Oracle® Healthcare Translational Research

Secure Development Guide

Release 3.2

E87268-01

May 2017

The Oracle Healthcare Translational Research (OHTR) *Secure Development Guide* provides guidance on how to avoid introducing security flaws when developing code for applications that use the OHTR REST APIs and while using these applications. It helps developers be aware of the security tools and features they can use to implement application security.

This set of recommendations is not exhaustive and does not replace a formal code review process.

The Open Web Application Security Project (OWASP) publishes the OWASP Top 10 Application Security Risks to identify some of the most critical risks. This document briefly describes the top risks that apply and provides mitigation strategies. Users are encouraged to extend these strategies to secure their own applications and environments that use OHTR REST APIs.

This guide describes the practices and strategies used in OHTR to mitigate security vulnerabilities and is based on the OWASP Top 10 Application Security Risks as available at `https://www.owasp.org/images/f/f8/OWASP_Top_10_-_2013.pdf`.

The Top 10 risks include:

- Injection
- Broken Authentication and Session Management
- Cross Site Scripting (XXS)
- Insecure Direct Object References
- Security Misconfiguration
- Sensitive Data Exposure
- Missing Function Level Access Control
- Cross-Site Request Forgery (CSRF)
- Using Components with Known Vulnerabilities
- Non-validated Redirects and Forwards

Injection

SQL Injection can occur when untrusted data is used in a command or query when an attacker adds SQL commands that are executed against the connected database. To protect against this, OHTR does the following:

- Avoids dynamic SQL generation.

- Uses bind variables.
- Validates user input when possible.

You should extend this approach when you use the OHTR REST APIs.

Broken Authentication and Session Management

The OHTR UI uses OPSS for authentication. OHTR REST API applications should take advantage of WebLogic's user authentication functionality. Follow these guidelines:

- Avoid building a custom authentication system. The risk of insecure login credentials or session IDs increases with custom built authentication schemes.
- Do not store passwords (including third party) in the database, even though it is encrypted. Instead, store them in a WebLogic credentials store.
- Provide appropriate error messages: do not provide specific or detailed information on why the authentication failed. You may provide the details in an application log file for Support.
- Use TLS 1.2 for the traffic between user browser and the application server in a production environment.
- Use strong passwords and maintain security of account credentials.

Cross Site Scripting (XXS)

Not applicable for OHTR REST APIs.

In the OHTR UI, this is handled by using ADF security.

Insecure Direct Object References

Not applicable to OHTR REST APIs or the OHTR UI.

Security Misconfiguration

Attackers can take advantage when the security configuration is incorrect or incomplete and obtain unauthorized access to an application. The entire technology stack must be configured properly, including ADF, and processes should be in place to detect misconfigurations.

Ensure that your application does not contain opportunities for an attacker to gain unauthorized access to the system. Take care to secure default accounts, files, directories, servers, the network, and other data access channels outside of the API domain. See the OHTR *Security Guide*.

Sensitive Data Exposure

Be careful to hide sensitive information from unauthorized users.

- **Query predicates** One of the best ways to ensure sensitive data is kept that way is to retrieve it from the database only on a "need to know" basis. This is much safer than fetching all the data to the middle tier and then filtering or redacting information that the current user is not supposed to have access to.

- **Use proper data authorization**. In OHTR, the Virtual Private Database (VPD) is used to protect sensitive data.
- In the OHTR UI, do not increase the session timeout.

The OHTR UI uses **ADF Faces** to lock down the data access to the UI layer by dynamically controlling UI elements.

Missing Function Level Access Control

Use the defense in depth design pattern. Use authorization checks to guard application functionality that executes methods and operations. Never assume that a specific method will only be called within the context that it was initially designed for.

Users must have an appropriate application role to be able to execute a particular OHTR REST API.

In the OHTR UI, all the functions are protected by ADF Custom Resource Permissions and Security EL expressions, leveraging the ADF Security Context.

Cross-Site Request Forgery (CSRF)

Cross-site request forgery is the risk of a third-party request to a web application on behalf of an authenticated user. This is taken care of in the OHTR UI by using the ADF Framework.

Not applicable to OHTR REST APIs.

Using Components with Known Vulnerabilities

If an older version of a component with a known vulnerability is deployed in an environment, then an attacker who is aware of the vulnerability can take advantage and obtain unauthorized access. To guard against this:

- Use only third party components that are approved for use by Oracle Corporation.
- Adopt patches and vendor updates as quickly as possible, especially if a patch contains a security-related fix. Use the latest version available that has been approved to use.
- Be proactive about keeping up to date with releases of Oracle ADF, WebLogic and Oracle Critical Patch Updates.

Non-validated Redirects and Forwards

Not applicable for the OHTR UI or REST APIs, as these applications do not do forwards or redirects.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Oracle Healthcare Translational Research Secure Development Guide, Release 3.2
E87268-01