WARNING:root:connection from ('192.168.122.215', 34734)
WARNING:root:string diproses: GET pokijan.jpg
WARNING:root:memproses request: GET
```

Catatan: Source code lengkap dapat dilihat di <https://github.com/afiahana/progiar>