

Safebox Challenge

Challenge Overview

Welcome to the "Safebox Challenge." In this challenge, you are tasked with implementing a secure REST API that allows users to store their sensitive data in a way that ensures the utmost confidentiality. The goal is to protect user information from unauthorized access, providing a safe environment for data storage.

Challenge Instructions

Your mission is to create a RESTful API with the following key requirements:

1. **User Registration:** Implement an endpoint for user registration. Users should be able to sign up with a username and a strong, securely hashed password. Ensure the security of user credentials and protect against common security vulnerabilities like brute force attacks.
2. **User Authentication:** Develop an endpoint for user authentication. Registered users should be able to securely log in with their credentials. Use strong encryption and authentication mechanisms.
3. **Data Storage:** Create an endpoint that allows authenticated users to securely store their sensitive data. This data should be stored in a manner that is resistant to data leaks, breaches, or unauthorized access.
4. **Data Retrieval:** Implement an endpoint for authenticated users to retrieve their stored data. Data should only be accessible to the user who stored it, ensuring data privacy and security.
5. **Security Measures:** Utilize best security practices throughout your implementation, such as encryption, hashing, and other appropriate security mechanisms. Ensure that sensitive data is properly protected and not exposed to unauthorized users.
6. **Documentation:** Provide clear and comprehensive documentation for your API, including endpoint descriptions, authentication procedures, and data storage methods.

Evaluation Criteria

Your challenge solution will be evaluated based on the following criteria:

- **Security:** The effectiveness of security measures in place to protect sensitive user data.
- **Functionality:** The functionality and correctness of the API, ensuring that it meets the specified requirements.
- **Documentation:** The quality and clarity of the provided documentation, making it easy for others to understand and use the API.
- **Code Quality:** The organization, readability, and adherence to best practices in your code.

- **Authentication:** The strength and reliability of the authentication and user registration processes.

NB: Don't forget to add the challenge author to your repo and make it private.

Best of luck, and may your implementation provide a secure safebox for sensitive user data.