QuizMaker Security Assessment

By Michael Olson

Dec 12, 2020

Table of Contents

1. Introduction

This document presents the security team's evaluation of the QuizMaker website. The security evaluation was conducted to test QuizMaker for vulnerabilities. This evaluation covers threats that may appear through the intended use of QuizMaker. Regardless of testing, the technology supporting the product and the methods used to attack it continually advance. The focus of this report is to identify security threats, the threatened assets, and to suggest steps that can be taken to mitigate the risk posed by these threats.

1.1 Methodology

1.1.1 Asset identification

This vulnerability assessment is based around the idea that there are assets associated with QuizMaker that should not be accessed. The assembled list is not exhaustive but does represent those assets that are at risk of unauthorized access.

1.1.2 Threat identification

Identifying potential threats is a critical step in probing QuizMaker's security. To accurately identify and test attack vectors the type and typical execution of those threats must be known. This process also serves as a way to curate the most likely threats and ensure that testing is not only comprehensive but efficient.

1.1.3 Vulnerability appraisal

Once a list of potential threats are known the next step is to identify the vulnerabilities within QuizMaker that those threats may leverage to gain access to QuizMaker's assets.

1.1.4 Risk Assessment

Knowing what assets are at risk and the potential vectors of attack is not enough. It is crucial to identify the impact that attacks would have in exposing assets.

1.1.5 Risk Mitigation

Risk mitigation identifies what can be done to stop or reduce the risk of attacks through the identified vectors. Since technology is continuously advancing we may at most be able to reduce the risk of attack. Several suggestions are presented that the security team believes will do just this.

2. Assessment

2.1 Assets

The following assets have been shown to be at risk of exposure through QuizMaker.

The users database. Within the users database the following information exists and is available for exposure:

The users database used by QuizMaker contains the users names, emails, professor status, their enrolled classes, and their taught classes. This information is stored as json files and represents the majority of sensitive information. Due to this access to and the security of this database is a priority.

The quizzes database contains the user created quizzes and a quiz insertion method. This information represents the intellectual property generated by the users. This information is not especially sensitive but should be protected from unauthorized access and tampering.

The courses database contains the courses available, the professors of those courses, which students are enrolled in and the ability to create a new course. This information contains some identifying information and should be protected from access and tampering.

The code that makes up the front end of QuizMaker is a valuable asset. However the public availability of the code is not an issue. Regardless of access to the source code QuizMaker should be protected from unauthorized access.

2.2 Threats

The following list details the threats we have deemed to be of the greatest concern. Those that are not listed

A Man in the middle attack makes use of a terminal between two endpoints that is utilized to access and alter exchanged information. This process can be used to log in as one person, alter the identifying information and send the falsified data to the backend.

Javascript manipulation acts similarly to the man in the middle attack in that it allows data to be falsified. The difference is that the attacker only needs a browser to manipulate the data.

Dictionary Attacks are a popular way to attack passwords. They attempt to log in using a dictionary of potential passwords. Each is tried until the potential passwords run out or the attacker gains access to the database.

A DDOS attack is when a site is maliciously overwhelmed by the sheer number of terminals accessing the site, rendering the site inoperable. This is a very powerful attack that has a number of effects. Service is disrupted, costs of operating the host server may increase, attacks may be camouflaged, and unsaved data may be lost.

2.3 Vulnerabilities

The following list details the vulnerabilities identified in QuizMaker.

1. The lack of JWT or any form of authorization checks makes it possible for attackers to alter identifying information, impersonate anyone, and access anyone's account. Implementing JWT would protect against Man in the Middle, and Javascript manipulation attacks. This vulnerability should be addressed prior to release.
2. HTTPS is not required by QuizMaker. This allows information to be transmitted unencrypted.
3. The database methods are available to anyone who knows the address. One of these dumps the database revealing all information on it. This is a particularly easy vector of attack to access for anyone knowledgeable enough to know about it. It is through this vector of attack that others may gain access to the users, quizzes, and courses databases. Restricting access to these methods is critical to securing database information. This vulnerability should be addressed prior to release.
4. In browser javascript access before login. Allowing unauthorized users to access the javascript code allows attackers to scour the code and look for vulnerabilities. Disallowing access to the javascript is not an option as the code must be loaded on local machines to run and load the website. However restricting access until after the user is authorized adds another layer of security. This vulnerability is not critical to security but should be addressed when time allows.
5. Unmanaged database connections make it extremely easy for attackers to overwhelm the database and render it unusable. This effectively shuts down the site and forces admins to reset the database. This vector enables the use of DDOS attacks with relative ease and should be addressed prior to release. This vulnerability should be addressed before release.

2.4 Risk Assessment

Impact from exposure is described in terms of impact on QuizMaker, and the sensitivity of released information. The four most sensitive assets are: user names, user emails, professor status, and student enrollment. Usernames and emails, as identifying information, are the most sensitive. With this knowledge users can be researched and their accounts can come under attack. A user's professor status and the student enrollment lists are less sensitive, however they do provide information about users that is not explicitly publicly available. In terms of impact on QuizMaker there exists the possibility of legal repercussions from users whose data was exposed. If a data breach occurs, QuizMaker is required to inform users within a set amount of time. This in turn may result in a lack of trust in the service.

2.5 Risk Mitigation

1. Use HTTPS exclusively. Minimally use HTTPS during log in.
2. Implement JWT to authenticate user sessions.
    a. Ensure that JWT tokens expire after a set amount of time so that stolen tokens only provide limited access
3. Set up the login page as a separate microservice to isolate the core javascript from unauthorized parties.
4. Restrict access to database methods.
5. Manage database connections so that standard use does not crash the database

3. Conclusions

Vulnerabilities were found in the javascript, lack of JWT, lack of HTTPS, and in unmanaged database connections. To address these vulnerabilities it is suggested that JWT be implemented, login be made a separate microservice, enforce HTTPS for log in, and better management of database connections.

Of the suggested changes, the team has implemented JWT, and better managed the database connections. The team implemented JWT in such a way that it also solves the issues in the javascript allowing users to spoof login credentials, and restricts access to the backend methods. The database connections have been managed such that the database can now handle approximately 800 connections/second. At this time the log in process has not been made into its own micro service and HTTPS connections are not mandatory. These changes were not implemented due to time constraints and are not product breaking issues, at most they represent minor issues.

This report details the methods believed by the security team to be the most pressing and likely. The security team concludes, based on the completed security measures, that QuizMaker is reasonably secure.