

Содержание

web-socket	3
Установление соединения	3
Передача данных	4

web-socket

В отличие от сокетов транспортного уровня, которые являются интерфейсом, через которые протоколы прикладного уровня обращаются к сервисам транспортного уровня, web socket это протокол, который описывает правила взаимодействия между разными хостами на прикладном уровне.

Web сокет работает через [HTTP](#). Зачем тогда нужен такой протокол «обёртка»? HTTP работает по принципу «запрос - ответ», таким образом сервер не может ничего отправить клиенту, пока он не запросит этого. Web сокет устанавливает постоянное соединение, по которому оба участника соединения могут в любой момент отправлять данные.

Web socket имеет специальный префикс в URL - `ws://` или `wss://` (если совместно в SSL TLS)

Работа протокола состоит из двух частей: установка соединения и передача данных.

Установка соединения

1) Клиент отправляет серверу HTTP GET запрос с опцией Upgrade, в которой указывает протокол, на который нужно перейти. Так же в заголовке указывается случайно сгенерированный ключ web сокета - 16 байт. Ключ предоставляется в кодировке base 64.

```
GET /chat HTTP/1.1
Host: www.asozykin.ru
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: dGhlIHhnbXBsZSBub25jZQ==
Origin: http://asozykin.ru
Sec-WebSocket-Version: 13
```

2) Сервер подтверждает установку соединения, посылая HTTP ответ. Код 101 означает смену протокола. В заголовке `Upgrade`, также указывается протокол на который происходит переключение.

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: s3pPLMBiTxaQ9kYGzzhZRbK+x0o=
```

После этого между клиентом и сервером устанавливается прямое TCP соединение по которому данные передаются уже без использования протокола HTTP.

Передача данных

Web socket поддерживает фрагментацию кадров.

В web сокетах есть кадры трех типов:

- кадры, которые передают текстовую информацию представленную в кодировке utf-8;
- кадры, которые передают данные в двоичном виде;
- управляющие кадры которые используются для поддержания или для закрытия соединения.

From:

<https://wiki.radi0.cc/> - radi0wiki

Permanent link:

https://wiki.radi0.cc/glossary:net:protocols:web_socket

Last update: **2025/11/09 12:07**

