Oracle® Enterprise Governance, Risk and Compliance

Installation Guide

Release 8.6.6.7000

Part No. E91000-01

October 2017

ORACLE

Oracle Enterprise Governance, Risk and Compliance Installation Guide

Part No. E91000-01

Copyright © 2017 Oracle Corporation and/or its affiliates. All rights reserved.

Primary Author: David Christie

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The 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 software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable.

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The software 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

The software and documentation may provide access to or information on 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. 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.

Contents

1

Introduction

Oracle Enterprise Governance, Risk and Compliance (GRC) is a set of products that regulate activity in business-management applications. This document provides instructions for the installation or upgrade of the following GRC products:

- Oracle Enterprise Governance, Risk and Compliance Manager (EGRCM) forms a documentary record of a company's strategy for addressing risk and complying with regulatory requirements.
- Oracle Advanced Controls enables users to create "models" and "continuous controls." Two Advanced Controls applications run from within the GRC platform:
  - In Oracle Enterprise Transaction Controls Governor (ETCG), models and controls specify circumstances under which individual transactions display evidence of error, fraud, or other risk.
  - In Oracle Application Access Controls Governor (AACG), models and controls define conflicts among duties that can be assigned in a company's applications, and identify users who have access to those conflicting duties.

    AACG can also implement "preventive analysis" — it can evaluate controls as duties are assigned to users of the company's applications, preventing them from gaining risky access.
- Oracle Fusion GRC Intelligence (GRCI) extracts data from GRC for display in dashboards and reports.

You can install GRC on its own, or to be integrated with an OID LDAP server that manages GRC users. (OID stands for Oracle Internet Directory; LDAP for Lightweight Directory Access Protocol.)

You can embed a GRCI instance within GRC. To use GRCI, install GRC first (see chapter 2). Then integrate GRCI with GRC (see chapter 3).

Prerequisites

You can install GRC 8.6.6.7000 only as an upgrade to version 8.6.6.6000. GRC runs on a 64-bit Linux server and requires the following.

- An Oracle 12.1.0.2 (12*c*) database, by preference. During earlier upgrades, you may have retained an Oracle 11.2.0.4 or 11.2.0.3 database. You can continue to use that database with version 8.6.6.7000.

  In the database, a GRC schema must be created. If you implement GRCI, a data analytics (DA) schema must exist as well. The database that supports the GRC schema must use the AL32UTF8 character set.

  If you want to upgrade to the 12*c* database, follow this sequence to install GRC 8.6.6.7000: Upgrade your database to 12*c*, import your GRC schema to the 12*c* database, upgrade GRC, then point your GRC instance to the new database (see "GRC Configuration" beginning on page 8).

- Java: Oracle JDK 1.7. GRC must have its own dedicated Java container. It was not designed to coexist in a container with other web applications.

- Middleware: To support GRC, use WebLogic Server 12*c* (12.1.2 or 12.1.3) or Tomcat Application Server 7.0.47.

  If you use WebLogic Server (WLS), you also need the version of Application Development Runtime (ADR) and Repository Creation Utility (RCU) that matches the number of your WLS version (12.1.2 or 12.1.3). In the 12*c* release, RCU is packaged with ADR.

  If you intend to run GRCI, you also need WLS 11*g* (10.3.6), installed with RCU 11.1.1.7 and ADR 11.1.1.7. This is true even if you use WLS 12*c* to support GRC itself.

As an option, an OID LDAP server can manage GRC users.

On the server or a client system, the following web browsers can display the GRC interface: Microsoft Internet Explorer 11*x* or FireFox 38.5 or 43.04.

For details about supported components, see the *Oracle Enterprise Governance, Risk and Compliance Certifications Document*.

Recording Configuration Values

Make a note of certain configuration values for version 8.6.6.6000, as you will need to re-enter them for version 8.6.6.7000. All these values are displayed in the GRC Manage Application Configurations page. (Start the GRC 8.6.6.6000 instance, then select Navigator → Setup and Administration → Setup → Manage Application Configurations).

- Select a Properties tab and note values you will need to enter in a ConfigUI page during 8.6.6.7000 installation.
- If you have set up GRC to work with an OID LDAP repository, select a User Integration tab and note the values entered there.
- If you use GRCI, select an Analytics tab and note the values entered there.

Downloading Files

Create a staging directory on your GRC server. (Throughout this document, *<grc_stage>* represents the full path to this directory.)

To upgrade GRC, download a file called grc866_7017.zip to <grc_stage>, and extract its contents there. To validate your download, generate a checksum and compare it with a checksum value published in *Release Notes* for the instance you are installing. To generate a checksum, run the command *md5sum grc.ear*.

If you have not embedded GRCI in your 8.6.6.6000 instance, but wish to do so for 8.6.6.7000, you can do so only if you created a Data Analytics (DA) schema for GRC 8.6.6.1000 and upgraded it (reconnected it to GRC) for in each subsequent release. If so, download files called grc865_1_OBIEE_ 1of3.tar.gz, grc865_1_OBIEE_ 2of3.tar.gz, and grc865_1_OBIEE_3of3.tar.gz to <grc_stage>.

For you to embed GRCI in GRC, your instance must run with WebLogic. If you use Tomcat, you can run GRCI only as a standalone application. You do not need the three grc865_1_OBIEE files for standalone GRCI, or if you have already embedded GRCI in your 8.6.6.6000 instance.

2

Upgrading GRC

As you upgrade to GRC 8.6.6.7000:

- You may continue to use the database installed for your 8.6.6.6000 instance or, if that database is 11.2.0.3 or 11.2.0.4, you may upgrade to 12*c*. In the latter case, perform the upgrade and import your GRC and DA schemas to the new database before completing the procedures in this chapter.
- You may reuse WebLogic or Tomcat middleware components already installed for GRC 8.6.6.6000. Or, if you use WebLogic, you may upgrade it from version 12.1.2 to 12.1.3.
- Reuse Java components already installed for GRC 8.6.6.6000.

GRC Repositories

For your earlier version of GRC, you should have created two “repositories” — directories that store data generated by GRC. A report repository stores copies of GRC reports that users schedule to be run. A second repository stores synchronization data used for transaction analysis.

Reuse these repositories for GRC 8.6.6.7000. Retain the contents of the transaction synchronization repository. Note the paths to the repositories, as you will need to supply them later as configuration values.

Upgrading GRC Only

If you intend to upgrade WebLogic as well as GRC, skip ahead to “Upgrading GRC and WebLogic” on page 15. If you intend to reuse your existing middleware components as you upgrade GRC, begin here. The process involves these steps:

1. Ensure that two directories, for the storage of GRC report data and ETL data, are ready for use (see “GRC Repositories,” above).
2. Remove some files installed for your GRC 8.6.6.6000 instance, and run a setup script.
3. Restart the WebLogic or Tomcat application server, then perform configuration steps in a GRC ConfigUI page.

4. Restart your application server to complete the installation.

Back up your database, schema, middleware components, and report and transaction ETL repositories.

GRC with WebLogic

If you installed GRC to run with WebLogic Server, complete the following steps:

1. Stop the WebLogic Administration Server and (if any exist in your installation) managed servers.
2. During installation of earlier GRC releases, a directory called grc866 was created, typically as a subdirectory of your middleware home directory (represented in this document as <MW_HOME>). Delete the contents of this directory.
3. Navigate to <grc_stage>/dist, and locate a grc.ear file.
4. Extract the contents of grc.ear into the grc866 directory.
5. Navigate to <grc_stage>/dist. From there, run the file grc_wls_setup.sh. Supply the path to the grc866 directory (into which you extracted the contents of the grc.ear file in step 4). For example:

   ```
   cmd> ./grc_wls_setup.sh <MW_HOME>/grc866
   ```

6. Remove content from the following directories. (In these paths, <grc_domain> represents the name of the WebLogic domain created for GRC during installation of version 8.6.6.1000, and <managed_server> is the name of a WebLogic managed server, if one was created during installation of version 8.6.6.1000.)

   <MW_HOME>/user_projects/domains/<grc_domain>/servers/AdminServer/logs
   <MW_HOME>/user_projects/domains/<grc_domain>/servers/AdminServer/cache
   <MW_HOME>/user_projects/domains/<grc_domain>/servers/AdminServer/tmp
   <MW_HOME>/user_projects/domains/<grc_domain>/servers/<managed_server>/logs
   <MW_HOME>/user_projects/domains/<grc_domain>/servers/<managed_server>/cache
   <MW_HOME>/user_projects/domains/<grc_domain>/servers/<managed_server>/tmp

7. Copy the files jython-2.5.1.jar and xdoparser-10.1.3.4.jar from <grc_stage>/lib to <MW_HOME>/user_projects/domains/<grc_domain>/lib.
8. Navigate to the file setDomainEnv.sh, and open it in a text editor. The file is located in the <MW_HOME>/user_projects/domains/<grc_domain>/bin directory.
9. In setDomainEnv.sh, locate the following lines:

   ```
   if [ "${PRE_CLASSPATH}" != "" ] ; then
   CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${CLASSPATH}"
   export CLASSPATH fi
   ```

10. Add the following before those lines:

    ```
    PRE_CLASSPATH="<MW_HOME>/user_projects/domains/<grc_domain>/
    lib/jython-2.5.1.jar:${PRE_CLASSPATH}" export PRE_CLASSPATH
    ```

    (Replace <MW_HOME> and <grc_domain> with the specific values appropriate for your environment.)

Note: You may skip steps 7–10 if you have already performed them for a prior release.

11. Restart the WebLogic servers.

GRC with Tomcat

If you installed GRC to run with Tomcat Application Server, complete the following steps:

1. Shut down the Tomcat Application Server.
2. Remove the directory <TomcatHome>/webapps/grc and all its contents.
3. Remove the Catalina directory from the Tomcat work area (<TomcatHome>/work/Catalina). Delete the contents of <TomcatHome>/temp. Also delete Tomcat logs, located at <TomcatHome>/logs. (You may want to save them to another location.)
4. Navigate to <TomcatHome>/webapps and delete the grc.war file.
5. Navigate to <grc_stage>/dist. From there, run the file grc_tomcat_setup.sh. Supply the paths to the <grc_stage>/dist subdirectory, <TomcatHome>, and the full path to your Java home as parameters:

   ```
   cmd> ./grc_tomcat_setup.sh <grc_stage>/dist <TomcatHome>
   <JavaHomePath>
   ```

6. Copy the file xdoparser-10.1.3.4.jar from <grc_stage>/lib to <TomcatHome>/webapps/grc/WEB-INF/lib.
7. Start the Tomcat application server.

GRC Log-On URL

When you installed release 8.6.6.1000 of GRC, you may have set it up to support Secure Sockets Layer (SSL). How you log on to GRC depends on whether it does or does not support SSL.

If you have not set up GRC to use SSL, log on with the following URL:

`http://host:http_port/grc`

Replace *host* with the fully qualified domain name (FQDN) of your GRC server. Select one of the following values for *http_port*:

- If you use WebLogic 12*c*, enter the port number you chose for the Administration Server as you created a WebLogic domain.
- If you use Tomcat, enter 8080 (if you accepted the default value when you installed Tomcat) or your configured value (if you changed the default during Tomcat installation).

If you have set up GRC to use SSL, log on with the following URL:

`https://host:https_port/grc`

Again, replace *host* with the FQDN of your GRC server. Select one of the following values for *https_port*:

- If you use WebLogic 12*c*, enter the port number you selected as the SSL Listen Port when you set up SSL for GRC.

- If you use Tomcat, enter the port number you selected as the Connector Port when you set up SSL for GRC.

However, a third logon URL applies if you set up GRC to recognize Single Sign On (SSO). See “Integrating with Single Sign On Authentication,” beginning on page 12.

GRC Configuration

Regardless of whether you use WebLogic or Tomcat, perform GRC-specific configuration:

1. Open a ConfigUI page: Navigate to your GRC log-on URL. If prompted, supply the *admin* user ID along with a password established for that user ID during installation of version 8.6.6.1000.
2. In the Installation Configuration section, type or select appropriate property values:
   - User Name: Supply the user name for the GRC database.
   - Password: Supply the password for the GRC database.
   - Confirm Password: Re-enter the password for the GRC database.
   - Port Number: Supply the port number at which the GRC database server communicates with other applications.
   - Service Identifier: Supply the service identifier (SID) for the GRC database server, as configured in the tnsnames.ora file.
   - Server Name: Supply the FQDN of the database server.
   - Maximum DB Connections: Default is 50. You can edit this value.
   - Report Repository Path: Supply the full path to the Report Repository directory discussed in “GRC Repositories” on page 5.
   - Log Threshold: Select a value that sets the level of detail in log-file entries. From least to greatest detail, valid entries are *error*, *warn*, *info*, *debug*, and *trace*. Select *trace* only if Oracle Support instructs you to do so.
   - Transaction ETL Path: Enter the full path to the directory you created to hold ETL data used by Enterprise Transaction Controls Governor (see “GRC Repositories” on page 5).
   - App Server Library Path: Enter the full path to the library subdirectory of your web application server (for use in the upload of custom connectors for AACG). If you use Tomcat Application server and intend to enable parallel processing (see step 4 below), set this field to the “lib/adf” subdirectory of the Tomcat home directory.
3. In the Language Preferences section of the ConfigUI page, select check boxes for up to twelve languages in which you want GRC to be able to display information to its users. “English (U.S.)” should be selected by default; do not deselect it.

4. In the Performance Configuration section of the ConfigUI page, select or clear check boxes:

   - Optimize Appliance-Based Operation: Select the check box to optimize performance if the GRC application and GRC schema reside on the same machine. Do not select this check box if the GRC application and schema do not reside on the same machine. When you select this check box, an ORACLE_HOME Path field appears. In it, enter the full, absolute path to your Oracle Home — the directory in which you have installed the Oracle database that houses the GRC schema.

   - Enable Graph Synchronization Date Limit: “Data synchronization” enables GRC to recognize data changes in each business application subject to models and controls. Although the concept applies to AACG and ETCG, the process works differently for the two applications.

     Either application recognizes “business objects,” each of which is a set of related fields from a “data source” (business application). ETCG distinguishes among three categories of business object — Transaction (in which records are created or updated frequently), and Operational and Configuration (consisting of master-data or setup records that change infrequently).

     For ETCG only, select the Enable Graph Synchronization Date Limit check box to cause the synchronization of Transaction business objects to operate only on records created or updated in data sources on or after a specified date.

     The setting of this check box has no effect on ETCG Operational or Configuration business objects, for which a synchronization run encompasses all records, no matter when they were created or updated. Moreover, AACG does not distinguish among business-object categories, and the setting of this check box has no effect on AACG synchronization runs.

     When you select the check box, a Transactions Created As Of field appears. In it, enter the cutoff date for the synchronization of ETCG Transaction business objects. When you click in the field, a pop-up calendar appears. Click left- or right-pointing arrows to select earlier or later months (and years), and then click on a date in a selected month.

   - Externalize Report Engine: Select the check box to enable the reporting engine to run in its own java process, so that the generation of large reports does not affect the performance of other functionality. But select the check box only if you have installed GRC on hardware identified as “certified” in the *Oracle Enterprise Governance, Risk and Compliance Certifications Document*; clear the check box if you use hardware identified as “supported.”

   - Enable Parallel Processing: Select this check box to enable multiple jobs to run simultaneously. When you select the Enable Parallel Processing check box, two fields appear:

     Number of Cores Available for Processing: Enter the number of processor cores you wish to devote to parallel processing. GRC uses one core for each job, until as many cores as you specify here are in use.

     Maximum Megabytes of Physical RAM Available: Specify an amount of memory for use in parallel processing. Ensure that this value is at least 16

GB times the number of cores. GRC then divides the memory value by the core value to determine the actual amount of memory per core.

This value is in addition to what is already allocated to the WebLogic Admin server or Tomcat server. Review the amount of memory allocated to other processes (such as Linux or database management) before allocating memory to parallel processing. **Important:** Allocating more memory than what is available causes disk swapping, which causes poor performance during peak load.

- Enforce Allocated Analysis Time Per Filter: Select this check box, and enter a number in the Minutes field, to limit the time that transaction models and controls can run.

  A model or control consists of filters, each of which defines some aspect of a risk and selects transactions that meet its definition. When the Allocated Analysis Time feature is enabled, each filter runs no longer than the number of minutes you specify. If time expires, the filter passes records it has selected to the next filter for analysis, but ignores records it has not yet examined. So a filter may not capture every record that meets its definition, and the model or control results are labeled "partial" in GRC job-management pages.

  Once enabled here, this feature may be disabled for individual models (and for the controls developed from those models). This feature applies only to transaction models and controls, not to access models and controls, and not to EGRCM objects.

5. In the ConfigUI page, click on Actions → Save. GRC tests the values you've entered and, if they are valid, saves them. (If any are invalid, an error message instructs you to re-enter them.)
6. Exit the ConfigUI page.

Completing the Installation

With components in place and properly configured, complete the installation, in effect by running your web application server.

1. Shut down your server — the Administration Server if you're using WebLogic, or the Tomcat application server if you're using Tomcat. Then restart the server.
2. In a web browser, enter the GRC URL (see "GRC Log-On URL," page 7).
3. Wait for a progress bar to indicate that initialization is complete.
4. You are redirected to a GRC logon page. Log on to the application. Use the *admin* user ID, with a password established for that user ID during the installation of version 8.6.6.1000.

If you have not set up an external OID LDAP repository to manage users, basic GRC installation is complete. (You may, however, choose complete other procedures described later.)

If you have set up an OID LDAP repository, complete these additional steps:

1. In GRC, select Navigator → Setup and Administration → Setup → Manage Application Configurations.

2. To configure external OID LDAP, select the User Integration tab and enter the following values:
   - Enable Single Sign On: See “Integrating with Single Sign On Authentication” on page 12.
   - Enable Integration: Select the check box to permit integration with LDAP to occur.
   - User Name: Supply the user name (common name) to log in to the LDAP server. This user should have admin privileges.
   - Password: Enter the password for the user identified in the User Name field.
   - Confirm Password: Re-enter the password for the user identified in the User Name field.
   - Port: Enter the port number at which the LDAP server communicates with other applications.
   - Server Name: Enter the host name of the LDAP server.
   - Bind DN Suffix: Enter the “User Base DN.”
   - Enable SSL Authentication: Select the box to allow GRC to access the LDAP server through SSL. If you select this option, the LDAP server must be configured to support SSL.
   - Perform LDAP Recursive Search: Select the check box to search recursively for users in subfolders along with those in the base path specified in the Bind DN Suffix field.
   - Unique User Identifier: uid
3. In the Manage Application Configurations page, click on Actions → Save. Then log off of GRC.
4. Stop the GRC Deployment in the WebLogic Console:
   - **a** Log in to the WebLogic Console at

     `http://host:port/console`

     Replace *host* with the FQDN of your GRC server, and *port* with the number you selected for the WebLogic Administration Server as you set up a WebLogic domain.
   - **b** From the Domain Structure menu, select Deployments.
   - **c** From the Deployment page, locate the GRC deployment and verify the state is Active.
   - **d** Click the checkbox next to the GRC deployment.
   - **e** From the toolbar, click Stop → Force Stop Now.
5. Start the GRC Deployment in the WebLogic Console:
   - **a** From the Domain Structure menu, select Deployments.
   - **b** From the Deployment page, locate the GRC deployment and verify the state is Prepared.

c. Click the checkbox next to the GRC deployment.

d. From the toolbar, click Start → Servicing All Requests.

Integrating with Single Sign On Authentication

Rather than use the GRC authentication system to authenticate GRC users, you can integrate GRC with Oracle Access Management (OAM) Single Sign On (SSO). To do so, you must have installed GRC to run with WebLogic; SSO is not supported in a GRC instance that runs with Tomcat. Moreover, you require not only OAM 11*g*R2, but also Oracle HTTP Server (OHS) 11*g* Webgate for OAM.

First, register OHS Webgate 11*g* Agent for OAM 11*g*:

1. Log on to the OAM console. Its URL is `http://<oam_host>:<oam_port>/oamconsole`, in which <*oam_host*> is the host name of the OAM server, and <*oam_port*> is its port number.
2. On the Agents grid of the Application Security tab, select Create Webgate.
3. On the Create Webgate page, enter the following values:
   - Version: Select 11*g*.
   - Name: Enter any value to create a name for the agent.
   - Base URL: Enter `http://<host>:<port>`, in which <*host*> is the host name of the machine where Oracle HTTP Server 11*g* Webgate is installed, and <*port*> is its port number.
   - Security: Select *Open*.
   - Host Identifier: Enter either the Name or the Base URL value.
   - Select the Auto Create Policies check box.
   - In the Protected Resource List, add */grc*.

   Leave the Access Client Password and User Defined Parameters fields blank, and leave the Virtual Host and IP Validation checkboxes unselected.
4. Click the Apply button.
5. The Create Webgate page refreshes. In a new Logout URL field, enter a logout URL, for instance /oamsso/logout.html.
6. Click the Apply button again.

Once the agent is created, update the authentication scheme to your LDAP scheme:

1. Select the Application Security tab.
2. Click Application Domains, then search for and select your agent name.
3. Select the Authentication Policies tab. On it, click Protected Resource Policy.
4. Click the Open icon.
5. Select your LDAP authentication scheme.
6. Click the Apply button.

Next, create a response:

1. Select the Application Security tab.
2. Click Application Domains, then search for and select your agent name.
3. Select the Authorization Policies tab. On it, click Protected Resource Policy.
4. Click the Add icon.
5. Enter these values:
    - Name: OAM_REMOTE_USER
    - Type: Header
    - Value: $user.userid
6. Click Add.
7. Click the Apply button.

Next, copy the Webgate artifacts.

1. Copy ObAccessClient.xml and cwallet.sso
    From: <oam_domain_home>/output/<agent>/
    To: <MW_HOME>Oracle_WT1/instances/instance1/config/OHS/ohs1/webgate/config

Next, modify OHS to redirect to GRC.

1. On the server on which you've installed OHS, navigate to <MW_HOME>/Oracle_WT1/instances/instance1/config/OHS/ohs1.
2. Open the mod_wl_ohs.conf file in a text editor and add the following information to it:

```
<IfModule weblogic_module>
  Debug ON
  WLLogFile /tmp/weblogic.log
</IfModule>
```

```
<Location /grc>
  SetHandler weblogic-handler
  WebLogicHost <GRC_HOST_NAME>
  WebLogicPort <GRC_PORT_NUMBER>
</Location>
```

    Replace <GRC_HOST_NAME> with the FQDN of your GRC server. Replace <GRC_PORT_NUMBER> the number of your HTTP Port if SSL is not enabled, or your HTTPS Port if SSL is enabled. See "GRC Log-On URL" (page 7) for more information on identifying these port numbers.

3. Save and close the mod_wl_ohs.conf file.
4. Restart the OAM server and Webgate.

Next, add the OAM Identity Asserter to the GRC domain:

1. Log in to the WebLogic Server Administration Console:

```
http://host:port/console
```

Replace *host* with the FQDN of your GRC server, and *port* with the number you selected for the WebLogic Administration Server.

2. Click Lock and Edit.
3. Click Security Realms (on the left under Domain Structure), then click myrealm.
4. In the Providers tab, click the New button, and enter *OAM Identity Asserter* for both Name and Type. Then click the OK button.
5. In the Providers tab, click the newly created OAM Identity Asserter.
6. In the Common tab, select:
   - ControlFlag: Required
   - Active Types — Choose: OAM_REMOTE_USER (remove ObSSOCookie from the Chosen box)

   Click the Save option.
7. Return to the Providers tab, click on DefaultAuthenticator, change the ControlFlag to SUFFICIENT, and click the Save option.
8. In the Providers tab, reorder the authentication providers so that OAM Identity Asserter is first, DefaultAuthenticator is second, and DefaultIdentityAsserter is third. Then click the OK button.
9. Click Activate Changes and restart the application server.

Finally, enable SSO in GRC:

1. Log on to GRC and select Navigator → Setup and Administration → Setup → Manage Application Configurations. Select the User Integration tab. (You are presumed to have already set OID LDAP values on this page, as described in “Completing the Installation,” beginning on page 11.)
2. Select the Enable Single Sign On check box.
3. Enter your logout URL in the OAM 11gR2 Logout URL field. (This should be the same value as you entered in the Create Webgate page as you registered an agent for GRC.)
4. Select the Save option from the Actions menu. Verify that you receive a message stating that configuration values were saved successfully.
5. Select Navigator → Setup and Administration → Security → Manage Users. Select the Import from LDAP option from the Actions menu and import users.
6. To enable SSO logout:
   - Log off of GRC and navigate to <MW_HOME>/wlserver/common/bin/.
   - Execute the following command to start wlst.sh:

     ```
     ./wlst.sh
     ```
   - Execute the following command at the wls:/> prompt:

     ```
     addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",
     logouturi="/oamsso/logout.html", autologinuri="/obrar.cgi")
     ```

Once you have configured SSO for GRC, the URL to access GRC is:

```
http://<webgate_host>:<webgate_port>/grc
```

Or, if a secure protocol like SSL or TLS has been enabled in the OAM application server, the leading characters of the URL are https://

```
https://<webgate_host>:<webgate_port>/grc
```

Communicate this URL to all GRC users.

GRC and SSL

When you installed release 8.6.6.1000 of GRC, you may have set it up to support Secure Sockets Layer (SSL). If so, complete the following steps to ensure that GRC communicates properly with preventive enforcement agents (PEAs) installed in Oracle E-Business Suite instances subject to AACG analysis.

1. Copy the file axis2.xml from <grc_stage>/conf to:
    - <MW_HOME>/user_projects/domains/<grc_domain>/grc/WEB-INF/conf, if you installed GRC to run with WebLogic.
    - <TomcatHome>/webapps/grc/WEB-INF/conf, if you installed GRC to run with Tomcat.
2. Copy the file addressing-1.7.3.mar from <grc_stage>/modules to:
    - <MW_HOME>/user_projects/domains/<grc_domain>/grc/WEB-INF/modules, if you installed GRC to run with WebLogic.
    - <TomcatHome>/webapps/grc/WEB-INF/modules, if you installed GRC to run with WebLogic.
3. Open the axis2.xml file in a text editor. In it, search for the following text:

```
<transportReceiver name="http"
class="org.apache.axis2.transport.http.AxisServletListener">
      <parameter name="port"><add HTTP Port></parameter>
</transportReceiver>

<transportReceiver name="https"
class="org.apache.axis2.transport.http.AxisServletListener">
   <parameter name="port"><add HTTPS Port></parameter>
</transportReceiver>
```

4. In that passage:
    - Replace the phrase “add HTTP Port” with the number of the port GRC would use if SSL were not enabled.
    - Replace the phrase “add HTTPS Port” with the number of the port GRC uses once SSL is enabled.

    See “GRC Log-On URL” (page 7) for more information on identifying these port numbers.

Upgrading GRC and WebLogic

If you intend to upgrade WebLogic as well as GRC, the process involves these steps:

1. Ensure that two directories, for the storage of GRC report data and ETL data, are ready for use (see “GRC Repositories,” page 5).

2. Install WebLogic Server, along with Application Development Runtime (which includes Repository Creation Utility).
3. Create a WebLogic domain. This entails setting up an Administration Server.
4. If you intend for an OID LDAP repository to manage GRC users, configure that repository.
5. Modify memory and other settings to conform to GRC requirements, and perform configuration steps in a WebLogic Server Administration Console.
6. Perform configuration steps in a GRC ConfigUI page.
7. Run WebLogic to complete the installation.

Setting Up WebLogic

As you install WebLogic Server (WLS) and related components, you make choices that support their use with GRC.

Initial WebLogic Installation

Ensure that Oracle JDK 1.7 is in the path to install and run WebLogic Server.

Install WLS 12*c* (12.1.3) as a standard default deployment. Also install ADR 12*c*, which includes RCU 12*c*. Consult WebLogic documentation for detailed procedures.

Once ADR 12*c* is installed, run RCU to create WebLogic repositories. Required RCU components include AS Common Schemas and all its subcomponents. As you run RCU, expand AS Common Schemas and verify that all subcomponents are selected.

Creating a WebLogic Domain

Create a new WebLogic domain. To do so, execute the following command to run a Fusion Middleware Configuration Wizard:

```
<MW_HOME>/wlserver/common/bin/config.sh
```

Note: <*MW_HOME*> represents the full path to the home directory of a given middleware installation — the highest-level directory in which Fusion Middleware components exist, including WebLogic. Also, <*grc_domain*> represents the name you give to the domain you create for a given WebLogic instance.

The Wizard prompts you to select a domain location, select templates, create a domain name, create a name and password for an "administration account," and complete other steps. In most cases, you should consult WebLogic documentation to understand your options, and are free to make configuration decisions that suit your preferences. However, note the following:

- Select these four templates:

  Basic WebLogic Server Domain — 12.1.3.0 [wlserver]
  Oracle Enterprise Manager — 12.1.3.0 [em]
  Oracle JRF — 12.1.3.0 [oracle_common]
  WebLogic Coherence Cluster Extension — 12.1.3.0 [wlserver]

- When prompted, select “Production” for your Domain Mode, and ensure that the correct JDK is selected. (This is the JDK instance you confirmed to be in the path to install and run WebLogic under “Initial WebLogic Installation” on page 16.) If necessary, use the “Other JDK Location” option to browse.
- When prompted, choose to create an Administration Server. You need not create any managed servers.

Preparing Additional Files

Complete these additional steps when the config.sh script finishes running:

1. Copy the following files from <grc_stage>/lib to <MW_HOME>/user_projects/domains/<grc_domain>/lib:
   - groovy-all-2.0.5.jar
   - xdoparser-10.1.3.4.jar
   - jython-2.5.1.jar
   - adfdt_common.jar
   - adfm-12.1.3.jar
   - adfmweb.jar
2. Create a directory called grc866 (for example, <MW_HOME>/grc866). This directory should be entirely distinct from the <grc_stage> directory you created as you downloaded GRC files.
3. Navigate to <grc_stage>/dist, and locate the file grc.ear. Extract its contents to the grc866 directory.

In addition, as you complete GRC installation procedures, you will use scripts named startWeblogic.sh and stopWeblogic.sh to start and stop the WebLogic Administration Server. First, edit the stopWeblogic.sh file as follows:

1. Open <MW_HOME>/user_projects/domains/<grc_domain>/bin/stopWeblogic.sh in a text editor.

1. Locate a line in the file similar to the following:

   ```
   echo "shutdown('${SERVER_NAME}','Server',
   ignoreSessions='true')"
   >>"shutdown.py"
   ```

2. Edit this line to include a *force='true'* parameter:

   ```
   echo "shutdown('${SERVER_NAME}','Server', force='true',
   ignoreSessions='true')"
   >>"shutdown.py"
   ```

3. Save and close the file.

Whenever you run the stopWeblogic.sh script, wait 30 seconds for all processes to terminate.

Configuring External OID LDAP

This section applies to you only if you intend to install GRC so that an external OID LDAP repository manages its users. If so, complete the following steps. If not, skip ahead to “Modifying Settings” (page 20).

1. Log in to the WebLogic Server Administration Console:

   `http://`*`host`*`:`*`port`*`/console`

   In this URL, replace *host* with the fully qualified domain name (FQDN) of your GRC server, and *port* with the number you selected for the WebLogic Administration Server as you set up a WebLogic domain.

2. Click on the “Security Realms” link in your application’s Security Settings.
3. Click on the “myrealm” link in the table.
4. Click on the “Providers” tab.
5. Click on the New button and enter the following values:
   - Name: OIDAuthenticator
   - Type: OracleInternetDirectoryAuthenticator
6. Click on the “OIDAuthenticator” link and then click on the “Provider Specific” tab.
7. Supply values for properties in the “Provider Specific” screen. (Italicized entries are literal values, to be entered as they are shown.)
   - Host: The FQDN of the LDAP provider (your OID instance).
   - Port: The port number at which the host communicates with other applications.
   - Principal: The username for the OID administrative user, preceded by *cn=*.
   - Credential: The password configured for the OID administrative user.
   - Confirm Credential: The password configured for the OID administrative user.
   - SSLEnabled: Leave this box unchecked.
   - User Base DN: The LDAP path to the store for user information. For example: cn=FusionUsers,cn=users,dc=us,dc=oracle,dc=com
   - All Users Filter*: (&(cn=*)(objectclass=person))* or *(&(uid=*)(objectclass=person))*, depending on your configuration.
   - User From Name Filter: *(&(cn=%u)(objectclass=person))* or *(&(uid=%u)(objectclass=person))*, depending on your configuration.
   - User Search Scope: *subtree*
   - User Name Attribute: *cn* or *uid*, depending on your configuration.
   - User Object Class: *person*
   - Use Retrieved User Name as Principal: Select this checkbox.
   - Group Base DN: The LDAP path to the store for group (enterprise role) information. For example: cn=FusionGroups,cn=groups,dc=us,dc=oracle, dc=com

- All Groups Filter: *(&(cn=*)(|(objectclass=groupofUniqueNames)(objectclass=orcldynamicgroup)))*
- Group From Name Filter: *(|(&(cn=%g)(objectclass=groupofUniqueNames))(&(cn=%g)(objectclass=orcldynamicgroup)))*
- Group Search Scope: *subtree*
- Group Membership Searching: *unlimited*
- Static Group Name Attribute: *cn*
- Static Group Object Class: *groupofuniquenames*
- Static Member DN Attribute: *uniquemember*
- Static Group DN from Member DN filter: *(&(uniquemember=%M)(objectclass=groupofuniquenames))*
- Dynamic Group Name Attribute: *cn*
- Dynamic Group Object Class: *orcldynamicgroup*
- Dynamic Member URL Attribute: *labeleduri*
- User Dynamic Group DN Attribute: Leave this field blank.
- Connection Pool Size: *6*
- Connect Timeout: *0*
- Connection Retry Limit: *1*
- Parallel Connect Delay: *0*
- Results Time Limit: *0*
- Keep Alive Enabled: Leave this box unchecked.
- Follow Referrals: Select this checkbox.
- Bind Anonymously On Referrals: Leave this box unchecked.
- Propagate Cause For Login Exception: Leave this box unchecked.
- Cache Enabled: Select this checkbox.
- Cache Size: *32*
- Cache TTL: *60*
- GUID Attribute: *orclguid*

8. Save your settings, then click on “Activate Changes” on the left, topmost panel.
9. Click the “OIDAuthenticator” link from the authenticator list, and set the Control Flag to SUFFICIENT.
10. Click the “DefaultAuthenticator” link from the authenticator list, and set the Control Flag to SUFFICIENT.
11. Click the Reorder button. Select “OIDAuthenticator” from the available providers, and move it to the top. To do so, click on the arrow on the right side, then click OK.
12. Click on “Activate Changes” from the Change Center, then log out.

13. Stop the WebLogic Administration Server.
14. Edit boot.properties files. There are two possibilities:
    - GRC and OID LDAP components exist on one instance of WebLogic Server (WLS). If so, a boot.properties file exists for the Administration Server.

      In this case, edit the file to set a *username* value equal to your OID administrative user name — the "Principal" in step 7 of this procedure, without the *cn=* prefix. Set a *password* value equal to that user's password — the "Credentials" value in step 7 of this procedure.
    - GRC and OID LDAP exist on distinct instances of WLS. In this case, two boot.properties files exist, for the GRC Administration Server on the GRC instance of WLS, and for the OID Administration Server on the OID LDAP instance of WLS.

      In this case, edit boot.properties files on the OID LDAP instance of WLS to set the *username* and *password* values equal to those for the OID administrative user (as defined earlier in this step). Edit boot.properties files on the GRC instance of WLS to set the *username* and *password* values to those you created as you set up a WebLogic domain (page 16).

    The boot.properties file for the Administration Server exists at <MW_HOME>/user_projects/domains/<grc_domain>/servers/AdminServer/security/boot.properties
15. Start the Administration Server. Check whether LDAP is configured successfully: Log in to the WebLogic console (see step 1of this procedure), go to Security Realms → myRealm, and click on Users and Groups. You should see your LDAP users and groups.

Modifying Settings

For any installation, modify settings in a file called setDomainEnv.sh, which is located in the <MW_HOME>/user_projects/domains/<grc_domain>/bin directory.

1. Stop the Administration Server.
2. Navigate to setDomainEnv.sh and open it in a text editor.
3. In the file, locate the following lines:

   ```
   # IF USER_MEM_ARGS the environment variable is set, use it
   to override ALL MEM_ARGS values

   if [ "${USER_MEM_ARGS}" != "" ] ; then
   ```

   Insert the following lines between those two lines:

   ```
   case "${SERVER_NAME}" in
   "AdminServer")
     USER_MEM_ARGS="-Xms4g –Xmx16g"
   ;;
   "bi_server1")
     USER_MEM_ARGS="-Xms2g –Xmx8g"
   ;;
   "soa_server1")
     USER_MEM_ARGS="-Xms2g –Xmx4g"
   ```

```
;;
"sample_server1")
  USER_MEM_ARGS="-Xms4g -Xmx16g"
;;
"sample_server2")
  USER_MEM_ARGS="-Xms4g -Xmx16g"
;;
*)
  echo "Unknown Server Detected!!. Memory set as Xms1g
Xmx2g.";
USER_MEM_ARGS="-Xms1g -Xmx2g"
;;
esac
```

```
USER_MEM_ARGS="${USER_MEM_ARGS} -XX:PermSize=256m
-XX:MaxPermSize=512m -XX:ReservedCodeCacheSize=128M
-Djava.awt.headless=true
-Djbo.ampool.maxpoolsize=600000
-Dfile.encoding=UTF-8
-Djavax.xml.bind.context.factory=com.sun.xml.internal.bind.
v2.ContextFactory"
-Doracle.jdbc.autoCommitSpecCompliant=false
```

Replace “placeholder” names (*AdminServer*, *bi_server1*, *soa_server1*, *sample_server1*, and *sample_server2*) with the names of your Administration Server and any managed servers included in your installation.

You may use a maximum memory setting (-Xmx) larger than 16G if your server has enough memory to support the larger value.

4. Locate the following section of the file and ensure that “-da” appears after each of two “${enableHotSwapFlag}” elements:

```
if [ "${debugFlag}" = "true" ] ; then

 JAVA_DEBUG="-Xdebug -Xnoagent
 -Xrunjdwp:transport=dt_socket,address=${DEBUG_PORT},server
 =y,suspend=n -Djava.compiler=NONE"

 export JAVA_DEBUG

 JAVA_OPTIONS="${JAVA_OPTIONS} ${enableHotswapFlag} -da
 -da:com.bea... -da:javelin... -da:weblogic...
 -ea:com.bea.wli... -ea:com.bea.broker...
 -ea:com.bea.sbconsole..."

 export JAVA_OPTIONS

else

 JAVA_OPTIONS="${JAVA_OPTIONS} ${enableHotswapFlag} -da"

 export JAVA_OPTIONS

fi
```

5. Locate the EXTRA_JAVA_PROPERTIES section of the file. In it, remove the following string:

```
-Dorg.apache.commons.logging.Log=org.apache.commons.logging.
impl.Jdk14Logger
```

6. Locate the following lines in the file:

```
if [ "${PRE_CLASSPATH}" != "" ] ; then
CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${CLASSPATH}"
export CLASSPATH
fi
```

Add the following before those lines:

```
PRE_CLASSPATH="<MW_HOME>/user_projects/domains/<grc_domain>
/lib/jython-2.5.1.jar:${PRE_CLASSPATH}"
export PRE_CLASSPATH
```

Replace <MW_HOME> and <grc_domain> with the actual values appropriate for your environment.

7. Save and close the file. Start the Administration Server.

Using the WebLogic Console to Deploy the GRC Application

For any installation, use the WebLogic Server Administration Console to complete additional configuration steps:

1. Log in to the WebLogic Server Administration Console at

   `http://host:port/console`

   Replace *host* with the FQDN of your GRC server, and *port* with the number you selected for the WebLogic Administration Server as you set up a WebLogic domain.

2. In the Change Center pane, click Lock & Edit.
3. In the Domain Structure pane, click on Deployments.
4. In the Summary of Deployments pane, select the Control tab.
5. In the Summary of Deployments pane, click on the Install button.
6. In the Path field of the Install Application Assistant pane, enter the full path to the grc866 directory you created earlier (see step 2 of "Preparing Additional Files" on page 17). Select "grc866 (open directory)" under Current Location.
7. In the Install Application Assistant pane, press next.
8. In the Install Application Assistant pane, choose *Install this deployment as an application* in the "Choose targeting style" section.
9. In the Install Application Assistant pane, press Next.
10. In the Install Application Assistant pane, choose *I will make this deployment accessible from the following location* in the "Source accessibility" section. Accept all other defaults.
11. In the Install Application Assistant pane, press Next.
12. In the Install Application Assistant pane, choose *Yes, take me to the deployment's configuration screen* in the "Additional configuration" section.
13. In the Install Application Assistant pane, press Finish.
14. In the Install Application Assistant pane, press Save, then Activate Changes. On the Deployments screen, the state of the grc866 application will be Prepared.

15. Select the grc866 application. Click Start, select *Servicing all requests*, click *Yes* on Start Application Assistant, and wait until the application status changes to Active.

Installing a Driver for RAC

If Real Application Clusters (RAC) is enabled in your GRC database, you must set up a jdbc-oci driver. (If you do not use RAC, this section does not apply; skip ahead to the next section, “GRC Configuration.”)

1. Shut down your WebLogic administration server.
2. In a web browser, go to http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html. Select the Instant Client link for the platform on which you are installing, then find the Basic download for 11.2.0.3.0.
3. Download and unzip the package into a single directory, such as “instantclient.”
4. Set the library loading path in your environment to this directory before starting the application. On many Linux platforms, LD_ LIBRARY_PATH is the appropriate environment variable.
5. Copy the file ojdbc6.jar from the instant client to grc866/grc/WEB-INF/lib. (You created grc866 as a home directory for your GRC installation in step 2 of “Preparing Additional Files” on page 17.)
6. Restart your web application server.

Continuing the Upgrade

Next, perform the procedures documented in these sections:

- “GRC Configuration,” page 8.
- “Completing the Installation,” page 10.
- If appropriate for your installation, “Integrating with Single Sign On Authentication,” page 12.

GRC and SSL

Once GRC in installed, you can set it up to support Secure Sockets Layer (SSL). GRC support for SSL is optional.

Implementing SSL

To install and configure SSL support for a GRC instance that runs with WebLogic, first create custom certificates, then enable and configure SSL.

To create custom certificates:

1. Navigate to the config directory of your WebLogic domain — <MW_HOME>/user_projects/domains/<grc_domain>/config.
2. Create a self-signed keystore. Run the following command. (Here and throughout this section, replace <Java_Home> with the full path to the highest-level directory in which Java components are installed.)

```
<Java_Home>/bin/keytool -genkey -alias grc -keyalg
RSA -keysize 1024 -dname "CN=KeyMachine,OU=Unit, O=Org,
```

```
L=Locality, ST=StateProvince, C=CountryCode" -keypass
KeyPassword -keystore KeyFileName.jks -storepass StorePassword
```

In this command, replace italicized values as follows (and enter other values as they are shown).

- -alias: Accept *grc*, or enter any other value. (If you choose a value other than *grc*, be sure to use that same value where you need to supply the alias in subsequent commands in this procedure.)
- -dname parameters:

  CN stands for Common Name. Replace *KeyMachine* with the fully qualified domain name of the machine on which the keystore is being generated (the GRC server).

  OU and O: Replace *Unit* with the name of an organizational unit, and *Org* with the name of the parent organization of that unit. You can supply any values you choose.

  L, ST, and C: Replace *Locality* with the name of a city or municipality; *StateProvince* with the name of a state, province, or other political subdivision of a country; and *CountryCode* with a two-letter country code.
- -keypass and -storepass: Replace *KeyPassword* and *StorePassword* with passwords that you create as you run this command. It's recommended that you use the same value for both passwords. (These values, established here, will be used in subsequent commands.)
- -keystore: Replace *KeyFileName* with any name for a keystore file. (The file extension must be .jks, and the name, established here, will be used in subsequent commands.)

3. Self-sign the certificate. Run the following command.

```
<Java_Home>/bin/keytool -selfcert -v -alias grc -keypass
KeyPassword -validity 8000 -keystore
KeyFileName.jks -storepass StorePassword -storetype jks
```

Again, replace italicized values as follows (and enter other values as they are shown).

- -alias: Replace *grc* with the alias you created in step 2 (or use *grc* if you accepted that value in step 2).
- -keypass and -storepass: Replace *KeyPassword* and *StorePassword* with the passwords you created in step 2.
- -keystore: Replace *KeyFileName* with the keystore file name you created in step 2.

4. Export the root certificate. Run the following command:

```
<Java_Home>/bin/keytool -export -v -alias grc -keystore
KeyFileName.jks -storepass StorePassword -file rootCA.der
```

Again, replace italicized values as follows (and enter other values as they are shown).

- -alias: Replace *grc* with the alias you created in step 2 (or use *grc* if you accepted that value in step 2).

- -keystore: Replace *KeyFileName* with the keystore file name you created in step 2.
- -storepass: Replace *StorePassword* with the password you created in step 2.
- -file: Replace *rootCA* with a file name of your choosing. (The file extension must be .der, and the name, established here, will be used in a subsequent command.)

5. Import the root certificate into a trusted keystore. Run the following command:

```
<Java_Home>/bin/keytool -import -v -trustcacerts -alias
grc -keystore trust.jks -storepass StorePassword -file
rootCA.der
```

Again, replace italicized values as follows (and enter other values as they are shown):

- -alias: Replace *grc* with the alias you created in step 2 (or use *grc* if you accepted that value in step 2).
- -keystore: Replace *trust* with a new keystore name. This must not be the same as the -keystore value entered in earlier steps, but otherwise may be any value you wish. (The file extension must be .jks.)
- -storepass: Replace *StorePassword* with the password you created in step 2.
- -file: Replace *rootCA* with the file name you created in step 4. (The file extension must be .der.)

When prompted "Trust this certificate? [no]," enter yes to confirm the key import.

To enable SSL:

1. Log in to the WebLogic Server Administration Console at

   `http://host:port/console`

   Replace *host* with the FQDN of your GRC server, and *port* with the number you selected for the WebLogic Administration Server as you set up a WebLogic domain.

   In the left panel of the Console, expand Environment and select Servers.
2. Click the name of the Administration Server on which GRC is installed.
3. In the Change Center of the Administration Console, click Lock & Edit.
4. Select the Configuration tab and then the General subtab.
5. Select the checkbox next to SSL Listen Port Enabled.
6. Enter a port number in the SSL Listen Port textbox.
7. Select Save. Click Activate Changes in the Change Center of the Administration Console.

Repeat this procedure for any managed servers.

To configure SSL:

1. Still in the Administration Console, expand Environment and select Servers.
2. Click the name of the Administration Server on which GRC is installed.

3. In the Change Center of the Administration Console, click Lock & Edit.
4. Select the Configuration tab and then the Keystores subtab.
5. In the Keystore row, click the Change button. From the drop-down list, select Custom Identity and Custom Trust.
6. In the Custom Identity Keystore field, enter the full path to your keystore file (the .jks file you created at step 2 in the procedure for creating custom certificates, page 23).
7. Enter JKS as the value for Custom Identity Keystore Type.
8. For the Custom Identity Keystore Passphrase, enter the value you chose for the -storepass parameter at step 2 in the procedure for creating custom certificates (page 23).
9. Re-enter the -storepass value in the Confirm Custom Identity Keystore Passphrase field.
10. In the Custom Trust Keystore field, enter the full path to your trusted keystore file (the .jks file you created at step 2 in the procedure for creating custom certificates, page 23).
11. Enter JKS as the value for Custom Trust Keystore Type.
12. For the Custom Trust Keystore Passphrase, enter the value you chose for the -storepass parameter at step 2 in the procedure for creating custom certificates (page 23).
13. Reenter the -storepass value in the Confirm Custom Trust Keystore Passphrase field.
14. Select Save. Click Activate Changes in the Change Center of the Administration Console.
15. In the Change Center of the Administration Console, click Lock & Edit.
16. Click on the SSL subtab, to the right of the Keystore subtab.
17. Ensure that the Identity and Trust Locations value is set to “Keystores.” If not, change it to this value.
18. Set the value of the Private Key Alias to the value you chose for the -alias parameter at step 2 in the procedure for creating custom certificates (page 23).
19. For the Private Key Passphrase, enter the values you chose for the -keypass parameter at step 2 in the procedure for creating custom certificates (page 23).
20. Reenter that -keypass value in the Confirm Private Key Passphrase field.
21. Expand the Advanced options of the SSL subtab by clicking on the Advanced pane title.
22. Select the Use JSSE SSL checkbox.
23. Select Save. Click Activate Changes in the Change Center of the Administration Console.
24. Log out of the WebLogic Administration Console: click on the Log Out link at the top of the console page.

25. Bounce the Administration Server, then ensure there are no SSL-related errors in the server log.
26. Navigate to https://*host:SSLport*/console to test your latest changes. (The SSLport is the value you selected in step 6 of the procedure for enabling SSL, page 25.)

Repeat this procedure for any managed servers.

PEA Support

To ensure that GRC communicates properly with preventive enforcement agents installed in Oracle E-Business Suite instances subject to AACG analysis, complete steps documented in the separate "GRC and SSL" section on page 15.

Accessing GRC

After configuring SSL successfully, access the GRC application at https://*host*:*SSLport*/grc, in which *host* is the FQDN of your GRC server, and *SSLport* is the port you selected to support SSL.

Because you are using a self-signed certificate, which is not signed by an official Certificate Authority, you get a security warning when you open GRC at this URL.

- For Internet Explorer, the warning reads, "There is a problem with this website's security certificate." Dismiss this warning by choosing "Continue to this website (not recommended)."
- For Firefox, the warning reads, "This Connection is Untrusted." Dismiss this warning by clicking "I Understand the Risks," and then "Add Exception."

3

Integrating GRCI

Oracle Fusion GRC Intelligence (GRCI) makes use of Oracle Business Intelligence Enterprise Edition (OBIEE) and a Data Analytics (DA) schema. (The database that supports the DA schema should have an initial temporary tablespace of 100 GB with autoextend enabled.)

You can run GRCI only if the DA schema was created for a release of GRC that could be installed independently of earlier releases — in this case, 8.6.6.1000 — then reconnected to GRC for each subsequent upgrade-only release of GRC.

GRCI components have not changed for this release. If you are updating an instance of GRCI already present in GRC 8.6.6.6000, you need only reconnect your DA schema (as described in the following section).

You may install a “fresh” instance of GRCI only if a DA schema was created for release 8.6.6.1000 and reconnected for each subsequent release. If so, obtain version 8.6.6.1000 of the *GRC Installation Guide* and follow its instructions for installing OBIEE and supporting middleware components.

Connecting to the DA Schema

The GRC schema used by GRC supplies data to the DA schema used by GRCI. For this to happen, you need to enter connectivity information in GRC.

1. Log on to GRC (see “GRC Log-On URL,” page 7). Select Navigator → Tools → Setup and Administration → Setup → Manage Application Configurations → Analytics.
2. In the Data Analytics Configuration section, enter values that identify the DA schema. (These are values that you noted earlier. See “Recording Configuration Values” on page 2.)
   - User Name: Supply the user name for the DA database.
   - Password: Supply the password for the DA database.
   - Confirm Password: Re-enter the password for the DA database.
   - Port Number: Supply the port number at which the database server communicates with other applications.

- Service Identifier: Supply the service identifier (SID) for the database server.
- Server Name: Supply the fully qualified domain name of the database server.

3. When you finish entering property values, click on Actions → Save. GRC tests the values you've entered and, if they are valid, saves them. (If any are invalid, an error message instructs you to re-enter them.)
4. Look for the prompt, "Successfully saved configuration values."

After that message appears, a one-time process runs in the background. It creates the DA schema tables and views. This process takes approximately fifteen minutes. Do not stop your WebLogic or Tomcat server during this period.

Once you have connected to the DA schema, set a schedule on which the schema is refreshed — on which the DA schema reads from the GRC schema. You can modify a schedule at any time. (A refresh can take up to 90 minutes to finish.)

1. Select the Analytics tab of the Manage Applications Configurations page.
2. Click on the Schedule Data Analytics Update button.
3. A Schedule Parameter dialog opens. Enter values that set the name of the schedule, its start date and time, the regularity with which the DA schema should be refreshed, and an end date (if any). Then click on the Schedule button.
4. Click on Actions → Save.

To view the status of a scheduled refresh, go to Tools → Setup and Administration → Manage Jobs. To view the Data Analytics schedule, go to Tools → Setup and Administration → Manage Scheduling.

4

Additional Advanced Controls Configuration

Once you've upgraded to GRC 8.6.6.7000, complete additional configuration procedures as needed if you intend to use AACG or ETCG:

- Define information with which GRC creates "global users." Business applications subject to models and controls may have user-account information that varies from one application to the next. GRC maps each person's business-application IDs to a global-user ID. You can determine what information GRC uses to do so.

  If you are upgrading, version 8.6.6.7000 inherits the global-user definition from your earlier version. If you are satisfied with your configuration for the earlier version, you need not redefine it for version 8.6.6.7000.
- Decide whether to implement a Page Access Configurations business object, which enables AACG users to build models and controls that take PeopleSoft user preferences into account. This feature is enabled by default. If your access models and controls do not cite PeopleSoft user preferences, you can disable this feature to improve performance and reduce memory requirements.
- Set up data sources — connections to business applications in which GRC is to perform analysis. However, if you are upgrading, version 8.6.6.7000 inherits data sources configured for your earlier version. For version 8.6.6.7000, you need to set up only new data sources.
- Complete the following procedures in the order indicated:
  1. Perform access synchronization on all data sources used for AACG analysis (see "How to Synchronize Data," page 37).
  2. Perform a graph rebuild on all data sources used for ETCG analysis (again, see "How to Synchronize Data" on page 37).
  3. Run all controls that compile data for user defined objects (controls for which the result type is "Dataset").
  4. Run all models and all controls that generate incidents (controls for which the result type is "Incidents").

Note, however, that if you are upgrading through several releases, then synchronize access data, rebuild the graph for transaction data, and run controls and models only once, after the final upgrade is complete. For information on running models and controls, and distinguishing between control types, see the user guides for AACG and ETCG.

Configuring Global Users

Implement one of the following options to determine the information GRC uses to create global users. **Important:** Select an option that identifies each person uniquely.

- EMAIL_ONLY: Match the global user to email addresses from distinct data sources (or within one data source). This is the default.
- EMAIL_AND_USERNAME: Match the global user to email address plus username from distinct data sources (or within one data source). Because PeopleSoft implementations often do not use the email address for users, customers who implement PeopleSoft usually select this option as well.
- EMAIL_AND_ALL_NAMES: Match the global user to email address, username, given name, and surname from distinct data sources (or within one data source).

GRC users regularly synchronize data and analyze controls to produce "incidents" (records of control violations). If no data has been synchronized and no controls have been analyzed (in version 8.6.6.7000 or any earlier version), complete the following three steps to change a global-user configuration.

1. Use SQL*Plus, or any other tool with the ability to execute SQL commands on a database, to connect to the GRC schema.
2. Run the following SQL statement:

```
DELETE FROM GRC_PROPERTIES
WHERE NAME like 'GLOBAL_USER_CONFIG';
COMMIT;
```

3. Run *one* of the following SQL statements, depending on the global-user format you want to implement:

For email and username, run the following statement:

```
Insert into GRC_PROPERTIES (NAME, VALUE, DESCRIPTION, DEFAULT_VALUE,
VISIBLE, CONFIGURABLE, DATA_TYPE_ID)Values ('GLOBAL_USER_CONFIG',
'EMAIL_AND_USERNAME', 'Global User configuration. Possible values:
EMAIL_ONLY, EMAIL_AND_USERNAME, EMAIL_AND_ALL_NAMES', 'EMAIL_ONLY',
0, 0, 0);

COMMIT;
```

For email, username, given name, and surname, run the following statement:

```
Insert into GRC_PROPERTIES (NAME, VALUE, DESCRIPTION, DEFAULT_VALUE,
VISIBLE, CONFIGURABLE, DATA_TYPE_ID) Values ('GLOBAL_USER_CONFIG',
'EMAIL_AND_ALL_NAMES', 'Global User configuration. Possible values:
EMAIL_ONLY, EMAIL_AND_USERNAME, EMAIL_AND_ALL_NAMES', 'EMAIL_ONLY',
0, 0, 0);

COMMIT;
```

For email only, run the following statement. (As already noted, email-only is the default configuration. Run this statement only if you have changed your global-user configuration to one of the other formats, and want to change back.)

```
Insert into GRC_PROPERTIES (NAME, VALUE, DESCRIPTION, DEFAULT_VALUE,
VISIBLE, CONFIGURABLE, DATA_TYPE_ID) Values ('GLOBAL_USER_CONFIG',
```

```
'EMAIL_ONLY', 'Global User configuration. Possible values: EMAIL_ONLY,
EMAIL_AND_USERNAME, EMAIL_AND_ALL_NAMES', 'EMAIL_ONLY', 0, 0, 0);

COMMIT;
```

A second possibility is that data has been synchronized, but controls have not been analyzed. If so, changing your global-user configuration wipes out all existing global-user data.

1. Complete the steps outlined above for the first global-user-configuration scenario (in which data synchronization and control analysis have not occurred).

2. Sill logged on to your SQL tool, also run the following SQL statement:

```
TRUNCATE TABLE GRC_SRC_USER_MAPPING;
TRUNCATE TABLE GRC_GLOBAL USER;
COMMIT;
```

A third possibility is that data has been synchronized, controls have been analyzed, and incidents have been generated. In this case, when you change your global-user configuration, all existing incidents become invalid, and all existing global-user data is wiped out.

1. Log on to GRC (see "GRC Log-On URL," page 7).

2. From the Navigator, select Tools → Setup and Administration → Setup → Manage Application Configurations. Select the Maintenance tab, and from the Maintenance page, purge *all* existing incidents. (For detailed instructions on purging incidents, see the *Governance, Risk and Compliance User Guide*.)

3. Still logged on to GRC, select Navigator → Continuous Control Management → Results Management → Manage Incident Results. In a Manage Results page, select Incident Result in the View By list box. Confirm that no incidents exist.

4. Log off of GRC and shut down the application server.

5. Complete the steps outlined above for the first global-user-configuration scenario (in which data synchronization and control analysis have not occurred).

6. While logged on to your SQL tool, also run the following SQL statement:

```
TRUNCATE TABLE GRC_SUM_CTRL_INC;
TRUNCATE TABLE GRC_SRC_USER_MAPPING;
TRUNCATE TABLE GRC_GLOBAL USER;
COMMIT;
```

7. Clear the contents of your Transaction ETL Path folder. (This folder is specified as GRC properties are set. See "GRC Configuration," page 8.)

Enabling or Disabling Page Access Configurations

An access model or control may include filters that serve as conditions — they specify users or other objects that are exempt from analysis. Like any other access filter, a condition filter specifies a business object — a set of related fields from a data source (business application). A business object called Page Access Configurations makes PeopleSoft user-preference values available for use in condition filters. By default, processing of data provided by this business object is enabled.

If your site does not use PeopleSoft user-preference values in access models and controls, you may choose to disable the processing of Page Access Configurations data. This improves performance and reduces memory requirements.

Important Note: If you disable Page Access Configurations data processing, the business object will nevertheless appear to be available for use in models. Users may create filters that cite this object, but GRC will ignore those filters. This may cause models (and controls developed from those models) to return results that differ from those that users expect. If you disable Page Access Configurations data processing, alert users not to use the Page Access Configurations business object as they create models.

To disable Page Access Configurations data processing:

1. Shut down the GRC application server.
2. Use SQL*Plus, or any other tool with the ability to execute SQL commands on a database, to connect to the GRC schema.
3. Run the following SQL statement

   ```
   update GRC_PROPERTIES set VALUE = 'FALSE' where NAME =
   'grc.access.user.preferences';
   COMMIT;
   ```

4. Restart the GRC application server.

Configuring Data Sources and Synchronizing Data

Connect GRC to data sources (instances of business-management applications that are to be subject to GRC models or controls). Also synchronize data for each data source — collect information required for AACG or ETCG analysis.

Synchronization and Global Users

The order in which you synchronize access data from data sources determines how GRC creates global-user IDs: It adopts the ID configured for each user in the first data source to be synchronized. When data from a second data source is synchronized, GRC matches users who also exist in the first data source to their already-existing global-user IDs. For each user who did not exist in the first data source, GRC adopts the user ID from the second data source as the user's global ID. And so on.

AACG pages display the global-user ID for each business-application user. You may prefer to set IDs from a particular data source as the global-user IDs.

However, during an upgrade, GRC inherits the global-user IDs existing on the earlier version. For version 8.6.6.7000, global-user IDs are initially the same as they were for version 8.6.6.6000.

If you modify the global-user configuration (see page 32), existing global-user IDs are wiped out. In that case, or as you add new data sources, consider the following:

Configure all data sources in which you expect to apply AACG models and controls before you synchronize data for any of them. Next, choose a data source from which you want GRC to adopt IDs as global-user IDs, and synchronize that data source first. Establish an order for the remaining data sources, each of which sets global IDs for users who do not exist in the data sources for which synchronization has already been completed. Then synchronize the remaining data sources in that order.

To configure data sources or to synchronize their data, log on to GRC (see "GRC Log-On URL," page 7). Select Setup and Administration under Tools in the Navigator, then Manage Application Datasources under Setup.

Special Cases Involving SQL Server

You must install the Microsoft JDBC Driver 4.0 for SQL Server if your GRC instance connects to a Microsoft SQL Server data source and if either of the following is true:

- Your GRC instance runs with Tomcat Application Server.
- Your GRC instance runs with WebLogic and implements Secure Sockets Layer.

Install the driver before you synchronize data for the SQL Server data source. However, if you are upgrading and have already competed this procedure for your earlier GRC version, you need not reinstall the driver.

On the GRC server:

1. Download the UNIX version of the JDBC driver — sqljdbc_*.tar.gz — from http://msdn.microsoft.com/en-us/data/aa937724.aspx.
2. Shut down your application server.
3. From the download file, extract the JDBC driver for SQL Server 2005 and newer — sqljdbc4.jar. (A SQL Server 2000 driver is also included in the download file, but is not supported by GRC.)
4. Copy the sqljdbc4.jar file to a directory appropriate for your application server:
   - If you use Tomcat, the directory is <TomcatHome>/webapps/grc/WEB-INF/lib.
   - If you use WebLogic, the directory is <MW_HOME>/user_projects/domains/<grc_domain>/lib
5. If you use WebLogic, edit the setDomainEnv.sh file. (Skip this step if you use Tomcat.)

   The setDomain Env.sh file is located in the <MW_HOME>/user_projects/domains/<grc_domain>/bin directory. In it, locate the following line:

   ```
   if [ "${PRE_CLASSPATH}" != "" ] ; then
   ```

   Immediately before that line, add the following line:

   ```
   PRE_CLASSPATH="<MW_HOME>/user_projects/domains/<grc_domain>
   /lib/sqljdbc4.jar:${PRE_CLASSPATH}" export PRE_CLASSPATH
   ```

   Replace <MW_HOME> and <grc_domain> with the specific values appropriate for your environment.
6. Restart your application server.

How to Configure Data Sources

To configure a data source, complete these steps. However, remember that GRC version 8.6.6.7000 inherits data sources configured for your earlier GRC version, and you need not reconfigure them.

1. In the GRC Manage Application Datasources page click on Actions → Create New. A Create Datasource window opens. Enter the following values:
   - Datasource Name: Create a name for the data source.
   - Description: Type a brief description of the data source (optional).
   - Application Type: Select the type of business application to which you are connecting, such as EBS or PeopleSoft.
   - Application Type Version: Select the version number of the business-management application to which you are connecting.
   - Default Datasource: Select the checkbox to make the data source you are configuring the default for use in transaction models. Only one data source can have this value selected.
   - Connector Type: For an Oracle EBS or PeopleSoft data source, select Default. For any other application, you would need to have created and uploaded a custom connector; select it.
   - Connector Properties: Enter values required for the connector you specified in Connector Type. Values vary by connector. They may include:
     - ERP Database Type: Select the type of database — Oracle, Oracle RAC, MS SQL Server, DB2, or MySQL — used by the business-management application being configured as a data source.
     - Hostname: For Oracle EBS or PeopleSoft, supply the FQDN for the machine that hosts the database used by the business-management application. Or, if the database is RAC-enabled, enter RAC@<SCAN_NAME>, where <SCAN_NAME> is the IP address/host name configured for the RAC database.
     - Service Name: For Oracle EBS or PeopleSoft, supply the SID value configured for the business-application database in the tnsnames.ora file. Or, if the database is RAC-enabled, enter the RAC service name configured for the RAC database.
     - Port: For Oracle EBS or PeopleSoft, enter the port number that the business-application database uses to communicate with other applications.
     - Username: For Oracle EBS or PeopleSoft, supply the user name for the business-application database. (For an Oracle database, this is the same as Schema Name; for an Oracle EBS instance, this is typically APPS.)
     - Password: Supply the password that authenticates the user name for the business-application database.
2. After entering values, click on the Test Connection button.
3. When the test completes successfully, click the Save or Save and Close button. A row representing the data source appears in the Manage Application Datasources grid.

How to Synchronize Data

You must synchronize data from every data source used for access analysis, and "rebuild the graph" for every data source used for transaction analysis — even data sources inherited from your earlier GRC version.

An ordinary synchronization run creates or updates only records that are new or have changed since the previous synchronization. A graph rebuild deletes all data for a given data source and replaces it with a complete set of current data. This typically takes longer than ordinary synchronization.

To synchronize access data, complete these steps:

1. In the Manage Application Datasources page, select the row for the data source with which you want to synchronize data.
2. Click on Actions → Synchronize Access.
3. A confirmation message appears; click its OK button.

To rebuild the graph for transaction data, complete these steps:

1. In the Manage Application Datasources page, select the row for a transaction data source.
2. Select Actions → Rebuild Graph.
3. A confirmation message appears; click its OK button.

There is an option to synchronize transaction data; it's the preferred option for routine use of ETCG. However, do not use it in this instance. During a GRC upgrade, you must perform a graph rebuild on transaction data sources.

Each time a data source is synchronized, GRC updates fields in the row for that data source: Last Access Synchronization Date and Last Access Synchronization Status show the date of the most recent access synchronization, and its completion status. Last Transaction Synchronization Date and Last Transaction Synchronization Status do the same for the most recent transaction synchronization or graph rebuild.

Determining Data Source IDs

When you configure a data source, GRC assigns an ID number to it. If you intend to implement preventive analysis for an Oracle EBS datasource, you need to know its data source ID. To determine the number, configure the data source, then complete the following steps:

1. In the Manage Application Datasources page, select View → Columns.
2. A list of available columns appears. In it, select Datasource ID.
3. The Manage Application Datasources page now displays a Datasource ID column. In it, note the ID number assigned to the data source you've configured.

If, having determined the data source IDs for your data sources, you wish to remove the Datasource ID column from view, repeat this procedure but clear the Datasource ID selection.

5

Installing PEAs

In support of the AACG preventive analysis feature, ensure that a Preventive Enforcement Agent (PEA) is installed on each Oracle EBS instance that is to be subject to AACG analysis.

If a PEA for the prior release of GRC exists on an EBS instance, you need not reinstall the PEA on that instance. However, if you want to set up an EBS instance as an access data source, and that instance does not have a PEA, you need to install one.

- See the *Oracle Enterprise Governance, Risk and Compliance Certifications Document* for supported versions of Oracle EBS.
- See the *GRC Installation Guide* for version 8.6.6.6000 for the procedure to install an Oracle EBS PEA.