EX NO:10b   Customized ping command to test the server connectivity using socket
Date:22.9.24

Aim:

To develop a ping program to test server connectivity using socket.

Algorithm: Servers.py

1. Import the socket package

2. Initialize local IP address and local port

3. Create a socket

4. Bind the IP address and port number

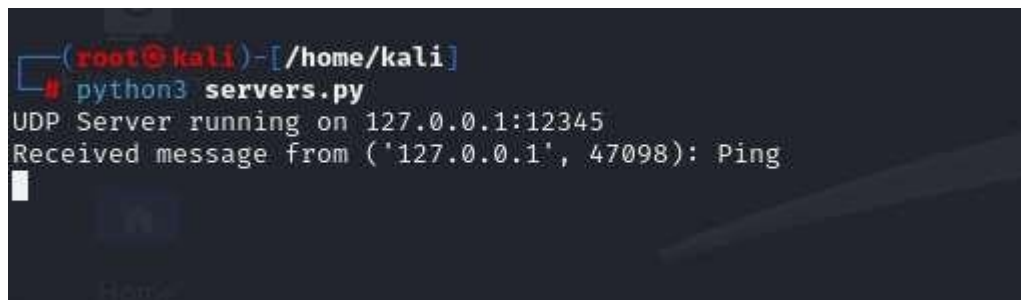5. Receive client message and send reply

Clients.py

1. Import the socket package

2. Initialize server IP address and port

3. Create a socket

4. Start the timer and send message

5. Receive reply and stop timer

Server Program:

import socket def start_server(host='127.0.0.1', port=12345):   with

socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:

    s.bind((host, port))

    print(f"UDP Server running on {host}:{port}")

```python
    while True:
        data, addr = s.recvfrom(1024)
        print(f"Received message from {addr}: {data.decode()}")          s.sendto(b'Pong',
addr)


if __name__ == "__main__":
    start_server()
```
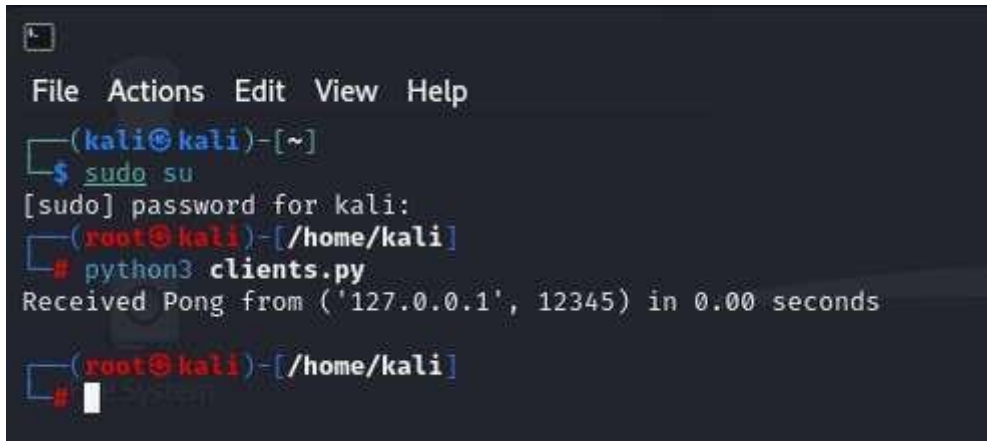
Output:

Client Program: import socket import time

```python
def ping_server(host='127.0.0.1', port=12345):
    with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as s:
try:
        s.settimeout(2)
start = time.time()
        s.sendto(b'Ping', (host, port))
data, addr = s.recvfrom(1024)          end
= time.time()
        print(f"Received {data.decode()} from {addr} in {end - start:.2f} seconds")
except socket.timeout:
        print("Request timed out")
```

```
if __name__ == "__main__":
    ping_server()
```

Output:

Result:

      Ping program to test server connectivity using socket is verified.