

Oracle® Service Architecture Leveraging Tuxedo (SALT) Release Notes



Release 22c (22.1.0.0.0)

F70809-03

October 2022

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

ORACLE®

Oracle Service Architecture Leveraging Tuxedo (SALT) Release Notes, Release 22c (22.1.0.0.0)

F70809-03

Copyright © 1996, 2022, Oracle and/or its affiliates.

Primary Authors: Priya Pathak, Tulika Das

Contributors: Maggie Li

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, software documentation, data (as defined in the Federal Acquisition Regulation), 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," "commercial computer software documentation," or "limited rights data" 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

1	Oracle SALT 22c (22.1.0.0.0) New Features	
1.1	Integrate with Oracle Transaction Manager for Microservices	1-1
1.2	Support JWT Token Authentication	1-1
1.3	Updates to SALT Security	1-2
2	Upgrade Considerations	
3	SALT Platform Support	
4	Interoperability Considerations	
5	Major Enhancements Post Oracle SALT Release 12.2.2	

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.

1

Oracle SALT 22c (22.1.0.0.0) New Features

This release note contains new features and enhancements incorporated in Oracle SALT Release 22c (22.1.0.0.0).

1.1 Integrate with Oracle Transaction Manager for Microservices

In this release, SALT has been enhanced to allow Oracle Tuxedo services to participate in global transactions coordinated by Oracle Transaction Manager for Microservices.

Oracle Transaction Manager for Microservices enables enterprise users to adopt and increase use of microservices architecture for mission-critical applications by providing capabilities that make it easier to develop, deploy, and maintain such applications. You do not need any additional configuration to integrate SALT with the Oracle Transaction Manager for Microservices.



See Also:

[Oracle® Transaction Manager for Microservices Developer Guide](#)

1.2 Support JWT Token Authentication

JWT Token Authentication

You can use the new **TrustedIdpCert** element for JWT token authentication. Add this element in the block of Certificate. Next, set the value of this element to the name of the file that contains a list of PEM formats of X509 certificates.

When GWWS receives a REST inbound request, it checks the HTTP header. If there is an Authorization: Bearer header present, then GWWS assumes that the request uses the JWT bearer token. GWWS does the credential mapping if the JWT token is valid. By default, GWWS uses the sub claim in JWT as the Tuxedo username.



Note:

If you set the Tuxedo SECURITY to `NONE` in the UBBCONFIG file, then GWWS does not validate the JWT token. In this case, the Authorization header is ignored.

Example

```
<System>
  <Certificate>
    <TrustedIdpCert>idp.pem</TrustedIdpCert>
```

```
</Certificate>  
</System>
```

1.3 Updates to SALT Security


This release promotes a more secure environment by default. This section describes the default security behavior and the environment variables needed for backward compatibility.

- *TM_MIN_PUB_KEY_LENGTH*: When you use HTTPS, for RSA, the minimum key length is 2048. When you load the key/certificate, GWWS detects the key length. If the key length is smaller than 2048, it will fail to boot. In case you want to use a shorter key length, then use the environment variable *TM_MIN_PUB_KEY_LENGTH*.
- *TM_TLS_FORCE_VER*: TLS 1.2 is used by default. To use a different version of TLS (for SSL servers), use the environment variable *TM_TLS_FORCE_VER* to specify the forced TLS version.
- Following is the list of cipher suites supported by default:
 - TLS_RSA_WITH_AES_256_CBC_SHA256
 - TLS_RSA_WITH_AES_256_GCM_SHA384
 - TLS_RSA_WITH_AES_128_CBC_SHA256
 - TLS_RSA_WITH_AES_128_GCM_SHA256In case you want to use other cipher suites, use the environment variable *TM_CIPHERSUITES* to explicitly specify the ciphers. For example, to interoperate with old versions of Oracle Tuxedo.

2

Upgrade Considerations

If a previous SALT release is installed, then you must uninstall it before installing Oracle SALT 22c (22.1.0.0.0). This is because the current installation of the software cannot co-exist with a previous SALT installation if the Oracle Tuxedo installation is the same.

 **See Also:**

Upgrading from Tuxedo Previous Releases to Tuxedo 22c Release

3

SALT Platform Support

Oracle SALT 22c (22.1.0.0.0) supports the following platforms:

- Oracle Linux 7 (64-bit) on x86-64
- Oracle Linux 8 (64-bit) on x86-64
- Red Hat Enterprise Linux 7 (64-bit) on x86-64
- Red Hat Enterprise Linux 8 (64-bit) on x86-64
- SUSE Linux Enterprise Server 12.5 (64-bit) on x86_64

4

Interoperability Considerations

This section describes interoperability of Oracle SALT 22c (22.1.0.0.0) with internal and external products.


Oracle SALT 22c (22.1.0.0.0) certifies the interoperability with WLS 14.1.1. SALT supports most external web service applications that are described using WS-TX standard documents. For more information, see [WS-TX Interoperability Table](#)


For products that are certified with previous releases, see [Certified Interoperable Web Service Server Toolkits](#)

5

Major Enhancements Post Oracle SALT Release 12.2.2

The following section describes the major enhancements made to Oracle SALT 22c (22.1.0.0.0) post the Release 12.2.2.

BugDB Number	Description
Bug 29697086	Support long REQUEST-URI up to 8192 characters
Bug 29123988	GWWS SSL handshake supports multi-threading. A new variable <code>GWWS_SSL_HANDSHAKE_MULTITHREADING</code> has been added. If the value of this variable is set to <code>Y</code> , GWWS leverages multiple threads to handle the SSL handshake.
Bug 28694696	WSDL supports <code>soapenc:string</code>
Bug 28190520	Allows empty RECORD buffer
Bug 28079925	Outbound support <code>XSL:NIL</code>
Bug 27768540	<p>HTTP response code enhancements</p> <p>A new environment variable <code>GWWS_HTTP_RESP_FIELD</code> has been introduced. Following is the format of <code>GWWS_HTTP_RESP_FIELD</code>: <code>GWWS_HTTP_RESP_FIELD=FML32_FIELD_NAME[:keep]</code></p> <p>Example</p> <ul style="list-style-type: none"> • <code>export GWWS_HTTP_RESP_FIELD=Test_Resp_Code</code> • <code>export GWWS_HTTP_RESP_FIELD=Test_Resp_Code:keep</code> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>If the replied error buffer contains a <code>Test_Resp_Code</code> field, then the HTTP status code is applied to the value of <code>Test_Resp_Code</code>. Further, the default behavior removes the <code>Test_Resp_Code</code> from the error buffer and it is not converted to a JSON message. However, if you want it to be a part of JSON data, you must specify <code>:keep</code>. In addition, <code>Test_Resp_Code</code> must be present at the top level of the replied error buffer, else it does not take effect in an embedded FML32 buffer.</p> </div>
Bug 27608108	<p>Inbound service supports the following two attributes:</p> <ul style="list-style-type: none"> • <code>jsonTopLevelArray</code>: If the value of this attribute is set to <code>true</code> and the top level data of the inbound response is an array, then GWWS returns a no-name json array at the top level. The default value is <code>false</code>; Required:No • <code>enableRplyBuffer</code>: If the value of this attribute is set to <code>true</code>, then GWWS converts the TPFALL error buffer to a json message. The default value is <code>false</code>; Required:No

BugDB Number	Description
Bug 25735361	<p>Inbound service supports default content-type</p> <p><code>content-type</code>: A new attribute has been added to the Inbound->HTTP->Service element in the SALT deployment file. It is the default value for the Inbound request <code>content type</code>.</p> <p>If the Inbound request HTTP header contains the <code>content-type</code> field, SALT uses the value in HTTP header. Otherwise, SALT uses the default <code>content-type</code> attribute set in the Inbound->HTTP->Service element.</p>
Bug 25342939	<p>Outbound service supports override URI</p> <p>In order to specify the override URI, the Oracle Tuxedo client can add system FML32 field <code>TA_HTTP_CONNECT_URI</code> to <code>META_TCM</code> of the request data by invoking <code>tpsetcallinfo()</code>.</p>
Bug 24818790	<p>Support whitespace collapse for <code>xsd:string</code></p> <p>A new parameter level keyword <code>whitespace</code> has been added to Oracle Tuxedo Metadata Repository. When converting the Oracle Tuxedo buffer to XML data mapping, if the above keyword is set to <code>collapse</code>, then the white space collapses.</p>
Bug 24395972	<p>Inbound service supports specifying namespace for SOAP responses</p> <p>You can specify <code>namespace</code> at three different levels:</p> <ul style="list-style-type: none"> <p>service level: specifying <code>namespace</code> attribute in WSDL</p> <pre><Service name="operation_1" soapAction="opl" namespace="op_ns" /></pre> <p>out buffer level: <code>outbufschema</code> parameter in the MIF file</p> <pre>XSD_E:<element_local_name>@namespaceURI</pre> <p>parameter level: <code>paramschema</code> parameter in the MIF file</p> <pre>XSD_E:<element_local_name>@namespaceURI</pre> <p>The element <code>namespace</code> uses the value defined in the nearest scope. If none of above is specified, then the element uses <code>urn:pack.<WSDL-name>_typedef.salt11</code> as before.</p> <p>If you specify the service level <code>namespace</code> or <code>outbufschema</code>, then all the elements under <code><servicenameResponse></code> or <code><outbuf></code> inherit it. However if the <code>paramschema</code> is specified for the matched parameter, then the inbound <code>namespace</code> works only for the <code>document/literal</code> mode SOAP operation; it does not work for the <code>rpc/encoded</code> mode.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 20px;"> <p> Note:</p> <p><code>element_local_name</code> in <code>outbufschema</code> and <code>paramschema</code> is not used.</p> </div>

BugDB Number	Description
Bug 23742025	Metadata Repository supports <code>jsonarray</code> for SALT The Metadata Repository supports a new parameter level keyword <code>jsonarray</code> . If this keyword is set to <code>Y</code> , then GWWS maps the Oracle Tuxedo buffer to JSON array type, even if there is only one occurrence of a field.