



Projeto de Aplicação

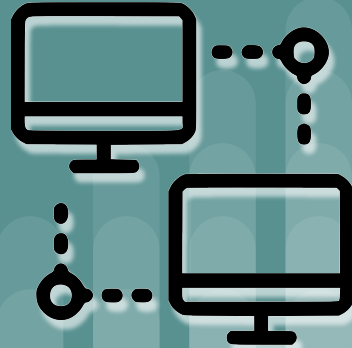
Redes II

Desenvolvimento de uma aplicação de
comunicação bilateral utilizando WebSockets

github.com/pksasso/redesIITrab

Alunos:

Arthur Abreu
Brendo Costa
Patrick Sasso



Objetivo

Desenvolvimento de uma aplicação de comunicação bilateral entre diferentes clientes utilizando o auxílio de um servidor.

A implementação foi realizada através de uma conexão que utiliza a tecnologia **WebSockets**, que possibilita abrir uma sessão de comunicação bilateral entre dois clientes com o intermédio de um servidor para a troca de mensagens assíncronas.





O que é WebSocket?

WebSocket é uma tecnologia relativamente nova que utiliza uma combinação do protocolo **RFC 6455** padronizado na IETF de 2011, e da API JavaScript para possibilitar comunicação entre dispositivos. Através dele, podemos criar um canal de **comunicação bidirecional** (full-duplex), de baixa latência que envia mensagens a um servidor e recebe respostas orientadas a eventos entre dois hosts através de uma conexão TCP/IP.



WebSocket



JavaScript





Comparação: WebSocket e HTTP

O **WebSocket** é um protocolo da camada de aplicação, assim como o **HTTP**. Todavia, o protocolo utiliza o HTTP durante o estabelecimento de uma conexão — *handshaking* — entre os hosts. No final do handshaking, um cabeçalho HTTP Upgrade sinaliza que toda a comunicação se dará exclusivamente através do protocolo WebSocket a partir daquele momento, até que um dos hosts decida encerrar a comunicação.

O WebSocket é uma alternativa à técnica de *polling* HTTP. Enquanto o HTTP é half-duplex (cliente solicita, servidor responde), o **WebSocket é full-duplex**, permitindo que ambos troquem mensagens simultaneamente. É também uma alternativa a padrões fechados de comunicação, como o Adobe Flash Player (que implementa sockets TCP/IP).



Handshake - Requisição do Cliente

GET /chat HTTP/1.1

Host: example.com:8000

Upgrade: websocket

Connection: Upgrade

Sec-WebSocket-Key: dGhlIHNhbXBsZSBub25jZQ==

Sec-WebSocket-Version: 13



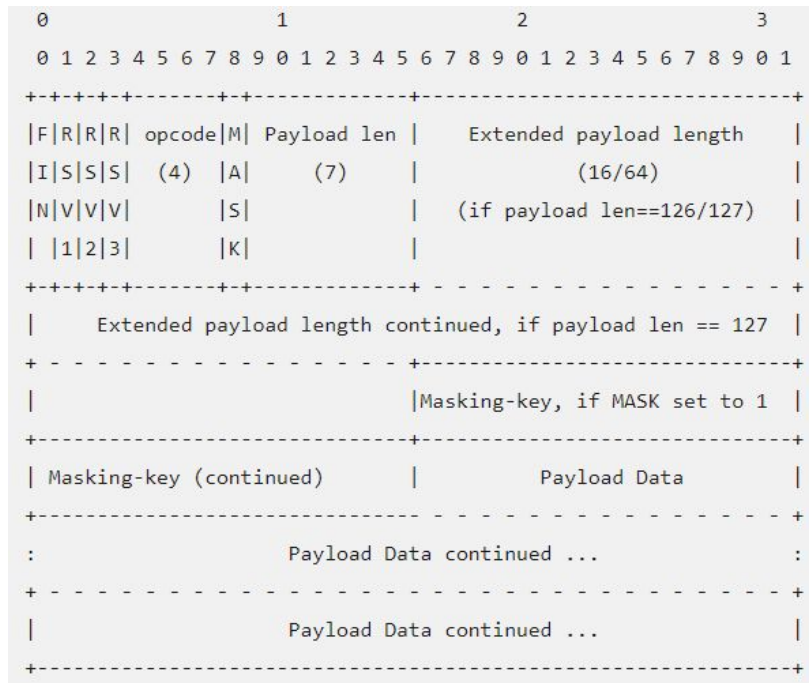
Handshake - Resposta do Servidor

HTTP/1.1 101 Switching Protocols

Upgrade: websocket

Connection: Upgrade

Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+x0o=





Implementações

O WebSocket pode ser implementado em várias linguagens diferentes. Uma lista de implementações do protocolo para diferentes linguagens e frameworks pode ser acessada em:

github.com/facundofarias/awesome-websockets

JavaScript: Socket.IO

PHP: Ratchet

Python: websockets

Go: Gorilla WebSocket

Entre outras...



socket.io



Ratchet



Gorilla



Onde usar?

O protocolo WebSocket é capaz de estabelecer conexões “soquete” para a troca de mensagens, por isso é muito utilizada para a comutação de dados entre hosts. É um protocolo que estabelece um fluxo de dados bilateral de forma muito rápida e eficiente, assim, pode ser utilizado para qualquer aplicação que precise trocar dados de forma rápida e eficiente. Entre as possíveis aplicações, podemos destacar:

- Aplicações multimídia
- Aplicações em tempo real
- Jogos multiplayer
- Telemetria/sondagem de dados
- Dispositivos IoT



Exemplo prático

Nossa Aplicação:



Chat de redes

Celular

1

Entrar na sala

192.168.100.39:8000

Chat ao vivo
ID: 1

Celular | 16:10

Olá mundo!

Digite algo aqui

Enviar



websocket

No.	Time	Source	Destination	Protocol	Length	Info
2597	25.439246800	192.168.100.36	192.168.100.39	WebSo...	78	WebSocket Text [FIN] [MASKED]
2599	25.439522489	192.168.100.39	192.168.100.36	WebSo...	74	WebSocket Text [FIN]
2605	25.549151512	192.168.100.36	192.168.100.39	WebSo...	73	WebSocket Text [FIN] [MASKED]
3729	39.385678175	192.168.100.36	192.168.100.39	WebSo...	91	WebSocket Text [FIN] [MASKED]
3749	39.391947866	192.168.100.36	192.168.100.39	WebSo...	91	[TCP Spurious Retransmission] WebSocket Text [FIN] [MASKED]
4295	46.094943178	192.168.100.36	192.168.100.39	WebSo...	161	WebSocket Text [FIN] [MASKED]
5090	50.437944304	192.168.100.39	192.168.100.36	WebSo...	69	WebSocket Text [FIN]
5102	50.495644515	192.168.100.36	192.168.100.39	WebSo...	73	WebSocket Text [FIN] [MASKED]

```

Frame 4295: 161 bytes on wire (1288 bits), 161 bytes captured (1288 bits) on interface wlxdc4ef40e3ff1, id 0
  Ethernet II, Src: e2:0c:7f:c4:0a:e7 (e2:0c:7f:c4:0a:e7), Dst: Shenzhen_0e:3f:f1 (dc:4e:f4:0e:3f:f1)
  Internet Protocol Version 4, Src: 192.168.100.36, Dst: 192.168.100.39
  Transmission Control Protocol, Src Port: 58760, Dst Port: 3001, Seq: 688, Ack: 138, Len: 95
  WebSocket
  Line-based text data (1 lines)
    42["send_message",{"room":"1","author":"Celular","message":"Olá mundo!","time":"16:10"}]

```

```

0000  dc 4e f4 0e 3f f1 e2 0c 7f c4 0a e7 08 00 45 00  .N..?.. ....E.
0010  00 93 08 c2 40 00 40 06 e8 06 c0 a8 64 24 c0 a8  .@.@. ....d$.
0020  64 27 e5 88 0b b9 48 03 af 98 27 bd c3 f7 80 18  d'....H. ....
0030  01 05 3a f8 00 00 01 01 08 0a 64 76 5d 32 2d 67  .:.....-dv]2-g
0040  64 a7 81 d9 a0 94 33 cb 94 a6 68 e9 d3 f1 5d af  d.....3. .h...].
0050  ff f9 56 b8 d3 f5 54 ae 82 b8 48 e9 d2 fb 5c a6  .V...T. .H...\.
0060  82 ae 11 fa 82 b8 11 aa d5 e0 5b a4 d2 b6 09 e9  .:.....[.....
0070  e3 f1 5f be cc f5 41 e9 8c b6 5e ae d3 e7 52 ac  ._-...A. .^...R.
0080  c5 b6 09 e9 ef f8 f0 6a 80 f9 46 a5 c4 fb 12 e9  .:.....] .F.....
0090  8c b6 47 a2 cd f1 11 f1 82 a5 05 f1 91 a4 11 b6  .G.....
00a0  fd

```



Wireshark



Nossa Aplicação:

Chat de redes

Computador

1

Entrar na sala

Chat ao vivo
ID: 1

Celular | 16:10

Olá mundo!

Computador | 16:11

Olá! Tudo bem!?

Celular | 16:19

Estou ótimo!!

Digite algo aqui

Enviar



Referências:

WebSockets for fun and profit - StackOverflow

<https://stackoverflow.blog/2019/12/18/websockets-for-fun-and-profit/>

Writing WebSocket client applications - mdn web docs

https://developer.mozilla.org/pt-BR/docs/Web/API/WebSockets_API/Writing_WebSocket_client_applications

10 Most Amazing Use Cases of WebSockets — Go Real-time!! - Vikas Sood

<https://medium.com/the-developer-journal/10-most-amazing-use-cases-of-websockets-go-real-time-166b71e0e711>

Documentation - Socket.IO

<https://socket.io/docs/v4/>

Writing WebSocket servers

https://developer.mozilla.org/pt-BR/docs/Web/API/WebSockets_API/Writing_WebSocket_servers

Obrigado!