Applied research

CRENIT | The Netherlands

WebSockets

|  |
| --- |
| Petar Dakov  S3-CB04    2023/2024 |

Applied research

2023

Contents

[Introduction 1](#_Toc155301687)

[Main Question 1](#_Toc155301688)

[Sub-questions 2](#_Toc155301689)

[Result 2](#_Toc155301690)

[Implementation 2](#_Toc155301691)

[Use Cases 2](#_Toc155301692)

[Advantages and Challenges 3](#_Toc155301693)

[Advantages 3](#_Toc155301694)

[Challenges 4](#_Toc155301695)

[Security threats & Prevention 4](#_Toc155301696)

[Broken authentication 4](#_Toc155301697)

[Broken access control 4](#_Toc155301698)

[Insecure Direct Object References (IDOR) 5](#_Toc155301699)

[Injection attacks (SQL Injection, XSS, etc.) 5](#_Toc155301700)

[Conclusion 5](#_Toc155301701)

[Bibliography 6](#_Toc155301702)

# Introduction

Have you ever wondered how websites update information instantly without you refreshing the page? Enter WebSockets – a cool technology making this real-time communication possible. Forget about the technical jargon; we're here to explore how WebSockets work, why they matter, and how they're changing the way websites interact. Join us on a journey to uncover the basics and discover what makes WebSockets a game-changer in modern web development!

# Main Question

What are WebSockets, how are they used in “QuizzTurn”, what are the advantages and disadvantages of them, and how to prevent risks when applied?

# 

# Sub-questions

1. How are WebSockets implemented in “QuizzTurn”?
2. Document analysis:
3. Community research:
4. Literature study:
5. What are the most common use cases with WebSockets?
6. Document analysis:
7. Expect interview:
8. Literature study:
9. What are the advantages and disadvantages of WebSockets?
10. Document analysis:
11. Community research:
12. Literature study:
13. How to prevent security threats and risks from happening?
14. Document analysis:
15. Expert interview:
16. Literature study:

# Result

## Implementation

In 'QuizTurn,' real-time chat functionality is seamlessly integrated using WebSockets, powered by the SignalR library. This implementation enables users to engage in instant communication during their quiz interactions, fostering a dynamic and interactive experience within the platform. The WebSocket setup in 'QuizTurn' establishes a persistent connection, allowing users to exchange messages effortlessly and participate in live conversations, enhancing the overall user experience. The use of SignalR further reinforces the application's capability for robust and user-friendly real-time communication.

## Use Cases

**Real-time Feeds:**  
Give your apps the power to provide real-time updates, notifications, social likes, and shares instantly to your users. Imagine how engaging that would be for your audience!

**Real-time Multiplayer Gaming:**  
Use WebSockets API to create a gaming experience similar to mobile devices. Enjoy the ability to send and receive data without constantly polling the webservers, making real-time multiplayer gaming a reality.

**Real-time Data Visualization:**Visualize marketing trends, sales graphs, and analytics in real-time. HTML5 WebSockets enable visually appealing data representations that update automatically as new data arrives, without the need for constant data polling.

**Real-time Multimedia Chat:**HTML5 WebSockets offer a seamless solution for building chat servers. Say goodbye to traditional approaches and embrace a new way to create chat servers that are efficient and production ready.

**Audio/Video Chat with WebRTC:**HTML5 WebSockets serve as an ideal signaling mechanism for WebRTC, enabling full-duplex communication. This is especially useful for creating audio and video chat applications.

**Real-time Sports/Event Updates:**  
Set high standards for your apps by using HTML5 WebSockets to provide real-time updates. Create a common platform for all channels, offering users instant updates on sports, events, and more. (Sood, 2017)

## Advantages and Challenges

Just like any other protocol, WebSocket has its pros and cons. This document aims to present the advantages and benefits WebSocket brings to the table for “QuizTurn”, as well as its disadvantages and limitations.

### Advantages

**Efficient Data Transfer:**WebSockets reduce data transmission overhead with persistent connections, offering lower bandwidth usage and latency compared to HTTP.

**Widespread Adoption:**  
WebSocket is widely adopted, with support across programming languages, frameworks, and most web browsers.

**Flexible Design:**  
WebSocket's design allows easy implementation of custom protocols and extensions, enhancing functionality like pub/sub messaging.

**Event-Driven Push:**  
WebSocket's event-driven model pushes data instantly, making it ideal for quick client responses to unpredictable events.

**Bidirectional Communication:**  
WebSocket enables simultaneous, low-latency messaging between server and client, making it superior for two-way, multi-user real-time applications compared to one-way protocols like Server-Sent Events (SSE).

### Challenges

**WebSockets are not optimized for streaming audio and video data.** A technology like WebRTC is better suited in these scenarios.

**WebSockets don’t automatically recover when connections are terminated** – this is something you need to implement yourself, and is part of the reason why there are many WebSocket client-side libraries in existence.

**Certain environments (such as corporate networks with proxy servers) will block WebSocket connections**. You will most likely have to consider supporting fallback transports for these scenarios, which adds additional engineering complexity and costs to the table.

**WebSockets are stateful,** which makes them hard to use in large-scale systems that consist of multiple WebSocket servers (you need to share connection state across servers). (Diaconu, 2023)

## Security threats & Prevention

Here are some of the most common security vulnerabilities that can occur in “QuizTurn” and how to prevent them from happening through a combination of modern approaches.

### Broken authentication

The impact of a WebSocket authentication vulnerability is severe. Once an attacker has bypassed authentication, they have access to all the data and functionality that the compromised account has.

One possible solution is to use your long-lived authentication token to make a HTTP request for an ephemeral one-time token from your authentication service. Send this one-time token with the WebSocket connection in the query string then discontinue it on the server-side. This way, if the token is logged somehow or shared accidentally, it's no longer usable.

### Broken access control

Broken access control is a common and serious security issue. It occurs when a user can access something or do something they shouldn't. This can lead to unauthorized access, changes, or destruction of data, or performing tasks beyond the user's permissions.

**Preventions**

* Unless a resource is intended to be publicly available, deny access by default.
* Explicitly declare the access that is allowed for each resource.
* Thoroughly audit and test access controls to ensure they’re working as designed.

### Insecure Direct Object References (IDOR)

IDOR vulnerabilities are among the most commonly-reported and are trivial to execute. They can result in unauthorized data exposure, leakage of personally identifiable information, and even create the potential for data manipulation or deletion.

**Preventions**

* Implement access control checks for each object that users try to access.
* Use complex identifiers like GUIDs instead of predictable incremental IDs as a defense-in-depth measure (but remember that access control is crucial even with these identifiers).

### Injection attacks (SQL Injection, XSS, etc.)

A successful DoS results in sluggish behavior or a total system crash. In both instances, the DoS attack deprives genuine users of the service they expected.

Though DoS attacks don't typically result in unauthorized access to sensitive data, they can have a devastating impact on critical services, leading to direct financial loss and reputational damage.

**Preventions**

* Only allow authenticated users to open a WebSocket connection to prevent an attacker from flooding the WebSocket server with a large number of connection requests.
* Ensure that your authentication mechanism is efficient so that DoS attacks using an invalid token can be quickly and cheaply refused.
* Limit the number of requests that can be made to the WebSocket server from the same IP address over a specific time period.
* Limit message size to a reasonable number to prevent message size attacks. (Booker, 2023)

## Conclusion

In summary, WebSockets, implemented in “QuizTurn”, revolutionize real-time web communication. They efficiently connect clients and servers for dynamic data exchange, powering applications for collaborative editing and more.

WebSockets excel in data transfer efficiency, widespread adoption, and flexible design. Their event-driven push and bidirectional communication make them vital for modern apps. Challenges include streaming optimization and statefulness.

Ensuring WebSocket security for “QuizTurn” is crucial. This document addresses threats like broken authentication and access control issues, providing practical preventive measures. As we navigate web development, embracing WebSockets' strengths and fortifying security practices becomes essential.

# Bibliography

Booker, A. (2023, August 21). *websocket-security*. Retrieved from ably: https://ably.com/topic/websocket-security

Diaconu, A. (2023, April 25). *websockets-pros-cons*. Retrieved from ably: https://ably.com/topic/websockets-pros-cons

jfriend00. (2016). *How Websockets are implemented?*

Sood, V. (2017, September 29). *10 Most Amazing Use Cases of WebSockets*. Retrieved from medium: https://medium.com/the-developer-journal/10-most-amazing-use-cases-of-websockets-go-real-time-166b71e0e711