Exp: 13
Date:

Ping Program

Aim:

To implement your own ping program.

Algorithm:

UDP server:

→ Create a UDP socket & bind it to a specific address & port.

→ Wait for message.

→ print message & client address.

→ send back ping to client.

UDP client:

→ Create a UDP socket & set a 2 sec timeout.

→ Send 'ping' to server.

→ If a response ping is received print response & calculate RTT.

→ If no response within 2 sec print timeout.

Code:

Server.py

```
import socket
def start_server(host='127.0.0.1',
                 port=12345):
    with socket.socket(socket.AF_INET, so
    s.bind((host, port))
    print(f"UDP server running on
          {host}:{port}")
    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()
```

client.py:

```
import time
import socket
def ping_server(host="127.0.0.1",
                port=12345)
```

```
with socket.socket (socket.AFINET,
                    socket.sock) as s:

try:
    s.settime(2)
    start = time.time()
    s.sendto (b'ping', (host, port))
    data, addr = s.recvfrom (1024)
    end = time.time()
    print(f"recieved {data.decode()}
          from {addr} in {end-start}
          :.2f} seconds")

except socket.timeout:
    print ("request timed out")

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

Output

Terminal sender

> python server.py

UDP server running on 127.0.0.1 : 12345

Recieved message from ('127.0.0.1', 50051 : ping)

Terminal receiver

>python client.py

Recieved ping from ('127.0.0.1', 12345) in 0.00 seconds.

Result:

Thus a ping program has been successfully executed.