

DO YOU  
REALLY NEED  
WEBSOCKETS



# Stanko Krtalic Rusendic

 [github.com/monorkin](https://github.com/monorkin)

 [@monorkin](https://twitter.com/monorkin)

 [@monorkin@mastodon.social](https://masto.host/monorkin)

 [hey@stanko.io](mailto:hey@stanko.io)

 [stanko.io](https://stanko.io)

**very  
big  
things.**

a digital products agency.

stanko.krtalic.rusendic@verybigthings.com

# 01 WHY WEBSOCKETS

Challenge – Response











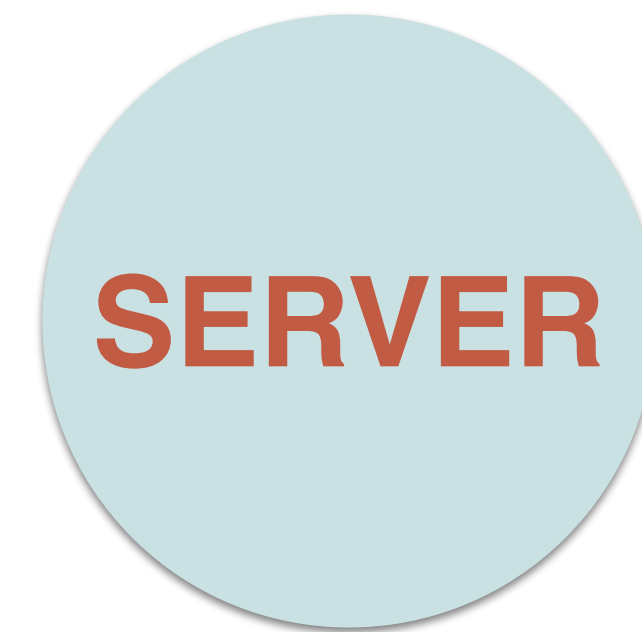
HTTP is Half-duplex

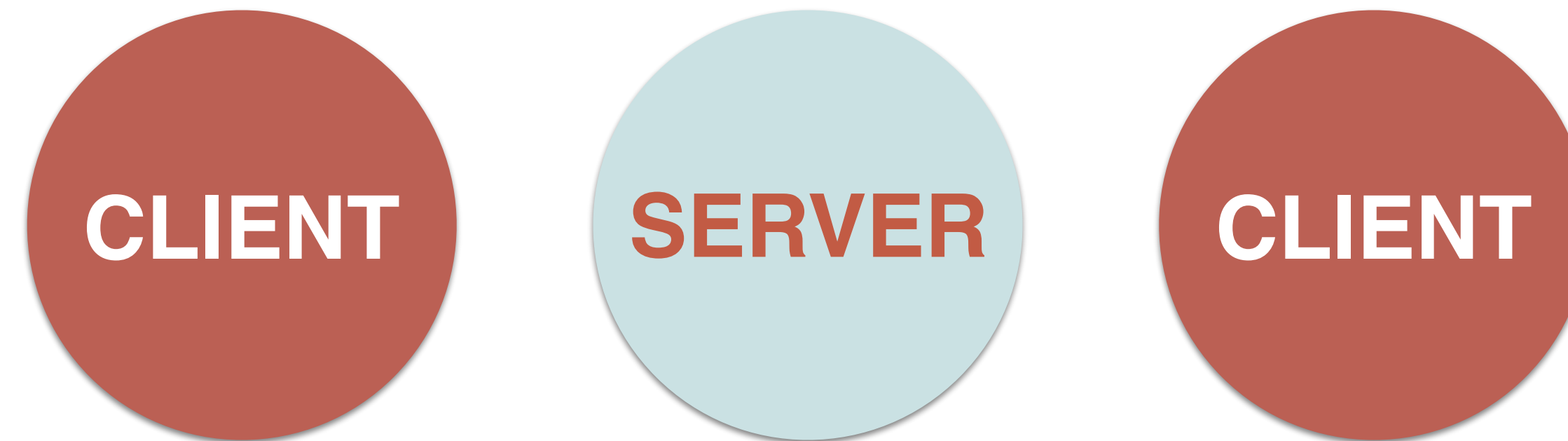
The problem?

A screenshot of a terminal window displaying IRC chat logs. The window has a dark background with a sidebar on the left containing various icons. The main area shows a series of messages, each starting with a timestamp [18:09] and a status indicator ==. The messages are a mix of text and actions. The text messages include a status message, a list of freenode policies and guidelines, and a list of sponsors. The actions include a user connecting and logging in. The messages are as follows:  
[18:09] == - developers and users. freenode live seeks to raise awareness of and  
[18:09] == - promote FOSS alternatives to proprietary software.  
[18:09] == -  
[18:09] == - Facilitating face-to-face interaction, creative workshops, talks and think  
[18:09] == - tanks, freenode #live will bring developers and users together in a nurturing  
[18:09] == - and dynamic environment stimulating the free exchange of ideas and  
[18:09] == - information while fostering cross-project collaboration and dialogue for  
[18:09] == - innovation.  
[18:09] == -  
[18:09] == - Please join us in the #live channel on the freenode network, or send us an  
[18:09] == - e-mail (team@freenode.live) if you have any questions and would like to  
[18:09] == - get involved! Further information will also be posted on the freenode.net  
[18:09] == - website.  
[18:09] == -  
[18:09] == - See <https://freenode.live/> for more information and to get your tickets!  
[18:09] == -  
[18:09] == - We hope to see you there!  
[18:09] == -  
[18:09] == -  
[18:09] == - By connecting to freenode you indicate that you have read and  
[18:09] == - accept our policies and guidelines as set out on <https://freenode.net>  
[18:09] == -  
[18:09] == - In the event that you observe behaviour that contravenes our policies,  
[18:09] == - please notify a volunteer staff member via private message, or send us an  
[18:09] == - e-mail to complaints@freenode.net -- we will do our best to address the  
[18:09] == - situation within a reasonable period of time, and we may request further  
[18:09] == - information or, as appropriate, involve other parties such as channel operators  
[18:09] == - Group Contacts representing an on-topic group.  
[18:09] == -  
[18:09] == - freenode runs an open proxy scanner.  
[18:09] == -  
[18:09] == - If you are looking for assistance, you will be able to find a list of  
[18:09] == - volunteer staff using the '/who freenode/staff/\*' command, and you may  
[18:09] == - message any of us at any time. Please note that freenode predominantly  
[18:09] == - provides assistance via private message, and while we have a network  
[18:09] == - channel the primary venue for support requests is via private message to  
[18:09] == - a member of the volunteer staff team.  
[18:09] == -  
[18:09] == - From time to time, volunteer staff may send server-wide notices relating to  
[18:09] == - the project, or the communities that we host. The majority of such notices  
[18:09] == - will be sent as wallops, and you can '/mode <yournick> +w' to ensure that you  
[18:09] == - do not miss them. Important messages relating to the freenode project, including  
[18:09] == - notices of upcoming maintenance and other scheduled downtime will be issued as  
[18:09] == - global notices.  
[18:09] == -  
[18:09] == - Representing an on-topic project? Don't forget to register, more information  
[18:09] == - can be found on the <https://freenode.net> website under "Group Registration".  
[18:09] == -  
[18:09] == - freenode organises an annual conference, and we would like to extend our  
[18:09] == - thanks to the attendees, exhibitors and speakers who made freenode #live 2017  
[18:09] == - possible. And of course, our generous sponsors: Bytemark, Canonical (Ubuntu),  
[18:09] == - Falanx Cyber Security, Private Internet Access and Yubico for footing the bill.  
[18:09] == -  
[18:09] == - Thank you also to our server sponsors for the sustained support in keeping the  
[18:09] == - network going for close to two decades.  
[18:09] == -  
[18:09] == - freenode #live returns to Bristol, UK on November 3rd-4th 2018. Our Call for  
[18:09] == - Proposals is live at <https://freenode.live> and open until July 31, 2018. If  
[18:09] == - you are interested in sponsoring this event, please send an e-mail to  
[18:09] == - sponsor@freenode.live  
[18:09] == -  
[18:09] == - Thank you for using freenode!  
[18:09] == -  
[18:09] == -  
[18:09] == End of /MOTD command.  
[18:09] == Usermode change: +Ri  
[18:09] == gateway/web/freenode/ip.213.149.61.184 is now your hidden host (set by syn.)  
[18:09] -freenode-connect- Due to the persistent ongoing spam, all new connections are being set +R (block messages from unidentified users) and will be scanned for vulnerabilities. This will not harm your computer, and vulnerable hosts will be notified.  
[18:09] freenode-connect [frigg@freenode/utility-bot/frigg] requested CTCP VERSION from monorkin:  
[18:09] == Connected and logged in -- ready to go!

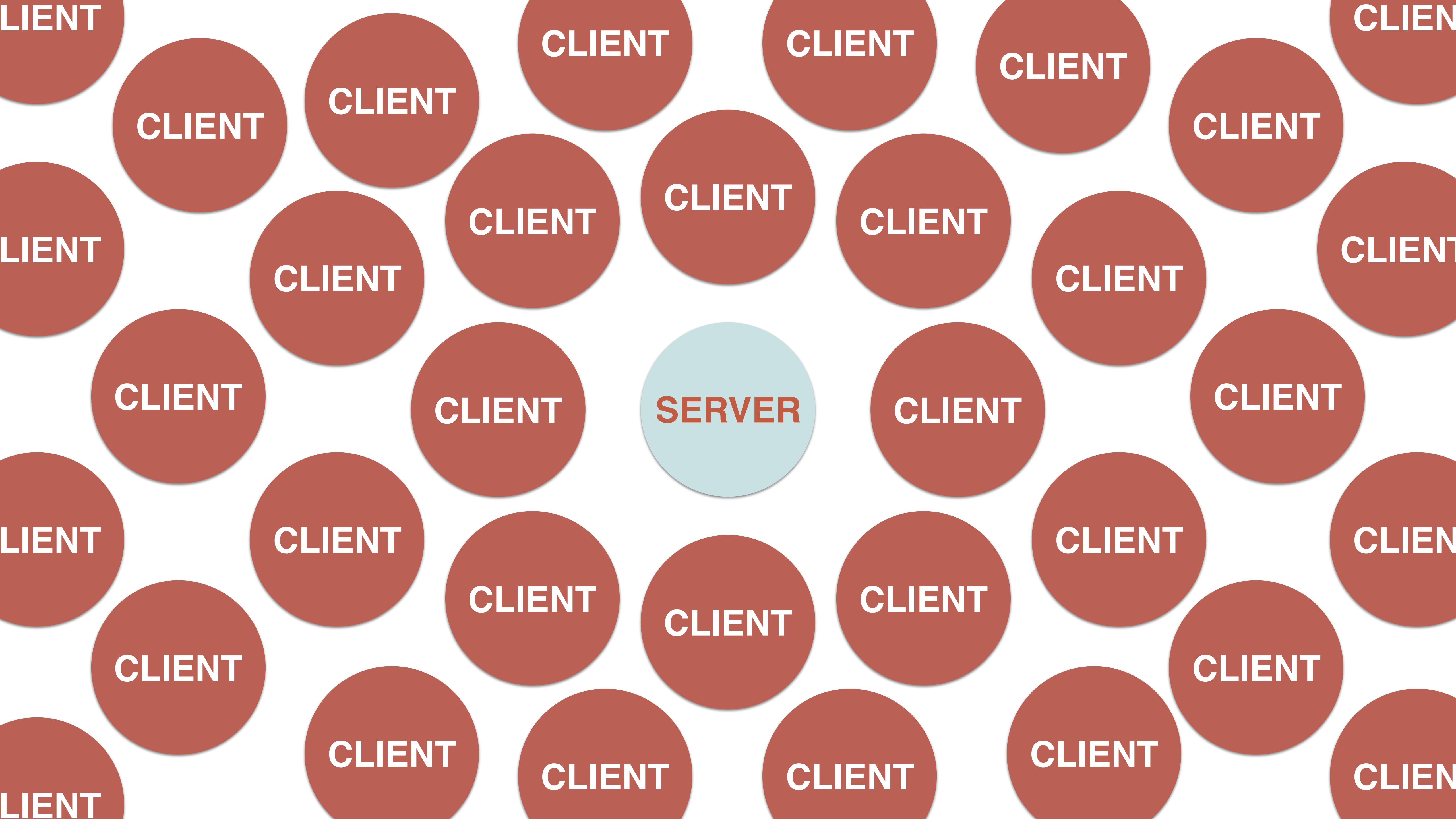
chat here! you can also use commands, like /JOIN or /HELP



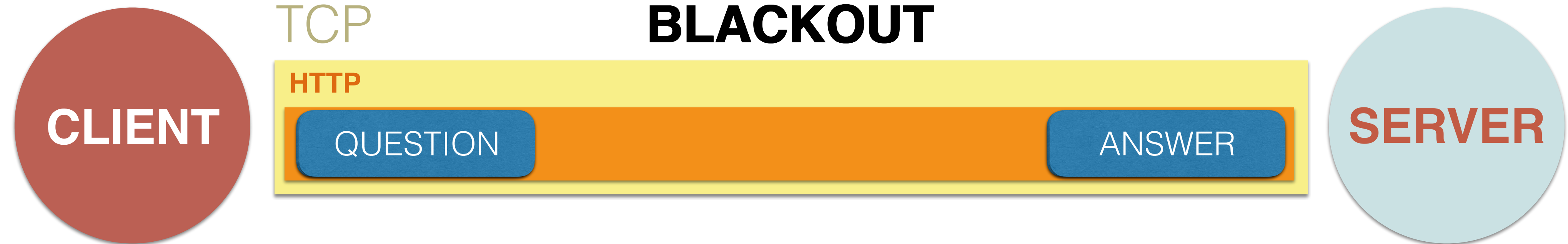








Blackouts



Full-duplex

WEBSOCKETS

# Realtime client-server communication

Introduced in 2011

RFC 6455

The protocol is...



...interesting











Every WebSocket starts out  
as a HTTP request

Part of HTML5 spec



```
1 // Create WebSocket connection.
2 const socket = new WebSocket('ws://localhost:8080');
3
4 // Connection opened
5 socket.addEventListener('open', function (event) {
6     socket.send('Hello Server!');
7 });
8
9 // Listen for messages
10 socket.addEventListener('message', function (event) {
11     console.log('Message from server ', event.data);
12 });
```

websocket-demo.js hosted with ❤ by GitHub

[view raw](#)

WebSocket JS API demo from <https://developer.mozilla.org/en-US/docs/Web/API/WebSocket>

Up to 1024 connections

Joined/Dropped client  
detection

No reconnection handling

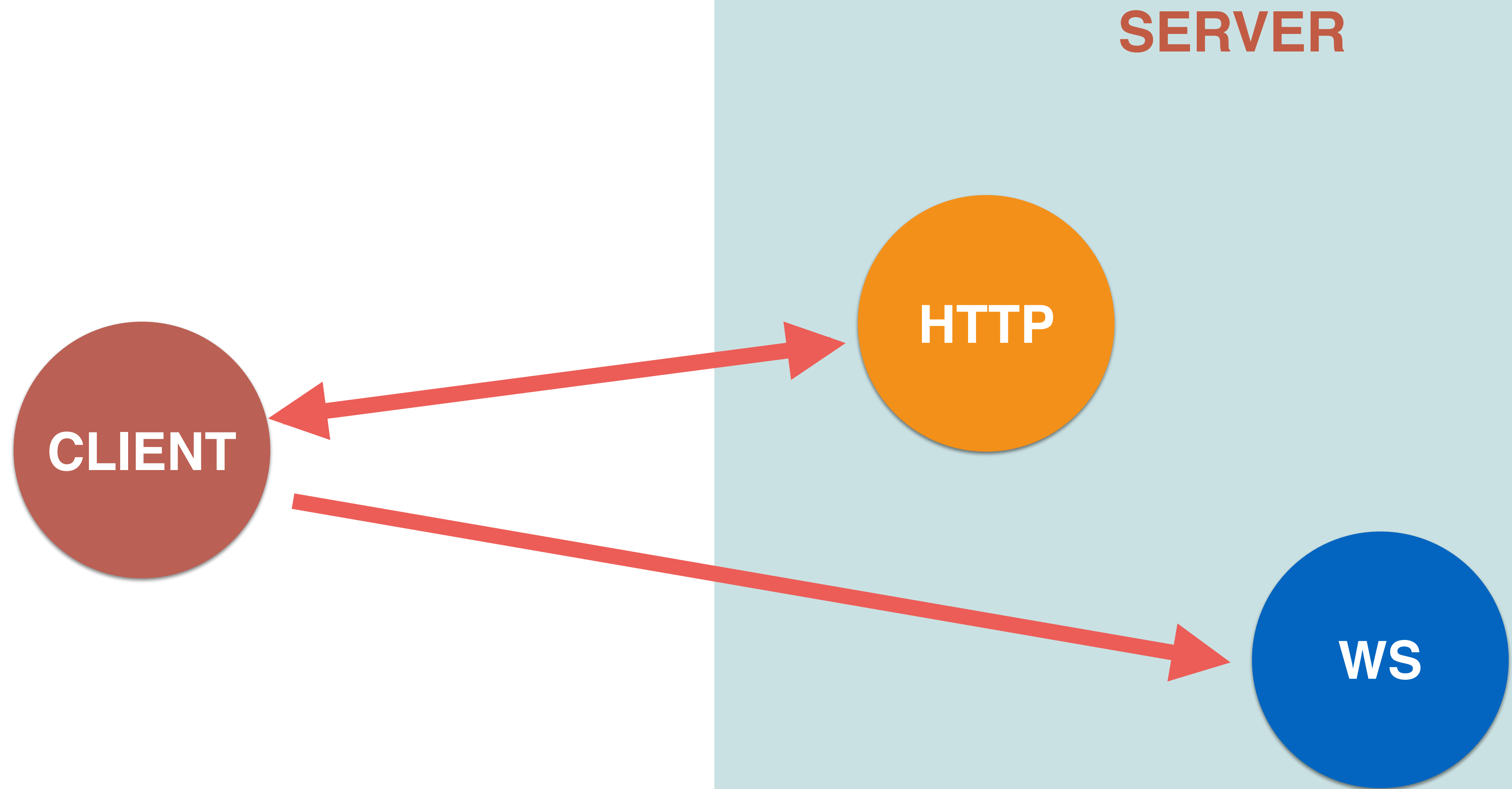
Non standard proxying

Can we solve our chat  
problem with them?

YES!

But...





Overkill

Voice chat

Video chat

Games

Real-time **client to server**  
communication

Alternatives?

SERVER SENT EVENTS

Introduced in 2006

WHATWG Web Applications 1.0

HTTP based

Relay on HTTP/1.1 streaming



Only Server to Client  
communication







Not supported on IE / Edge

```
1  var evtSource = new EventSource('/stream');
2
3  evtSource.onmessage = function(e) {
4      console.log('DATA', e.data);
5  }
```

event-source.js hosted with ❤ by GitHub

[view raw](#)

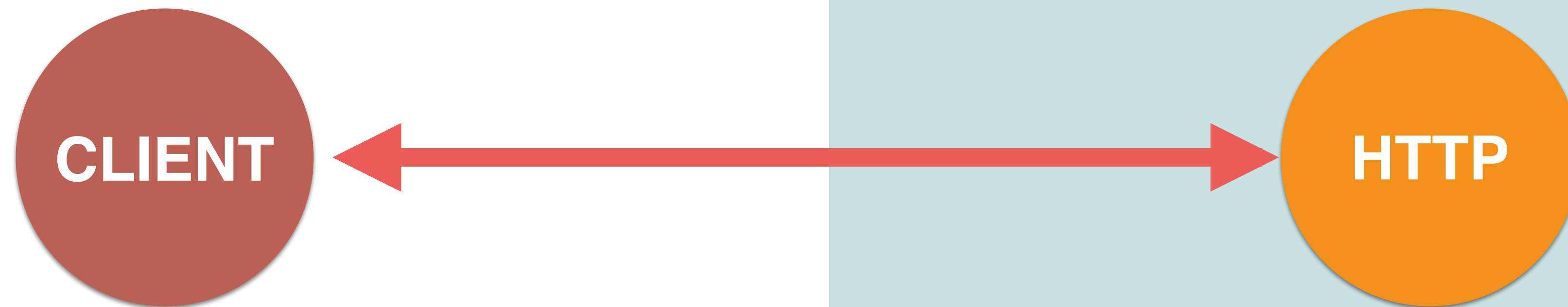
Subject to the  $\sim 6$   
connection limit

# Automatic reconnection handling



No dropped client  
detection

Benefits?



```
31     r.get 'stream' do
32       response['Content-Type'] = 'text/event-stream;charset=UTF-8'
33       q = Queue.new
34       QUEUES << q
35       q << { heartbeat: true }
36       stream(loop: true, callback: proc { QUEUES.delete(q) }) do |out|
37         loop do
38           out << "data: #{q.pop.to_json}\n\n"
39         end
40       end
41     end
42   end
43 end
```

sse\_chat.rb hosted with ❤️ by GitHub

[view raw](#)

Chat app example from — [https://github.com/stankek/lectures/tree/master/19-rabbitmq\\_is\\_more\\_than\\_a\\_sidekiq\\_replacement/demo/01-simple\\_job\\_queue](https://github.com/stankek/lectures/tree/master/19-rabbitmq_is_more_than_a_sidekiq_replacement/demo/01-simple_job_queue)

```
31     r.get 'stream' do
32       response['Content-Type'] = 'text/event-stream;charset=UTF-8'
33       q = Queue.new
34       QUEUES << q
35       q << { heartbeat: true }
36       stream(loop: true, callback: proc { QUEUES.delete(q) }) do |out|
37         loop do
38           out << "data: #{q.pop.to_json}\n\n"
39         end
40       end
41     end
42   end
43 end
```

sse\_chat.rb hosted with ❤️ by GitHub

[view raw](#)

Chat app example from — [https://github.com/stankec/lectures/tree/master/19-rabbitmq\\_is\\_more\\_than\\_a\\_sidekiq\\_replacement/demo/01-simple\\_job\\_queue](https://github.com/stankec/lectures/tree/master/19-rabbitmq_is_more_than_a_sidekiq_replacement/demo/01-simple_job_queue)

```
31     r.get 'stream' do
32       response['Content-Type'] = 'text/event-stream;charset=UTF-8'
33       q = Queue.new
34       QUEUES << q
35       q << { heartbeat: true }
36       stream(loop: true, callback: proc { QUEUES.delete(q) }) do |out|
37         loop do
38           out << "data: #{q.pop.to_json}\n\n"
39         end
40       end
41     end
42   end
43 end
```

sse\_chat.rb hosted with ❤️ by GitHub

[view raw](#)

Chat app example from — [https://github.com/stankec/lectures/tree/master/19-rabbitmq\\_is\\_more\\_than\\_a\\_sidekiq\\_replacement/demo/01-simple\\_job\\_queue](https://github.com/stankec/lectures/tree/master/19-rabbitmq_is_more_than_a_sidekiq_replacement/demo/01-simple_job_queue)

```
31     r.get 'stream' do
32       response['Content-Type'] = 'text/event-stream;charset=UTF-8'
33       q = Queue.new
34       QUEUES << q
35       q << { heartbeat: true }
36       stream(loop: true, callback: proc { QUEUES.delete(q) }) do |out|
37         loop do
38           out << "data: #{q.pop.to_json}\n\n"
39         end
40       end
41     end
42   end
43 end
```

sse\_chat.rb hosted with ❤️ by GitHub

[view raw](#)

Chat app example from — [https://github.com/stankek/lectures/tree/master/19-rabbitmq\\_is\\_more\\_than\\_a\\_sidekiq\\_replacement/demo/01-simple\\_job\\_queue](https://github.com/stankek/lectures/tree/master/19-rabbitmq_is_more_than_a_sidekiq_replacement/demo/01-simple_job_queue)

```
31     r.get 'stream' do
32       response['Content-Type'] = 'text/event-stream;charset=UTF-8'
33       q = Queue.new
34       QUEUES << q
35       q << { heartbeat: true }
36       stream(loop: true, callback: proc { QUEUES.delete(q) }) do |out|
37         loop do
38           out << "data: #{q.pop.to_json}\n\n"
39         end
40       end
41     end
42   end
43 end
```

sse\_chat.rb hosted with ❤ by GitHub

[view raw](#)

Chat app example from — [https://github.com/stankec/lectures/tree/master/19-rabbitmq\\_is\\_more\\_than\\_a\\_sidekiq\\_replacement/demo/01-simple\\_job\\_queue](https://github.com/stankec/lectures/tree/master/19-rabbitmq_is_more_than_a_sidekiq_replacement/demo/01-simple_job_queue)



Text chat  
Notifications

Real-time **server to client**  
communication

LONG POLLING

Pure HTTP

The server delays the  
response













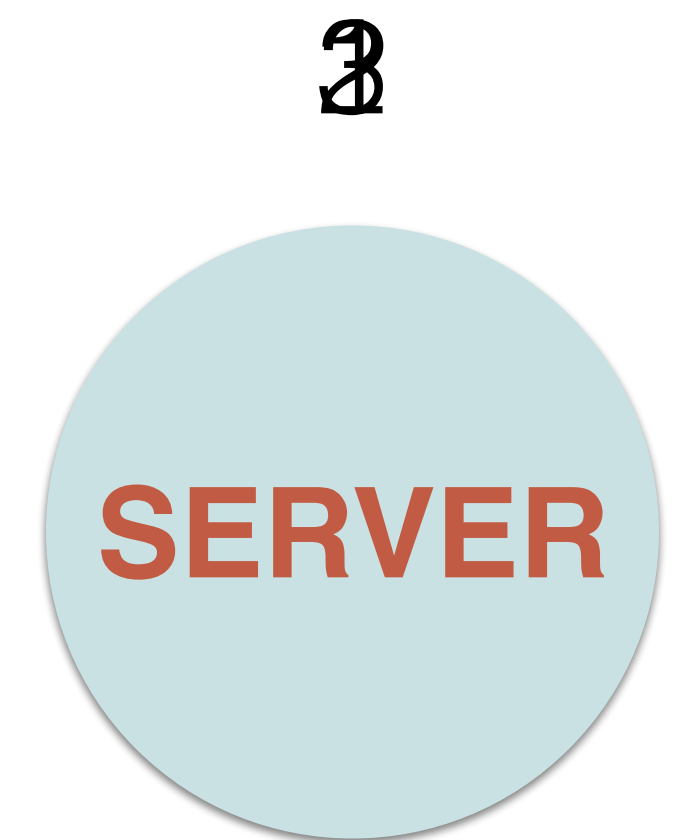


TCP

HTTP

GET /lp

CONTENT

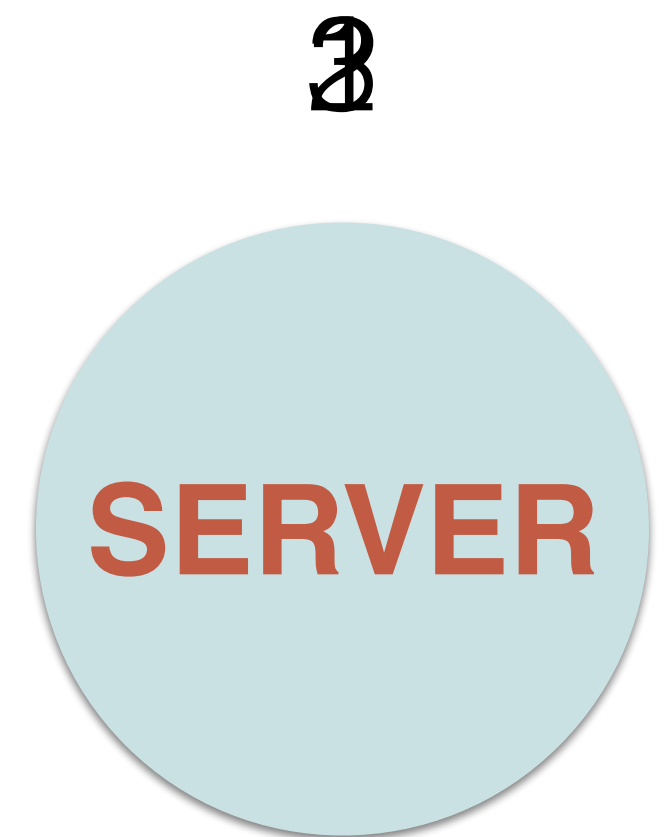
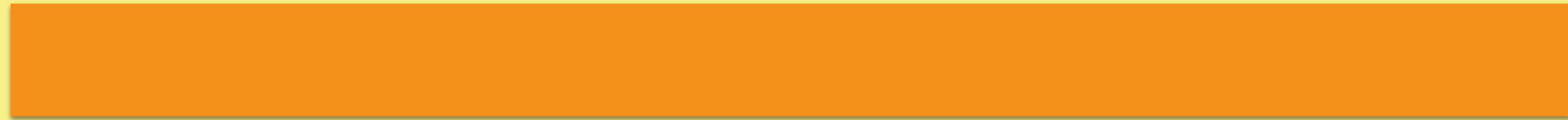


3



TCP

HTTP



3

No native APIs

Blackouts

Subject to the ~6  
connection limit











Supported everywhere



Same controls flow

Better for large numbers  
of users

CONCLUSION

	OPTIONS	eyJrZXkiOil1O...	 app.launchdarkl...	xhr		0 B	0 B	 → 0 ms							
200	OPTIONS	58f55132f1e0...	 app.launchdarkl...	xhr	plain	721 B	0 B	 → 1696 ms							
101	GET	socket?token...	 trello.com	websocket	plain	224 B	0 B	 → 1635 ms							
	GET	eyJrZXkiOil1O...	 clientstream.lau...	xhr		0 B	0 B	 → 606 ms							
200	GET	api.js?_=1538...	 apis.google.com	script	js	5.60 KB	11.95 KB	 → 597 ms							

200	GET	/	localhost:9292	document	html	2.18 KB	2.12 KB	→ 9 ms
304	GET	jquery-3.2.1....	code.jquery.com	script	js	cached	84.63 KB	→ 58 ms
	GET	stream	localhost:9292	xhr				
404	GET	favicon.ico	localhost:9292	img	html	70 B	0 B	→ 6 ms
200	POST	messages	localhost:9292	xhr	html	97 B	33 B	→ 2 ms

Code	Method	URL	Domain	Type	Size	Time
200	POST	s?r=e8b2a60...	webchat.freeno...	xhr	366 B	2 B → 30077 ms
200	POST	s?r=e8b2a60...	webchat.freeno...	xhr	366 B	2 B → 30103 ms
200	POST	s?r=e8b2a60...	webchat.freeno...	xhr	366 B	2 B → 30084 ms
	POST	s?r=e8b2a60...	webchat.freeno...	xhr		

	WebSockets	Server Sent Events	Long Polling
Number of parallel connections from Browser	1024	~6 per domain	~6 per domain
Load Balancing and Proxying	Non-Standard / Complicated	Standard / Easy	Standard / Easy
Supported on all browsers	Yes (90%)	No (84% - not on IE and Edge)	Yes (100%)
Dropped Client Detection	Yes	No	No
Reconnection Handling	No	Yes	No



QUESTIONS





[goo.gl/wZKhkB](https://goo.gl/wZKhkB)

 [github.com/monorkin](https://github.com/monorkin)

 [@monorkin](https://twitter.com/monorkin)

 [monorkin@mastodon.social](https://mstdn.social/@monorkin)

 [hey@stanko.io](mailto:hey@stanko.io)

 [stanko.io](https://stanko.io)