

Redes 2
Cliente/Servidor UDP/IP

Alex Matsuo - GRR20196993
Lucas Soni Teixeira - GRR20190395

Introdução

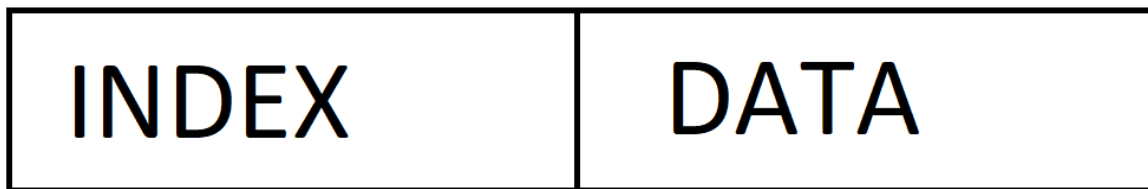
Neste trabalho foi implementado um servidor de streams em Python. Para fazer a comunicação foi utilizado um sistema de cliente/servidor usando sockets UDP. Vários clientes podem se conectar ao servidor e receber as mensagens sendo transmitidas. O servidor irá transmitir continuamente mensagens para todos os clientes conectados.

A Mensagem

A formatação da mensagem tem dois campos: um index que irá identificar a sequência da mensagem e o dado em si. O index simplesmente irá aumentar a cada mensagem enviada enquanto o dado irá somar 2.

Para isso foi criada uma classe Packet que foi instanciada e através da biblioteca pickle foi transformada em bits para que a comunicação entre cliente e servidor fosse possível.

Mensagem



Servidor

Para que o servidor possa funcionar foi utilizado socket UDP, em seguida foi criado um pacote inicial que será o objeto base para criar as mensagens que serão enviadas. O usuário também pode configurar um delay ao executar o servidor colocando no primeiro argumento o valor do delay em segundos. A funcionalidade principal será enviar continuamente pacotes para todos os clientes conectados e gerenciá-los.

Ex para delay : `python3 server.py 2`

Clientes Múltiplos

Para lidar com que o servidor possa ser conectado por múltiplos clientes foi utilizado uma lista de clientes que é atualizada a cada mensagem.

Cliente

O cliente irá tentar se conectar ao servidor enviando uma mensagem. A porta utilizada irá ser atualizada dinamicamente caso ela esteja sendo utilizada.

Depois o cliente irá escutar o servidor e mostrar todas as mensagens sendo recebidas imprimindo seu devido índice e dado.

Até que o cliente seja interrompido o cliente vai somando os dados e no final imprimir a soma de todos os dados recebidos.

Estatísticas

Ao final da execução do cliente foi exibido estatísticas com quantidade de pacotes recebidos, perdidos ou fora de ordem.

Logs

O programa gera 2 arquivos logs um para o cliente e outro para o servidor descrevendo o que aconteceu na comunicação.

Server log - exemplo

UDP server up and streaming

Client - 127.0.0.1:8889, Index - 0, Data - 2
Client - 127.0.0.1:8889, Index - 1, Data - 4
Client - 127.0.0.1:8889, Index - 2, Data - 6
Client - 127.0.0.1:8889, Index - 3, Data - 8
Client - 127.0.0.1:8889, Index - 4, Data - 10
Client - 127.0.0.1:8889, Index - 5, Data - 12
Client - 127.0.0.1:8889, Index - 6, Data - 14
Client - 127.0.0.1:8890, Index - 6, Data - 14
Client - 127.0.0.1:8889, Index - 7, Data - 16
Client - 127.0.0.1:8890, Index - 7, Data - 16
Client - 127.0.0.1:8889, Index - 8, Data - 18
Client - 127.0.0.1:8890, Index - 8, Data - 18
Client - 127.0.0.1:8889, Index - 9, Data - 20
Client - 127.0.0.1:8890, Index - 9, Data - 20
Client - 127.0.0.1:8889, Index - 10, Data - 22
Client - 127.0.0.1:8890, Index - 10, Data - 22
Client - 127.0.0.1:8889, Index - 11, Data - 24

Client - 127.0.0.1:8890, Index - 11, Data - 24
Client - 127.0.0.1:8891, Index - 11, Data - 24
Client - 127.0.0.1:8889, Index - 12, Data - 26
Client - 127.0.0.1:8890, Index - 12, Data - 26
Client - 127.0.0.1:8891, Index - 12, Data - 26
Client - 127.0.0.1:8889, Index - 13, Data - 28
Client - 127.0.0.1:8890, Index - 13, Data - 28
Client - 127.0.0.1:8891, Index - 13, Data - 28
Client - 127.0.0.1:8889, Index - 14, Data - 30
Client - 127.0.0.1:8890, Index - 14, Data - 30
Client - 127.0.0.1:8891, Index - 14, Data - 30
Client - 127.0.0.1:8889, Index - 15, Data - 32
Client - 127.0.0.1:8890, Index - 15, Data - 32
Client - 127.0.0.1:8891, Index - 15, Data - 32
Client - 127.0.0.1:8889, Index - 16, Data - 34
Client - 127.0.0.1:8890, Index - 16, Data - 34
Client - 127.0.0.1:8891, Index - 16, Data - 34
Client - 127.0.0.1:8889, Index - 17, Data - 36
Client - 127.0.0.1:8890, Index - 17, Data - 36
Client - 127.0.0.1:8891, Index - 17, Data - 36
Client - 127.0.0.1:8889, Index - 18, Data - 38
Client - 127.0.0.1:8890, Index - 18, Data - 38
Client - 127.0.0.1:8891, Index - 18, Data - 38
Server stopped.

Cliente log - exemplo

UDP client listening on port: 8889

Received: Index - 0, Data - 2
Received: Index - 1, Data - 4
Received: Index - 2, Data - 6
Received: Index - 3, Data - 8
Received: Index - 4, Data - 10
Received: Index - 5, Data - 12
UDP client listening on port: 8890

Received: Index - 6, Data - 14
Received: Index - 6, Data - 14

Received: Index - 7, Data - 16
Received: Index - 7, Data - 16
Received: Index - 8, Data - 18
Received: Index - 8, Data - 18
Received: Index - 9, Data - 20
Received: Index - 9, Data - 20
Received: Index - 10, Data - 22
Received: Index - 10, Data - 22
UDP client listening on port: 8891

Received: Index - 11, Data - 24
Received: Index - 11, Data - 24
Received: Index - 11, Data - 24
Received: Index - 12, Data - 26
Received: Index - 12, Data - 26
Received: Index - 12, Data - 26
Received: Index - 13, Data - 28
Received: Index - 13, Data - 28
Received: Index - 13, Data - 28
Received: Index - 14, Data - 30
Received: Index - 14, Data - 30
Received: Index - 14, Data - 30
Received: Index - 15, Data - 32
Received: Index - 15, Data - 32
Received: Index - 15, Data - 32
Received: Index - 16, Data - 34
Received: Index - 16, Data - 34
Received: Index - 16, Data - 34
Received: Index - 17, Data - 36
Received: Index - 17, Data - 36
Received: Index - 17, Data - 36
Total sum of received data: 186
Total packets received: 7
Total packets lost: 11
Packets out of order: 0
Total sum of received data: 286
Total packets received: 12
Total packets lost: 6
Packets out of order: 0
Received: Index - 18, Data - 38

Total sum of received data: 380

Total packets received: 19

Total packets lost: 0

Packets out of order: 0