

## LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Pemograman Jaringan

Kelas : 4IA02

Praktikum ke- : 2

Tanggal : 14/04/2023

Materi : Membuat DNS Menggunakan Python

NPM : 56419075

Nama : Sinatra Kusuma

Ketua Asisten : Muhammad Fikri Maulana

Paraf Asisten :

Nama Asisten :

Jumlah Lembar : 4 Lembar

LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2024

1. DNS\_TABLE adalah tabel domain name dan IP address. Saat server menerima permintaan, ia mencari domain name di tabel. Jika ditemukan, IP address akan dikirim ke client. Jika tidak, pesan "Not Found" dikirim. Socket dibuat dan diikat ke alamat tertentu, kemudian program akan menampilkan pesan. Server akan terus menerima permintaan dan mencari domain name di tabel untuk mengirim IP address ke client.

```
part 2 > server.py > ...
1  import socket
2  DNS_TABLE = {
3      'www.google.com': '192.168.1.1',
4      'www.facebook.com': '192.168.1.2',
5      'www.yahoo.com': '192.168.1.3'
6  }
7
8  server_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
9  server_socket.bind(('localhost', 53))
10 print('DNS server is listening...')
11 while True:
12     data, address = server_socket.recvfrom(1024)
13     domain_name = data.decode()
14     if domain_name in DNS_TABLE:
15         ip_address = DNS_TABLE[domain_name]
16     else:
17         ip_address = 'Not Found'
18     server_socket.sendto(ip_address.encode(), address)
19
```

2. Modul Socket Pada Python Untuk Mengirim Permintaan DNS Lookup Menggunakan Protokol UDP. Tujuannya Adalah Untuk Mendapatkan Informasi Tentang Nama Domain, Seperti Alamat IP Yang Terkait Dengan Domain Tersebut. Penjelasan Setiap Baris Kode Sudah Dijelaskan Secara Detail Pada Rangkuman Tersebut.

```
part 2 > client.py > ...
1  import socket
2  SERVER_IP_ADDRESS = 'localhost'
3  DOMAIN_NAME = 'www.nomu.com'
4  client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
5  client_socket.sendto(DOMAIN_NAME.encode(), (SERVER_IP_ADDRESS, 53))
6  response, _ = client_socket.recvfrom(1024)
7  ip_address = response.decode()
8  print(f"Alamat IP dari {DOMAIN_NAME} adalah {ip_address}")
```

3. Menjalankan server.py untuk memberi respon

```
ASUS@DESKTOP-6VKKARO MINGW64 /d/D '/labti/pjar/part 2
$ python server.py
DNS server is listening...
█
```

4. Menambahkan 3 DNS lain

```
part 2 > server.py > ...
1  import socket
2  DNS_TABLE = {
3      'www.google.com': '192.168.1.1',
4      'www.facebook.com': '192.168.1.2',
5      'www.yahoo.com': '192.168.1.3',
6      'www.twitter.com': '192.168.1.4',
7      'www.youtube.com': '192.168.1.5',
8      'www.apple.com': '192.168.1.6'
9  }
10 }
11 server_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
12 server_socket.bind(('localhost', 53))
13 print('DNS server is listening...')
14 while True:
15     data, address = server_socket.recvfrom(1024)
16     domain_name = data.decode()
17     if domain_name in DNS_TABLE:
18         ip_address = DNS_TABLE[domain_name]
19     else:
20         ip_address = 'Not Found'
21     server_socket.sendto(ip_address.encode(), address)
22
```

5. Menjalankan client.py mendapatkan respon dari server.py

```
part 2 > client.py > ...
1 import socket
2 SERVER_IP_ADDRESS = 'localhost'
3 DOMAIN_NAME = 'www.youtube.com'
4 client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
5 client_socket.sendto(DOMAIN_NAME.encode(), (SERVER_IP_ADDRESS, 53))
6 response, _ = client_socket.recvfrom(1024)
7 ip_address = response.decode()
8 print(f"Alamat IP dari {DOMAIN_NAME} adalah {ip_address}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
ASUS@DESKTOP-6VKKARO MINGW64 /d/D '/labti/pjar/part 2
$ python server.py
DNS server is listening...
█
```

```
ASUS@DESKTOP-6VKKARO MINGW64 /d/D '/labti/pjar/part 2
$ python client.py
Alamat IP dari www.youtube.com adalah 192.168.1.5
ASUS@DESKTOP-6VKKARO MINGW64 /d/D '/labti/pjar/part 2
$ █
```