RAC Web Application System Security Plan

Maxwell Haley

Computer Science – Sixth Semester

2018-04-28

Table of Contents

[User Authentication 3](#__RefHeading___Toc2919_1467112414)

[User Authorization 3](#__RefHeading___Toc2921_1467112414)

[Encryption 3](#__RefHeading___Toc2923_1467112414)

[Confidentiality of Data 3](#__RefHeading___Toc2925_1467112414)

[Use of SSL on the Server 3](#__RefHeading___Toc2927_1467112414)

[Backup of Data 3](#__RefHeading___Toc2929_1467112414)

[Privacy Considerations 4](#__RefHeading___Toc2931_1467112414)

[Database Security 4](#__RefHeading___Toc2933_1467112414)

[Protection Against Path Truncation and Reverse Directory Traversal 4](#__RefHeading___Toc2935_1467112414)

[Protection Against XSS and SQL Injection 4](#__RefHeading___Toc2937_1467112414)

[Testing Techniques 4](#__RefHeading___Toc2939_1467112414)

# User Authentication

User authentication is handled by the Microsoft ASP.NET Membership system. Membership handles storing both standard user information—name, address, contact information, etc—and confidential user information, specifically passwords. When a user enters their email and password, Membership confirms the credentials are valid.

# User Authorization

User authorization is handled by the CSAdmin system. When a user logs in with their Heritage College LDAP credentials CSAdmin confirms which role that user has—if any. Currently the system has a single role—RAC Advisor—but this may expand in the future. If it does, using CSAdmin as our centralized authorization platform makes adding new roles to RAC very simple.

# Encryption

The only thing encrypted inside the RAC Application are user passwords. The encryption is managed by the Membership program, which ensures industry standard hashing algorithms are being used. This also prevents oversights by the developers creating security holes that could be exploited to gain access to plaintext passwords.

# Confidentiality of Data

Some of the documents that can be uploaded to the system contain *Protected B* information. This information must be handled carefully and the Candidate must know the risks involved with uploading this information. A *Privacy Policy* acceptance is manditory before registering an account. This policy outlines the risks involved with sharing *Protected B* information and who can view this information once uploaded. Only RAC Advisors can view documentation uploaded by a Candidate.

# Use of SSL on the Server

SSL is enabled on the production server the application will be deployed on. This prevents important information such as passwords or documentation from being intercepted. This is set-up by the operations team, not by the developers.

# Backup of Data

Backup of data is not handled by the development team.

# Privacy Considerations

As mentioned before, some documents uploaded may fall under the *Protected B* status. This means the documents contain: *“information or assets that, if compromised, could* ***cause serious injury*** *to an individual, organization or government”[[1]](#footnote-2)*. This means we are expected to ensure that no one other than the Candidate and RAC Advisor can view a Candidates documents within the system. Other Candidates cannot view another Candidates files and files should only be retrievable after being authenticated and authorized.

# Database Security

The database is not owned by the development team. The only security steps the developers can take is by creating a single user that read/write to the database. This user is unique to the RAC system and has no other permissions.

# Protection Against Path Truncation and Reverse Directory Traversal

The system protects against path truncation and reverse directory traversal by comparing the user ID/role stored in the session server side against the ID/role the page is meant for. A malicious user cannot change their session without compromising the server itself. If a non-authorized user attempts to navigate to any RAC Advisor page, they will be sent back to the login page. If a Candidate attempts to trigger a download for a document that is not associated with their ID, the server will deny the download. All of this checking is done server side, preventing client interference.

# Protection Against XSS and SQL Injection

To protect against XSS attacks, all input that may be shown to screen is shown literally without any evaluation. A script tag added to a Candidates name will appear verbatim and will not be recognized as valid HTML by the browser. *Entity Framework—*used for all database calls—prevents SQL injection by parametrizing all SQL transactions. This means input from the user is taken verbatim and not interpreted as a raw SQL script. This prevents malicious users from directly reading/writing to the database.

# Testing Techniques

Security testing is done manually, by attempting to attack the system using common vectors like the ones mentioned above. Unit testing is also used in the cause of verifying that a given ID has ownership over a RAC request or document.

1. https://www.tpsgc-pwgsc.gc.ca/esc-src/protection-safeguarding/niveaux-levels-eng.html [↑](#footnote-ref-2)