

September 20, 2024

UDP Pinger Lab

Mehraz Abedin Raz 2022293
Nakul Garg 2022309

Objective & System Overview

Objective

- To implement a UDP-based ping client-server system.
- Simulate packet loss and calculate key network performance metrics.
- Implement a UDP heartbeat mechanism to monitor server availability.

System Architecture:

- Client-Server Model
- UDP protocol is used for communication (connectionless and unreliable).
- The client sends pings, and the server responds with the message echoed back.

UDP Protocol and Key Features

Why UDP?

- Connectionless: No need for connection establishment.
- Lightweight: No overhead for ensuring reliable transmission.
- Suitable for real-time applications like pings or heartbeat monitoring.

Key Features:

- Simple packet structure.
- Unreliable, meaning packet delivery is not guaranteed.
- Low latency, as it doesn't wait for acknowledgment like TCP.

Client Implementation

- Client Code (client.py)
 - Sends 10 pings to the server.
 - Each ping includes a sequence number and timestamp.
 - Waits for a response (timeout set to 1 second).
 - Calculates RTT (Round Trip Time) for each received ping.
 - Logs RTT or prints “Request timed out” if the packet is lost.
- Key Functions
 - `socket.sendto()`: Used to send a message to the server.
 - `socket.recvfrom()`: Used to receive a response from the server.
 - RTT calculation: The difference between the time a message was sent and when the response was received.

Server Implementation

Server Code (server.py):

- Listens for UDP packets on a specific port.
- Processes received messages and randomly drops 30% of packets to simulate packet loss.
- If a packet is not dropped, it echoes the message back to the client.

Packet Loss Simulation:

- A random integer between 0 and 10 is generated for each packet.
- If the integer is less than 4, the packet is dropped (30% chance).

Key Functions:

- `socket.bind()`: Binds the server to a port to listen for incoming messages.
- `socket.recvfrom()`: Receives messages from the client.
- `socket.sendto()`: Sends the echoed message back to the client.

UDP Heartbeat Mechanism

Heartbeat Client (UDPheartbeatclient):

- Sends pings continuously to the server.
- Tracks the number of consecutive missed pings.
- Stops if three consecutive pings are missed, assuming the server is down.

Server Monitoring:

- Useful for monitoring whether a server is up and running.
- If the server fails to respond for a set period, the client assumes the server is no longer available.

Example Output:

- Displays pings sent and received.
- Terminates after three consecutive timeouts.

Conclusion

Summary

- The UDP Pinger lab demonstrated how UDP can be used to simulate network conditions with packet loss and timeouts.
- We successfully calculated key network metrics like RTT and packet loss rate.
- The UDP heartbeat implementation provides a useful tool for server availability monitoring.

Challenges

- Some RTT values were recorded as 0.00000000 ms, indicating potential timing issues or extremely quick responses in a local environment.

Screenshots & Results

- Client output (10 pings)

MinRTT: The shortest time it takes for a packet to travel from the client to the server and back.

MaxRTT: The longest time it takes for a packet to travel from the client to the server and back.

AvgRTT: The average time it takes for packets to make the round trip, calculated by adding up the RTT for all pings and dividing by the number of successful pings.

```
Response:- Ping no.:1 is recieved in 0.0060 sec
RTT: 6.04343414 ms
Request timed out
Response:- Ping no.:3 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Response:- Ping no.:4 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Response:- Ping no.:5 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Response:- Ping no.:6 is recieved in 0.0010 sec
RTT: 1.00922585 ms
Response:- Ping no.:7 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Response:- Ping no.:8 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Request timed out
Response:- Ping no.:10 is recieved in 0.0000 sec
RTT: 0.00000000 ms
MinRTT: 0.00000000 ms
MaxRTT: 6.04343414 ms
AvgRTT: 0.88158250 ms
packet loss: 20.00000000 %
```


Screenshots & Results

- UDP Heartbeat output (udp packet sent until 3 consecutive timeouts)

RTT: RTT is the time difference between when the packet was sent and when the response is received.

```
Response:- Ping no.:1 is recieved in 0.0061 sec  
RTT: 6.13522530 ms  
Request timed out  
Response:- Ping no.:3 is recieved in 0.0000 sec  
RTT: 0.00000000 ms  
Response:- Ping no.:4 is recieved in 0.0000 sec  
RTT: 0.00000000 ms  
Request timed out  
Request timed out  
Response:- Ping no.:7 is recieved in 0.0000 sec  
RTT: 0.00000000 ms  
Request timed out  
Response:- Ping no.:9 is recieved in 0.0000 sec  
RTT: 1.01971626 ms  
Response:- Ping no.:10 is recieved in 0.0010 sec  
RTT: 1.01065636 ms  
Request timed out  
Request timed out  
Request timed out  
SERVER IS CLOSED
```

Screenshots & Results

- UDP Heartbeat output (udp packet sent until 3 consecutive timeouts)

In this screenshot, server reached ping response till 46.

```
Response:- Ping no.:34 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Response:- Ping no.:35 is recieved in 0.0011 sec
RTT: 1.28173828 ms
Request timed out
Response:- Ping no.:37 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Request timed out
Request timed out
Response:- Ping no.:40 is recieved in 0.0008 sec
RTT: 0.80633163 ms
Response:- Ping no.:41 is recieved in 0.0010 sec
RTT: 0.98609924 ms
Response:- Ping no.:42 is recieved in 0.0010 sec
RTT: 1.04832649 ms
Request timed out
Request timed out
Response:- Ping no.:45 is recieved in 0.0010 sec
RTT: 1.04832649 ms
Response:- Ping no.:46 is recieved in 0.0000 sec
RTT: 0.00000000 ms
Request timed out
Request timed out
Request timed out
SERVER IS CLOSED
```

September 20, 2024

Thank you

Mehraz Abedin Raz 2022293
Nakul Garg 2022309