

Oracle® Healthcare Data Repository

Secure Development Guide



Release 8.1.3

F52479-01

July 2022

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Copyright © 2018, 2022, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Documentation accessibility	iv
Diversity and Inclusion	iv

1 Secure Development Guide

OWASP Top 10 Security Vulnerabilities 2017	1-1
Injection	1-1
Broken Authentication	1-1
Sensitive Data Exposure	1-2
XML External Entities (XXE)	1-2
Broken Access Control	1-2
Security Misconfiguration	1-2
Cross Site Scripting (XSS)	1-3
Insecure Deserialization	1-3
Using Components with Known Vulnerabilities	1-3
Insufficient Logging and Monitoring	1-3

Preface

This preface contains the following sections:

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>.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

Secure Development Guide

The *Secure Development Guide* provides an overview of the security options for customers who will use Healthcare Data Repository (HDR) user database accounts and middle tier WebLogic user accounts to access the HDR APIs. Note that the set of recommendations in this document is not exhaustive and that no guarantee is given that implementing all the suggestions in this document provides sufficient protection for all security threats. The reason for this disclaimer is that you cannot delegate responsibility for secure application development to a third party or a single document. This document is to help developers be aware of the security tools and features that they can use to implement application security. This document does not replace a formal code review process.

Guidelines are presented here to assist in mitigating common security risks when customers are using the HDR APIs. The Open Web Application Security Project (OWASP) publishes the OWASP Top 10 to identify some of the most critical application security risks. This document briefly describes each Top 10 risk, provides the HDR mitigation strategies, and encourages our users to extend these strategies to secure their own applications and environments that use our APIs. Some of the web-specific Top 10 items don't apply to HDR; these are marked as **Not Applicable**.

OWASP Top 10 Security Vulnerabilities 2017

This paper discusses the practices and strategies used by the HDR application to mitigate risks posed by the security vulnerabilities documented in the OWASP Top 10 – 2017. Customers using the HDR APIs should be aware of and protect against these threats as well.

For the OWASP Foundation's description of the OWASP Top 10 Application Security Risks see the OWASP Top 10 - 2017 document (https://owasp.org/www-pdf-archive/OWASP_Top_10-2017_%28en%29.pdf.pdf).

Injection

SQL Injection can occur when untrusted data is used in a command or query. If an attacker sends hostile data, this can result in executing harmful commands or in unauthorized access of data.

The HDR APIs accept query parameters which could potentially contain hostile data. HDR code adheres to the recommended standards to avoid SQL Injection possibilities by using bind variables and by checking and validating the user input.

Customers using the HDR APIs to build applications should code carefully, using proper data validations and checking user input where needed.

Broken Authentication

The risk of insecure login credentials increases with custom built authentication schemes.

The HDR APIs are secured using the standard WebLogic authentication mechanism. Follow the WebLogic recommendations on creating and configuring a suitable authentication

provider where the HDR WebLogic user accounts will be stored. Refer to <https://docs.oracle.com/middleware/12213/wls/SECMG/toc.htm>.

Avoid building a custom authentication provider for using the HDR APIs. Use strong passwords and maintain security of account credentials.

Sensitive Data Exposure

Attackers may obtain unauthorized access to poorly protected sensitive data. Caution should be used to hide sensitive information from unauthorized users.

User authorization is not part of the HDR application itself and we strongly recommend that the applications developed using the HDR APIs have their own user management, authorization, and access control mechanisms to allow controlled access to data stored in HDR.

Also, transmission of data between HDR and client applications is always encrypted using TLS protocol.

XML External Entities (XXE)

Attackers can exploit vulnerable XML processors if they can upload XML or include hostile content in an XML document, exploiting vulnerable code, dependencies, or integrations.

In HDR, incoming XML is validated using XSD and a server-side data validation is implemented to prevent hostile (vulnerable) data coming from XML documents, headers, and nodes. HDR SOAP services use the SOAP 1.2 standard. HDR FHIR uses JSON data for data exchange.

The HDR build process is integrated with Fortify (static code analysis tool) to identify security vulnerabilities.

REST end points are tested using WebInspect tool to identify vulnerabilities within the rest API web layer.

Broken Access Control

Access control enforces policy such that users cannot act outside of their intended permissions.

APIs or resources in HDR are protected resources. Only valid users and those requests that contain proper access tokens and scopes can access protected resources. The application delegates the authorization decision to the WebLogic Server OPSS security framework. HDR also uses OAuth 2.0 authorization.

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, and processes should be in place to detect misconfigurations including missing patches, accounts with default passwords, insecure settings in frameworks or libraries, and so forth.

The HDR, being an on-premise application, must be deployed in a secure WebLogic instance.

Ensure that your WebLogic configuration where HDR is deployed does not contain opportunities for an attacker to gain unauthorized access to the system. Take care to secure default accounts, files or directories, servers, the network, and other data access channels of the HDR deployment and APIs.

Cross Site Scripting (XSS)

Not applicable for access to HDR APIs.

Insecure Deserialization

Applications and APIs will be vulnerable if they deserialize hostile or tampered objects supplied by an attacker.

HDR does not accept serialized data from external client applications.

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 could take advantage and obtain unauthorized access.

HDR is built on a technology stack where patches and new releases offer improvements, including security-related modifications. The HDR application is an on-premise application and users should download and apply recommended security patches for the respective versions of components like WebLogic server and Oracle Database.

HDR users should follow the same policy of applying patches and updating to the latest versions of components being used, especially if security vulnerabilities have been reported in older versions.

Insufficient Logging and Monitoring

Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack business systems.

HDR API access is logged as part of an audit framework. The audit log contains key details such as user ID, resource name, timestamp, request URL, request IP address, and so forth.