Oracle® GoldenGate

Administering Oracle GoldenGate Monitor

12*c* (12.1.3)

E48290-03

July 2014

This guide explains how to install, configure, and run Oracle GoldenGate Monitor to monitor the status of Oracle GoldenGate processes. It includes information on setting up monitoring with the Oracle Enterprise Manager Oracle GoldenGate Plug-in.

Oracle GoldenGate Administering Oracle GoldenGate Monitor 12*c* (12.1.3)

E48290-03

Primary Author: Helen Haldeman

Contributing Author: Edwin Spear

Contributor: Joseph DeBuzna, Sreenath Sreekantham

Contents

1 Introduction to Oracle GoldenGate Monitoring

2 Using Oracle GoldenGate Monitor Server

3 Using the Oracle GoldenGate Monitor Agent

4 Configuring and Using Alerts

5 Using SSL Communication

6 Understanding Instance Level Security

7 Commands and Parameters

8 Properties

Preface

This preface contains information about and conventions for Oracle GoldenGate Monitor Administrator's Guide.

Audience

This document is intended for installers and system administrators who are installing, configuring, and running Oracle GoldenGate Monitor. It also provides information for installers and system administrators responsible for monitoring Oracle GoldenGate instances using Oracle Enterprise Manager .

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 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.

Related Documents

For more information, see the following documents in the Oracle GoldenGate Monitor documentation set:

- *Using Oracle GoldenGate Monitor*
- *Installing and Configuring Oracle GoldenGate Monitor Server*
- *Upgrading to Oracle GoldenGate Monitor Server 12.1.3*
- *Installing and Configuring Oracle GoldenGate Monitor Agent*
- *Release Notes for Oracle GoldenGate Monitor*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select **Save**." Boldface also is used for terms defined in text or in the glossary.
`italic`, *italic*	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: `TABLE` *`table_name`*. Italic type also is used for book titles and emphasis.
`MONOSPACE, monospace`	Monospace type indicates code components such as user exits and scripts; the names of files and database objects; URL paths; and input and output text that appears on the screen. Uppercase monospace type is generally used to represent the names of Oracle GoldenGate parameters, commands, and user-configurable functions, as well as SQL commands and keywords.
UPPERCASE	Uppercase in the regular text font indicates the name of a utility unless the name is intended to be a specific case.
{ }	Braces within syntax enclose a set of options that are separated by pipe symbols, one of which must be selected, for example: `{`*`option1`* | *`option2`* | *`option3`*`}`.
[ ]	Brackets within syntax indicate an optional element. For example in this syntax, the `SAVE` clause is optional: `CLEANUP REPLICAT` *`group_name`* `[, SAVE` *`count`*`]`. Multiple options within an optional element are separated by a pipe symbol, for example: `[`*`option1`* | *`option2`*`]`.

What's New?

The following topics describe new features introduced in Oracle GoldenGate Monitor 12*c* (12.1.3) and other significant changes in the product, and provides pointers to additional information. This document is the new edition of the formerly titled *Oracle® GoldenGate Monitor Administrator's Guide*.

The new features in Oracle GoldenGate Monitor 12*c* (12.1.3) are:

- New Console Icons Can Start, Stop, and Kill Oracle GoldenGate Monitor Processes
- Configuration Files are Editable
- Information Files Displayed in the Oracle GoldenGate Monitor Console
- JAgent Replaced by Oracle GoldenGate agent; New Functionality Added
- Partial Solutions Now Displayed
- Security Can Be Granted at Instance Level
- A New Class Of "Delta" Metrics Has Been Added
- Globalized Object Names Now Supported In Parameter Files
- z/OS Support Added
- SSL Authentication Now Supported
- Java 1.7 Support (JRE) RequiredInclude Files Can be Displayed Using Hyperlinks
- Include Files Can be Displayed Using Hyperlinks
- All Attributes Graphed

New Console Icons Can Start, Stop, and Kill Oracle GoldenGate Monitor Processes

You can now start, stop or kill a Replicat or Extract process from the Diagram Area of the Data and Alerts View tab, the Log tab, the Configuration tab, and the Problem Summary tab on the Oracle GoldenGate Monitor console by clicking one of the following icons:

Icon	Purpose
▶	**Start**
■	**Stop**

Icon	Purpose
☒	**Kill**

Configuration Files are Editable

Configuration files (`*.prm`) are now editable. When you are on the Configuration tab with an instance selected, the configuration file for that instance appears in the lower panel on the tab. Clicking **Edit** in the bottom right corner of the panel converts it to a text editor where you can modify the configuration as you deem necessary.

Information Files Displayed in the Oracle GoldenGate Monitor Console

You can now see Discard and Report files and the GSSERR.log messages in a panel on the lower half of the Oracle GoldenGate Monitor console.

Discard and Report Files

Discard and report files are now displayed for selected instances at the bottom of the Logs tab. Separate tabs identify the report type. These tabs are divided into two sections, a list of reports, on the left and the report content on the right.

- The Discard tab contains the Discard file. This file contains information about data that failed the respective Extract or Replicat session.
- The Report tab shows run-time reports generated for a specific Extract or Replicat instance. These reports contain the following information:
  - Version information and select environmental settings
  - Runtime parameters
  - Runtime Messages and statistics

GGSERR.log Messages Rendered in the Logs Tab

The Oracle GoldenGate error log file is now displayed in the bottom half of the Logs tab. The messages are arrayed in descending order, with the most recent one at the top. Each line of the error log contains time stamp, severity, error code, process reporting the message, and the actual message. This enables you to track an error or warning to data on the associated report. Additionally, at the bottom of the report is the link Download ggserr.log. Clicking this link will download the `ggserr.log` file so you can view it.

JAgent Replaced by Oracle GoldenGate agent; New Functionality Added

JAgent has been replaced by Oracle GoldenGate agent. The agent is now a separate download and requires a separate installation process (described in *Installing and Configuring Oracle GoldenGate Monitor Agent*). You must install Oracle GoldenGate agent 12*c* (12.1.3) to ensure full functionality and take advantage of all command and control features (for example, edit, stop, and display logs). Additionally, in version 12c (12.1.3), the agent will recognize a new process or that an existing process has been deleted without requiring restart. For more information, see Chapter 3, "Using the Oracle GoldenGate Monitor Agent".

Partial Solutions Now Displayed

While in earlier versions of Oracle GoldenGate Monitor, the diagram view of a process would only show complete solutions (that is, process that begin and end at a database), in version 12*c* (12.1.3), the diagram view of a solution will now display any process that is strung together, even if it does not begin and end at a database (referred to as a *partial solution*).

Security Can Be Granted at Instance Level

Instance Level Security—that is, granting a user access to selected instances—is now featured in Oracle GoldenGate Monitor 12*c* (12.1.3) In addition to the current functional level of security the Instance level security restricts access for different users to different hosts/instances. For more information, see:.

- Chapter 6, "Understanding Instance Level Security"
- "Assigning Instances to a User" in *Using Oracle GoldenGate Monitor.*

A New Class Of "Delta" Metrics Has Been Added

Delta metrics are an extension to all existing *total count* metrics and track the change between samples. For example, if the total number of operations changes from 100 to 5000 between sample periods, the Delta Operations metric would be 4900 during the second sample period. If no change occurred in the third sample period, Delta Operations would become zero.

Globalized Object Names Now Supported In Parameter Files

The parameter files will support different character encoding, along with UTF-8 character set.

z/OS Support Added

Oracle GoldenGate Monitor 12*c* (12.1.3) now supports IBM's z/OS operating system. For more information, see *Installing and Configuring Oracle GoldenGate for DB2 z/OS*.

SSL Authentication Now Supported

Oracle GoldenGate Monitor 12*c* (12.1.3) now supports Secure Sockets Layer authentication. For more information, see Chapter 5, "Using SSL Communication".

Java 1.7 Support (JRE) Required

Oracle GoldenGate Monitor 12*c* (12.1.3) requires the Java 1.7 JRE. For more information, see "Install JDK 1.7 on the Target Machine" in *Installing and Configuring Oracle GoldenGate Monitor Server*.

Include Files Can be Displayed Using Hyperlinks

At the bottom of the console's Configuration table, you can see the content of the respective instance's configuration (`.prm`) file. If a configuration file contains one or more include files:

```
extract ext1
userid oggsrc,password oggsrc
discardfile ./dirdat/ext1.dsc,purge
exttrail ./dirdat/e1
table oggsrc.tcustmer;
include dirprm/my.inc
```

The filename(s) are hyperlinks. You can now click the filename to display the file. The file will appear in a separate tab labeled with the include file's name, in the configuration file area.

All Attributes Graphed

All numeric attributes collected for an instance can now be displayed in a graph view of the data.

1 Introduction to Oracle GoldenGate Monitoring

This chapter identifies the components that make up Oracle GoldenGate Monitor and explains their roles. It provides an overview of the monitoring process and how it discovers the targets that it monitors.

This chapter includes the following sections:

- Section 1.1, "Overview"
- Section 1.2, "Oracle GoldenGate Monitor Architecture"
- Section 1.3, "Understanding the Discovery Process"
- Section 1.4, "12c (12.1.3) Prerequisites"
- Section 1.5, "Platform Support"

1.1 Overview

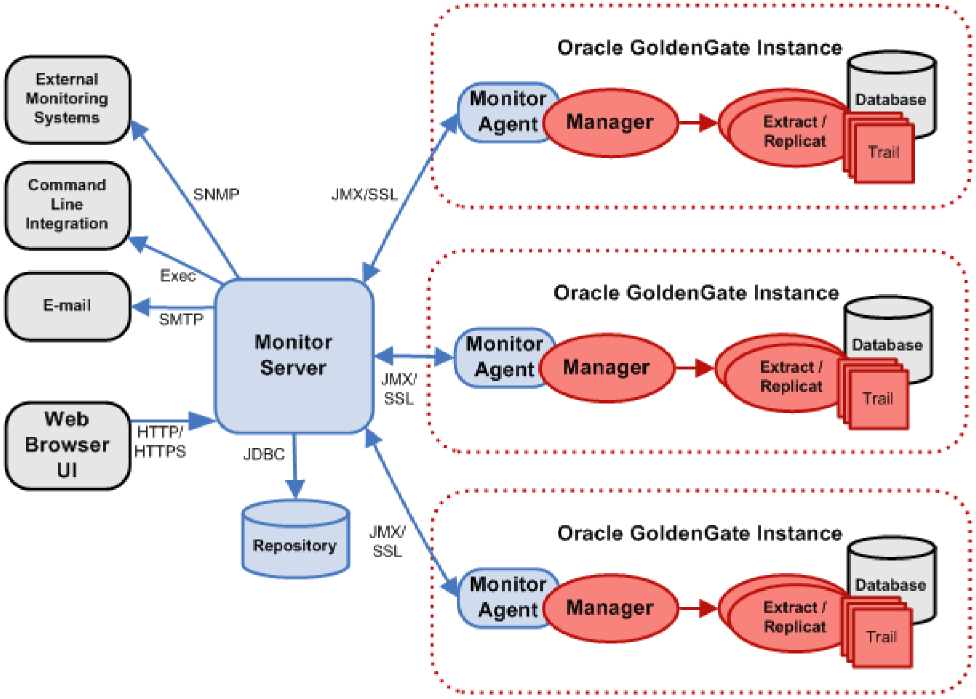
Oracle GoldenGate Monitor is a real-time, Web-based monitoring console that delivers an at-a-glance, graphical view of all of the Oracle GoldenGate instances and their associated databases within your enterprise. Instantly, you can view statistics, targeted views, and alerts that will help you to monitor the performance of all of the objects in the Oracle GoldenGate configuration and detect problems, such as lag or abended processes, the moment that they occur. Oracle GoldenGate Monitor can send alert messages to its own console workspaces, as well as to e-mail, SNMP, and CLI clients.

Oracle GoldenGate instances can be configured for monitoring by a remote client. When monitoring is enabled, Extract, Replicat, and Manager processes supply periodic updates of monitoring points such as status, lag, and checkpoints. The Manager sends these monitoring points to the , a Java agent that communicates with the client.

> **Note:** This documentation supports Oracle GoldenGate Monitor 12*c* (12.1.3).
>
> Oracle GoldenGate releases 11.2.1 and later also support monitoring with Oracle Enterprise Manager. Refer to the *Oracle Enterprise Manager System Monitoring Plug-in Installation Guide for Oracle GoldenGate* for information on this option.

1.2 Oracle GoldenGate Monitor Architecture

Oracle GoldenGate Monitor uses a browser-based graphical user interface to monitor Oracle GoldenGate instances remotely. It includes the components shown in the diagram.



1.2.1 Oracle GoldenGate

An Oracle GoldenGate Monitor Server communicates with one or more Oracle GoldenGate instances through either Java Management Extensions (JMX) or the Secure Sockets Layer (SSL), but not both. The Manager process for each Oracle GoldenGate instance is associated with an Oracle GoldenGate agent that supplies information about the instance to the Oracle GoldenGate Monitor Server.

1.2.2 Oracle GoldenGate Monitor Agent

An Oracle GoldenGate agent should be used with each Oracle GoldenGate instance. It collects information about the instance and sends it to the Oracle GoldenGate Monitor Server.

The agent is a separate download and requires a separate installation process (described in *Installing and Configuring Oracle GoldenGate Monitor Agent*). You must install Oracle GoldenGate Monitor Agent 12c (12.1.3) to ensure full functionality and take advantage of all command and control features (for example, edit, stop, and display logs).

1.2.3 Oracle GoldenGate Monitor Server

The Oracle GoldenGate Monitor Server coordinates the monitoring of multiple Oracle GoldenGate instances. The Oracle GoldenGate Monitor Server is a Java application that 1) processes information from Oracle GoldenGate agents and communicates it to the web browser and 2) manages user access roles, history, the display of information, and notifications triggered by events.

1.2.4 Oracle GoldenGate Monitor Repository

The Oracle GoldenGate Monitor Server uses a database as a central repository to store information about users and groups, process status, events, and other information. Oracle, MySQL and SQL Server repositories are supported. Details can be found online in the Certifications tab in My Oracle Support.

1.3 Understanding the Discovery Process

Oracle GoldenGate Monitor has preset definitions and rules that determine how it defines and automatically discovers solutions and databases.

1.3.1 Discovering Solutions

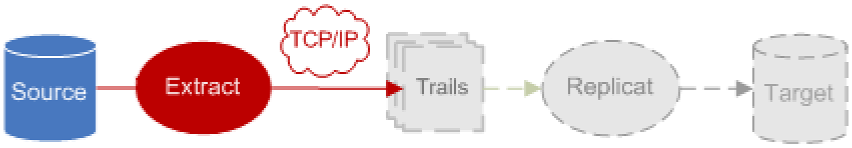
When you start your Oracle GoldenGate processes, the agent registers with the Oracle GoldenGate Monitor server. The server uses the information provided by the agent to look for solutions. Then when you log into the browser user interface, these solutions are available to display.

To be classified as a complete solution, there must be a continuous flow capturing and replicating changes from a source to a target database. The discovery process looks for complete solutions starting from a source database, creating a trail, and replicating the changes to a target database (a single end-to-end solution). Or it looks for processing that captures source database changes and delivers them to a target, and also captures changes from the target and delivers them to the source (a bi-directional solution).

1.3.2 Partial Solutions

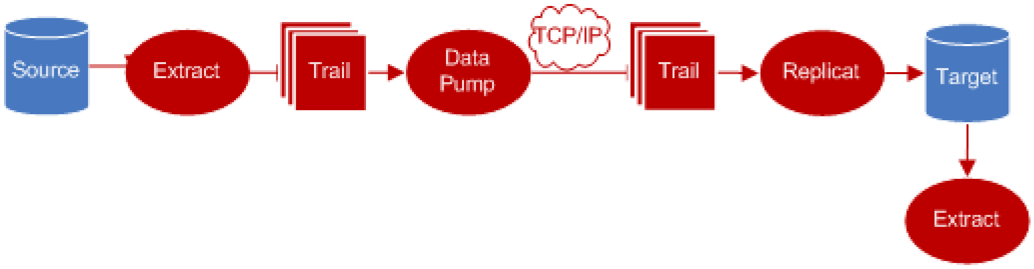
Partial solutions—that is, solutions that are not registered as continuously linked from the source to the target database—will appear in the Oracle GoldenGate Monitor console Diagram View. They will be in Solution node but have a default name starting with "Partial Solution" instead of "Solution".

This diagram represents a partial solution with the Replicat configured but not started. Since the Replicat has never registered, the server does not see a continuous link to the target. Some pieces will show up in the tree view, but the configuration will not be included in the solutions list.



Registration of the Manager includes the names of its processes, so the Extract and Replicat names will be listed with the instance in the tree view whether they have registered or not. Linked databases and remote trails do not show up until the process registers, so the remote trail and the target database in the example will not be listed until the user starts the Replicat and it registers. This example will become a solution once the Replicat registers.

In this second example there are continuous links from the source to the target databases, but there is no link from the Extract that is attached to the target. Thus, this will show up as a partial solution.



1.3.3 Discovering Databases

Recognition of your source or target database by Oracle GoldenGate Monitor processing depends on the database type, where it is installed, and the process that is registering it. The solution discovery process compares the database instance, the Manager's port, and the host name to decide whether databases are the same.

The same database instance on the same host will be categorized as a separate database if it is registered by two different Managers.

1.4 12*c* (12.1.3) Prerequisites

Oracle GoldenGate Monitor 12*c* (12.1.3) requires the following:

- JDK 1.7
- WebLogic Server 12c (12.1.3) with JRF
- One of the following databases:
  - Oracle 11gR1, 11gR2, 12c
  - MySQL 5.5, 5.6
  - SQL Server 2008, 2012

For more information on these prerequisites, see "Preparing to Install" in *Installing and Configuring Oracle GoldenGate Monitor Server*.

1.5 Platform Support

For information on Oracle GoldenGate Monitor 12*c* (12.1.3) platform support, see "Platform Support" in *Installing and Configuring Oracle GoldenGate Monitor Server*.

2 Using Oracle GoldenGate Monitor Server

This chapter describes tasks performed when using Oracle GoldenGate Monitor Server, such as starting and stopping the Oracle GoldenGate Monitor Server, starting the user interface, changing passwords, and changing settings for the memory allotment or the timeout interval.

This chapter includes the following sections:

2.1 Starting and Stopping Oracle GoldenGate Monitor Server

This section describes how to start and stop the Oracle GoldenGate Monitor Server.

2.1.1 Starting Oracle GoldenGate Monitor Server

Starting Oracle GoldenGate Monitor Server is a two or three step process

1. Start the WebLogic Administration Server.
2. If necessary, update the Oracle GoldenGate Monitor server credentials.
3. Start the WebLogic Managed Server

For complete details, see "Starting Oracle GoldenGate Monitor Server" in *Installing and Configuring Oracle GoldenGate Monitor Server*.

2.1.2 Stopping the Oracle Golden Monitor Server

Follow these steps to stop the Oracle GoldenGate Monitor Server:

- You should close any running user interface sessions.
- Shut down the server by navigating to `OGGMON_DOMAIN/bin` (`OGGMON_DOMAIN\bin` on Windows) and enter the following command:

  Linux:

  ```
  $ ./stopManagedWebLogic.sh
  ```

  Windows:

  ```
  C:\path\to\bin> stopManagedWebLogic.cmd
  ```

- In some instances, you might need to shut down an administration server; for example, if you are deinstalling Oracle GoldenGate Monitor. To stop an administration server, see "To stop the Administration Server".

2.2 Starting Oracle GoldenGate Monitor Console

With the WebLogic Server administration server and the managed Oracle GoldenGate Monitor Server both started, you can start the Oracle GoldenGate Monitor Console. To do so, follow the instructions in "Start the Oracle GoldenGate Monitor Console" in *Installing and Configuring Oracle GoldenGate Monitor Server*.

Once you are logged into the application, the solution discovery process will discover configured solutions.

2.2.1 Running Multiple Sessions

You can run multiple sessions of Oracle GoldenGate Monitor user interface from the same computer and browser if you are using Internet Explorer. You can also run one Internet Explorer and one Mozilla Firefox session at the same time, but multiple Mozilla Firefox sessions on the same computer are not supported.

2.3 Changing Oracle GoldenGate Monitor Server and Repository Passwords

Utilities are provided to allow you to change Monitor Server and repository passwords when necessary. You can also change the memory allotment to better tune Oracle GoldenGate Monitor to your needs.

2.3.1 Changing Passwords

Oracle GoldenGate Monitor Server passwords are initially set based on what you enter when you install the server. The Oracle Wallet is created by the install program to store the passwords on all supported platforms.

Change passwords by using the WebLogic Scripting Tool (WLST) command `updateCred()`. WLST is a command-line scripting environment by which create, manage, and monitor WebLogic domains.

1. Navigate to the `bin` subdirectory in `ORACLE_HOME/common` and launch WLST.

   On Linux

   ```
   $ORACLE_HOME/common/bin>./wlst.sh
   ```

   On Windows:

   ```
   C:\ORACLE_HOME\common\bin>./wlst.cmd
   ```

2. Connect to the server:

   ```
   wls:/offline> connect('username','password','host:port')
   ```

3. Use `updateCred()` to change the password, as shown here:

   - To change the JMX password, enter this:

     ```
     wls:/test_domain/serverConfig>updateCred(map="OGGMONITOR",key="
     WEB.JMX.PASSWORD",user="username",password="new_jmx_password",desc="JMX
     Password")
     ```

- To change the keystore password, enter this:

  ```
  wls:/test_
  domain/serverConfig>updateCred(map="OGGMONITOR",key="MONITOR.KEYSTORE.PASSW
  ORD",user="username",password="new_keystore_password",desc="Keystore
  Password")
  ```

- To change the truststore password, enter this:

  ```
  wls:/test_
  domain/serverConfig>updateCred(map="OGGMONITOR",key="MONITOR.TRUSTSTORE.PAS
  SWORD",user="username",password="new_truststore_password",desc="Keystore
  Password")
  ```

- To change the SMTP password, ener this:

  ```
  wls:/test_
  domain/serverConfig>updateCred(map="OGGMONITOR",key="MONITOR.SMTP.PASSWORD"
  ,user="username",password="new_smtp_password",desc="SMTP Password")
  ```

2.4 Changing the Memory Allotment

The amount of RAM allocated to the Oracle GoldenGate Monitor server affects the number of Oracle GoldenGate instances and processes that can be monitored.

> **Note:** For more information on setting the memory requirements, see 'Tuning Java Virtual Machines (JVMs)" in *Tuning Performance of Oracle WebLogic Server*.

Oracle recommends the following memory settings:

Setting	Size
Minimum Heap (`-Xms`)	512 MB
Maximum Heap (`-Xmx`)	1024 MB
Permgen Size (`-XX:PermSize`)	256 MB
Maximum Permgen Size (`-XX:MaxPermSize`)	512 MB

Use this command:

```
JAVA_OPTS=-Xms512m -Xmx1024m -XX:PermSize=256m -XX:MaxPermSize=512m
```

> **Note:** For Oracle GoldenGate Monitor Server to run on a Windows 32 bit system, the maximum memory allotment must be reduced to 800 MB and the MaxPermSize must be reduced to 340 MB. Reducing the memory allotment reduces the number of target systems that can be supported for monitoring. These installations will therefore not be able to monitor the number of target systems supported for other operating systems.

3

Using the Oracle GoldenGate Monitor Agent

This chapter explains how to use the Oracle GoldenGate agent. It includes information such as how to change passwords and how to change memory allotment settings.

This chapter includes the following sections:

- Section 3.1, "Installing Oracle GoldenGate Monitor Agent"
- Section 3.2, "Starting the Oracle GoldenGate Monitor Agent"
- Section 3.3, "Updating Oracle GoldenGate Monitor Agent Passwords"
- Section 3.4, "Changing Oracle GoldenGate Monitor Agent Memory Allotment"

3.1 Installing Oracle GoldenGate Monitor Agent

To install and configure Oracle GoldenGate agent, use the procedures described in *Installing and Configuring Oracle GoldenGate Monitor Agent*. Oracle GoldenGate agent *must* be installed on the same host as Oracle GoldenGate Monitor.

3.2 Starting the Oracle GoldenGate Monitor Agent

You need to start Oracle GoldenGate agent whenever you start Oracle GoldenGate Monitor. To do so, go to Oracle GoldenGate Core GGSCI and execute the `start jagent` command:

```
GGSCI>START JAGENT
```

Notes: If processes were added or deleted, Oracle GoldenGate agent will detect them without requiring restart.

3.3 Updating Oracle GoldenGate Monitor Agent Passwords

Oracle GoldenGate agent passwords are stored in the Oracle Wallet for all supported platforms except IBM z/OS, where they are stored in the `password.properties` file (found in the *`installation_location`*`/cfg` directory).

Use the `pw_agent_util.bat` and `pw_agent_util.sh` utilities to change agent passwords on all supported platforms.

To change agent passwords:

1. Navigate to the installation directory.

   ```
   Shell> cd ./installation_directory/
   ```

2. Using the appropriate runtime argument, run the appropriate `pw_agent_util` file.

   > **Note:** The password utility can be run only by the user who installed the Oracle GoldenGate instance.

   On Windows enter the following at the command line:

   ```
   Shell> pw_agent_util.bat -[updateAgentJMX | updateServerJMX | updateKeystore |
   updateTruststore]
   ```

   On UNIX enter the following command:

   ```
   Shell>./pw_agent_util.sh -[updateAgentJMX | updateServerJMX |  updateKeystore |
   updateTruststore]
   ```

   Where the `pw_agent_util` options are:

   - `-updateAgentJMX`, which changes the agent's JMX password.
   - `-updateServerJMX`, which changes Oracle GoldenGate Monitor Server's JMX password.
   - `-keystore`, which adds the Java keystore password.
   - `-truststore`, which adds the Java truststore password.
   - `-updateKeystore,Which` changes the Java keystore password.
   - `-updateTruststore`, which changes the Java truststore password.

   If the wallet is needed and it exists, the utility will prompt with the password to be modified. If the Oracle Wallet is needed and does not exist, the utility will return a message indicating that you will need to first run with the -create option and then the utility will stop. To create the wallet, use this command (using the same options as above):

   On Windows:

   ```
   Shell> ./pw_agent_util.bat -create | -jagentonly
   ```

   On Linux:

   ```
   Shell> ./pw_agent_util.sh -create | -jagentonly
   ```

3. Enter and confirm the new password to implement the change. Press **Enter** without entering any data to cancel the request.

4. To activate the changes, navigate to the Oracle GoldenGate installation location and bring up GGSCI. Then do one of the following depending on your Oracle GoldenGate release:

   - For Oracle GoldenGate release 11.1.1.1, stop and restart the Oracle GoldenGate Manager.

     ```
     GGSCI> STOP MANAGER
     GGSCI> START MANAGER
     ```

   - For Oracle GoldenGate release 11.2.1 and later, stop and restart the Oracle GoldenGate agent.

     ```
     GGSCI> STOP JAGENT
     GGSCI> START JAGENT
     ```

3.4 Changing Oracle GoldenGate Monitor Agent Memory Allotment

You can change the memory allotment for the standalone agent of Oracle GoldenGate release 11.2.1 and later by following these steps:

1. Navigate to the Oracle GoldenGate installation location.
2. Start GGSCI and edit the agent parameter file.

   ```
   Shell> GGSCI
   GGSCI> EDIT PARAMS JAGENT
   ```

3. The settings for the default memory allotment, `-Xms`, and the maximum memory allotment, `-Xmx`, are included in the start-up string for the agent. The following example sets the default to `64 MB` and the maximum to `512 MB`.

   ```
   java -jar -Xms64m -Xmx512m dirjar/jagent.jar
   ```

4. Change the allotment numbers as needed, save the parameter file, and exit the editor.
5. Stop and restart the agent to implement the changes.

   ```
   GGSCI> STOP JAGENT
   GGSCI> START JAGENT
   ```

4 Configuring and Using Alerts

This chapter describes how to create alerts to notify you when a certain condition exists. Optionally, these alerts can be delivered by email, command line interface, or Simple Network Management Protocol (SNMP).

This chapter includes the following sections:

- Section 4.1, "Overview"
- Section 4.2, "Configuring E-mail Alerts"
- Section 4.3, "Configuring CLI Alerts"
- Section 4.4, "Configuring SNMP Alerts"
- Section 4.5, "Enabling and Disabling Alerts"

4.1 Overview

Oracle GoldenGate Monitor alerts notify you when a specified condition exists for an Oracle GoldenGate component. For example, you can request notification when a process stops or when a specified lag threshold is reached. You select the information to include in the message. To define alerts go to **Alert Definitions** in the user interface and follow the instructions in the online help.

Each user specifies which types of alerts Oracle GoldenGate Monitor should produce for them. To enable alerts for a user, go to the **User Profile** in the Oracle GoldenGate Monitor user interface and follow the instructions in the online help.

4.2 Configuring E-mail Alerts

You can configure Oracle GoldenGate Monitor alerts to be delivered to e-mail accounts.

To use this feature you must:

1. Enable e-mail alerts by checking the SMTP (Simple Mail Transfer Protocol) alerts box during installation or later setting the e-mail alerts properties in the `monitor.properties` file as explained in the next section.
2. Enter the user's e-mail address in the **User Management** tab of the Oracle GoldenGate Monitor user interface.
3. Go to the **User Profile** in the Oracle GoldenGate Monitor user interface and select e-mail as the notification type for the appropriate severity level.

4.2.1 Setting E-mail Alert Properties

If you did not set up e-mail alerts during the installation, you can do so by setting the following properties in the `monitor.properties` file:

- Enable e-mail alerts by setting the following property to `true`:

  ```
  monitor.smtp.alerts.enabled=true
  ```

- Specify the name of the sender for Oracle GoldenGate Monitor communications generated from the e-mail server.

  ```
  monitor.smtp.from=sender_name
  ```

- Specify the host name of the e-mail server.

  ```
  monitor.smtp.host=email_host_name
  ```

- Specify the port that the e-mail server uses.

  ```
  monitor.smtp.port=port_number
  ```

- Specify whether the e-mail server is in secure mode.

  ```
  monitor.smtp.secure={true | false}
  ```

- If the e-mail server is in secure mode, specify the user authorized to log in.

  ```
  monitor.smtp.user=user_name
  ```

4.2.2 Setting the Password for Secure Mode

If the e-mail server is running in secure mode, to change your password, you must log in to the WebLogic Scripting Tool (WLST) and use the `createCred()` command; for example:

```
wls:/test_
domain/serverConfig>createCred(map="OGGMONITOR",key="WEB.SMTP.EMAIL.PASSWORD",user
="<email user id>",password="<email password>",desc="SMTP EMAIL Password")
```

For more information on using WLST, see *Understanding the WebLogic Scripting Tool*.

4.3 Configuring CLI Alerts

The Oracle GoldenGate Monitor Command-Line Integration (CLI) allows you to run a script or object file on the Oracle GoldenGate Monitor Server when an alert is triggered.

To use this feature you must enable CLI alerts by checking the CLI alerts box during installation or later setting the `monitor.cli.alerts.enabled` property equal to `true` in the `monitor.properties` file.

4.3.1 Setting Up Command-Line Handlers

The Oracle GoldenGate Monitor installation delivers files to help you configure your CLI interface. These are delivered to the `cfg` subdirectory of the installation location.

- `CommandLineHandlers.xml`

  The CLI interface is configured in the `CommandLineHandlers.xml` file.

  Two example `CommandLineHandlers.xml` files, one for UNIX and one for Windows, are included with the installation. Each contains sample syntax for configuring a CLI interface. You can copy the appropriate version and then add and change arguments to create the `CommandLineHandlers.xml` that will configure your CLI interface.

> **Note:** The `CommandLineHandlers.xml` file must be set up outside of the Oracle GoldenGate Monitor user interface by the Oracle GoldenGate Monitor Server host administrator that installed the system.

- `CommandLineHandlers.xsd`

  This file contains the definition for the `CommandLineHandlers.xml` file. It can be used to generate the `CommandLineHandlers.xml` using a commercial or open source XML generation tool that creates sample XML from XSD.

After you configure the `CommandLineHandlers.xml`, stop and restart the Oracle GoldenGate Monitor Server to activate the changes. See "Starting and Stopping Oracle GoldenGate Monitor Server" on page 2-1 for directions on how to do this.

4.3.1.1 Command-Line Handler Arguments

The example UNIX configuration below illustrates the structure and arguments of the XML configuration file. The header values should not be changed. These values specify the version and coding of the XML.

Arguments are specified by entering a value within quotation marks after the equal sign (=) as shown in the example. In this illustration namespace and schema information has been omitted as indicated by the ellipses (. . .).

```
<?xml version="1.0" encoding="UTF-8"?>
<CommandLineHandlers . . .>
<CommandLineHandler dateTimeFormat="MMddyyyyHHmmssSSS"
  executeIn="/home/user" name="CMDLINE">
    <externalCommand>touch</externalCommand>
    <arguments>
      <argument argText="filename" name="hostname"
      presentIfEmpty="true" quoted="false"/>
  </arguments>
  <alertMappings>
    <alertMapping alertField="host" name="hostname"/>
  </alertMappings>
  </CommandLineHandler>
</CommandLineHandlers>s
```

`CommandLineHandler` is the parent tag for the CLI alert handler. This is specified within the `CommandLineHandlers` tags.

```
<CommandLineHandler dateTimeFormat="MMddyyyyHHmmssSSS"
executeIn="/home/user" name="CMDLINE">
```

The `CommandLineHandler` tag includes the following arguments:

- `dateTimeFormat`

  This is the standard Java format argument described in Java documentation.

- `executeIn`

  The `executeIn` argument triggers the processing to move into the specified directory before running the external script or object file. The default is to use the current run directory of the virtual machine (VM); the directory in which the script or command was started.

A `RunTimeException` is generated when the alert is triggered if the specified directory does not exist or if the `executeIn` attribute is empty or not present.

- name

  This will always be "`CMDLINE`".

The following example illustrates the tags that can be nested within the `CommandLineHandler` tags:

```
<externalCommand>touch</externalCommand>
<arguments>
    <argument argText="filename" name="hostname"
    presentIfEmpty="true" quoted="false"/>
</arguments>
<alertMappings>
    <alertMapping alertField="host" name="hostname"/>
</alertMappings>
```

- `externalCommand`

  The value in `externalCommand` specifies the absolute path to the script or object file. If the system path environment variable points to the directory of the file to be run, you can specify the script or object file name without the path.

- `arguments`

  The `arguments` tag specifies one or more values that are appended to the directory value specified in the `externalCommand` tag.

  For each argument the following attributes can be specified:

  `argText` - Specifies a literal text argument that is sent with the `externalCommand` tag.

  `name` - Can be a name or it can work with `alertMappings` to find a name as explained below.

  `presentIfEmpty` - Works with the `alertMappings` tag to add selected information associated with the alert definition to the `externalCommand` tag. See alertMappings below for more detail.

  `quoted` - Specifies whether quotation marks should be added.

- `alertMappings`

  The `alertMappings` tag appends the value extracted from the alert definition information to the value specified in the `externalCommand` tag.

  ```
  <alertMappings>
      <alertMapping alertField="host" name="hostname"/>
  </alertMappings>
  ```

  The alertField can be one of the following values associated with the alert definition:

  `alertName` - The name of an alert definition.

  `host` - The host of the Oracle GoldenGate object whose monitoring point triggers the alert.

  `alertObjectName` - The name associated with the object whose monitoring point triggers the alert, such as an Extract process named EXACCT.

alertTime - The time that the alert was triggered.

alertSeverity - The severity level defined for the alert; either Warning or Error.

alertMessage - The message generated by the alert. This is a combination of the condition defined for the alert, the value of the monitoring point, and literal text.

changedValue - The new monitoring point value that triggered the alert. For example, you create an alert that is triggered when lag is greater than 5 seconds. The lag is 4 seconds and then it goes to 7 seconds. This triggers the alert and the changedValue is 7.

In the following example, the name attributes in the argument and the alertMapping tags are matched to extract the value from the alertField attribute. The argument name "hostname" is matched to the alertMapping name "hostname" to find the value of alertField, which is "host". This tells the system to append the host of the Oracle GoldenGate object that triggered the alert to the value specified in the externalCommand tag.

```
<arguments>
    <argument argText="text" name="hostname" presentIfEmpty="true"
    quoted="false"/>
</arguments>
<alertMappings>
    <alertMapping alertField="host" name="hostname"/>
</alertMappings>
```

The presentIfEmpty attribute works with the alertMappings tag to determine what to do if the alertField is not valid or the name attributes do not match:

– presentIfEmpty="true"

  The value in the argText attribute is used in the external command.

– presentIfEmpty="false"

  The entire argument is omitted.

4.3.2 Sample Command-Line Handlers

These examples run a batch script on the Oracle GoldenGate Monitor Server.

4.3.2.1 Running on a Windows Server

The following example runs the batch script sample_cli.bat on a Windows server hosting the Oracle GoldenGate Monitor Server. The server of the Oracle GoldenGate instance ("host") that triggered the alert is appended to the name of the batch script specified in the externalCommand.

```
<?xml version="1.0" encoding="UTF-8"?>
<CommandLineHandlers
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.yourlocation/monitor/commandlinehandlers/CommandLin
eHandlers.xsd">
  <CommandLineHandler dateTimeFormat="MMddyyyyHHmmssSSS"
  executeIn="C:\" name="CMDLINE">
    <externalCommand>c:\sample_cli.bat</externalCommand>
    <arguments>
      <argument argText="" name="hostname" presentIfEmpty="true" quoted="false"/>
    </arguments>
    <alertMappings>
      <alertMapping alertField="host" name="hostname"/>
    </alertMappings>
  </CommandLineHandler>
</CommandLineHandlers>
```

4.3.2.2 Running on a UNIX Host

The following example runs the `sample_cli.sh` script on the UNIX server hosting the Oracle GoldenGate Monitor Server. The server of the Oracle GoldenGate instance ("`host`") that triggered the alert is appended to the name of the batch script specified in the `externalCommand`.

```
<?xml version="1.0" encoding="UTF-8"?>
<CommandLineHandlers
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.yourlocation/monitor/commandlinehandlers/CommandLin
eHandlers.xsd">
  <CommandLineHandler dateTimeFormat="MMddyyyyHHmmssSSS"    executeIn="/home/user"
name="CMDLINE">
    <externalCommand>bash</externalCommand>
    <arguments>
      <argument argText="/home/user/sample_cli.sh" name="hostname"
presentIfEmpty="true" quoted="true"/>
    </arguments>
    <alertMappings>
     <alertMapping alertField="host" name="hostname"/>
    </alertMappings>
  </CommandLineHandler>
</CommandLineHandlers>
```

4.4 Configuring SNMP Alerts

The Oracle GoldenGate Monitor Simple Network Management Protocol (SNMP) interface sends alerts in the form of datagrams. These are picked up by an SNMP trap recipient listening on a specified port.

To use this feature you must enable SNMP alerts during installation or later set the `monitor.snmp.alerts.enabled` property equal to `true` in the `monitor.properties` file.

4.4.1 Importing the MIB File

The `GoldenGate-Monitor-mib.mib` file is delivered to the `cfg` subdirectory during the installation of Oracle GoldenGate Monitor. This contains the Management Information Base (MIB) definitions the target uses to interpret the alerts. If you need to interpret information received in the traps, import this file to the target tool.

4.4.2 Configuring SNMP Alerts

The SNMP alert is configured in the `SNMPJMXMapping.xml` file that is delivered to the `cfg` subdirectory during the Oracle GoldenGate installation.

Any changes to the `SNMPJMXMapping.xml` file must be made outside of the Oracle GoldenGate Monitor user interface by the host administrator that installed the Oracle GoldenGate Monitor Server software.

You should only change the sections of the `SNMPJMXMapping.xml` file that set the SNMP version and define the targets.

.

```
.
.
<MIBTree>
.
.
.
 <notifications type="NOTIFICATIONS">
  <notification version="2" enabled="true">
   <targets>
    <target timeout="200" retry="0">localhost/162
    </target>
   </targets>
.
.
.
  </notification>
  <notification version="1" enabled="false">
   <targets>
    <target>localhost/162
    </target>
   </targets>
.
.
.
  </notification>
 </notifications>
</MIBTree>
```

4.4.2.1 Setting the SNMP Version

The SNMP version is initially set based on the entry during installation. You can change it later by resetting the `notification version 1 enabled` value and `notification version 2 enabled` value. Set one to `true` and the other to `false`.

The targets that are defined for the enabled version are used. The targets for the disabled version are ignored.

4.4.2.2 Setting the SNMP Targets

Define your targets within the `<target> </target>` tags by entering the host name and port number separated by a forward slash.

4.5 Enabling and Disabling Alerts

An Oracle GoldenGate Monitor administrator can disable any category of alerts so that those alerts are not delivered to remote clients. You might, for example, want to disable alerts during planned outages, such as when processes are stopped during maintenance windows. Although the disabled alerts are not sent to remote clients, they will be recorded in the Oracle GoldenGate Monitor user interface.

To disable alerts:

1. Set the appropriate properties in the `monitor.properties` file of Oracle GoldenGate Monitor Server to false.

   - To disable SMTP delivery, set `monitor.smtp.alerts.enabled=false`

   - To disable CLI delivery, set `monitor.cli.alerts.enabled=false`

   - To disable SNMP delivery, set `monitor.snmp.alerts.enabaled=false`

2. Stop and restart Oracle GoldenGate Monitor Server. See "Starting and Stopping Oracle GoldenGate Monitor Server" on page 2-1 for directions on how to do this.
3. To enable the alerts again, set the properties back to true, and then stop and restart Oracle GoldenGate Monitor Server.

5 Using SSL Communication

This chapter describes how to use Secure Sockets Layer (SSL) for the Java Management Extensions (JMX) communication between Oracle GoldenGate Monitor Server and Oracle GoldenGate agents when monitoring Oracle GoldenGate instances.

The chapter includes the following sections:

- Section 5.1, "Using Secure Sockets Layer (SSL)"
- Section 5.2, "Enabling SSL"
- Section 5.3, "Creating Self Signed Certificates"
- Section 5.4, "Using Certificate Authority (CA) Documents"

5.1 Using Secure Sockets Layer (SSL)

Oracle GoldenGate Monitor communication is both from the Oracle GoldenGate Monitor Server to the agent of the Oracle GoldenGate instance and from the agent to the server. Oracle GoldenGate Monitor supports either SSL and or secure JMX communications but not both concurrently.

5.1.1 Considerations for using SSL

Normally the Oracle GoldenGate Monitor Server and agents encrypt the data sent between them. Using SSL adds the following security:

- SSL communication between the Oracle GoldenGate Monitor Server and the Oracle GoldenGate agents must be verified by credentials in the form of certificates.
- Information passed between Oracle GoldenGate Monitor Server and the Oracle GoldenGate agents on a network by using SSL is encrypted in a manner that helps to ensure the data will not be modified in transit by third parties.

You can elect to use SSL or not to use it, but you cannot do both at the same time. If your Oracle GoldenGate Monitor Server uses SSL, all the agents that communicate with it must use SSL. The option of using SSL communication for monitoring Oracle GoldenGate instances is subject to the following:

- The default is not to use SSL; you must enable SSL to use it.
- You can use SSL for monitoring Oracle GoldenGate instances on any supported Oracle GoldenGate platform except HP NonStop and IBM AS/400, which are not supported for monitoring.
- If you use SSL, the repositories it uses, called keystores, are protected by a password. This password is stored in the Oracle Wallet on all supported platforms

except IBM z/OS. For z/OS all passwords are stored in the `password.propterties` file on the `cfg` subdirectory of the installation directory.

- Oracle GoldenGate Monitor supports only two-way SSL authentication and Oracle GoldenGate Monitor Server and Oracle GoldenGate agent use bidirectional communication. The Oracle GoldenGate Monitor Server and Agent will each sometimes act as the server (receiving requests) and sometimes as the client (sending requests).
- Setting up SSL involves creating the keystore and certificates using the Java Keytool. This chapter describes some of the steps to generate keys and certificates for SSL, but assumes that you have knowledge of the Java Keytool and the keystore, have read the Java Keytool documentation, and are aware of the recommended security considerations.

5.1.2 Terminology

The following terms are Java terms that are used in this chapter. Refer to the Java Keytool documentation for more extensive definitions and definitions of related terms.

Keystore

Keystores are databases used for SSL authentication. At a minimum, the keystore stores the private keys and certificates with corresponding public keys that are used to identify one client or one server. However, the truststore can be configured to merge into the keystore, adding the certification from other parties.

Certificate

A certificate is a digitally signed statement from an entity stating that the public key it references is valid because the signature can be verified to check authenticity.

Truststore

Truststores store the public keys and certificates from other parties that you will communicate with and from third party certification authorities (CAs) trusted to sign (issue) certificates for other entities. The truststore may be configured to merge into the keystore, so that all keys and certificates are in one place.

Keytool

Keytool is a Java utility for generating and managing keys and certificates.

5.2 Enabling SSL

Oracle GoldenGate Monitor Server and Oracle GoldenGate Monitor Agents are not enabled for SSL by default. If you decide to use SSL, you must enable the properties for the server and the agents. You must also create the keystores and ensure each resides in the designated location.

5.2.1 Setting SSL Properties

To enable SSL, you must change a property setting for both the server and the agent .

For the Oracle GoldenGate Monitor Server:

1. Navigate to the installation directory of the Oracle GoldenGate Monitor Server.
2. Edit the `monitor.properties` file to set the `monitor.ssl` property to `true`.

3. Stop and restart the Oracle GoldenGate Monitor server to activate the new settings.

For each Oracle GoldenGate Monitor Agent:

1. Navigate to the installation directory of the Oracle GoldenGate instance.
2. Edit the `Config.properties` file for the agent to set the `jagent.ssl` property to `true`.

> **Note:** If you enable SSL for the Oracle GoldenGate Monitor Server, you must also enable SSL for all of the agents that communicate with the server.

5.2.2 Creating the Keystore

SSL requires that you create keystores to store the certificates and key pairs. You can create these using the Keytool for Java Secure Sockets Extension (JSSE). To access the Keytool documentation for Solaris/UNIX or Windows, see "JDK Tools and Utilities" at: `http://docs.oracle.com/javase/6/docs/technotes/tools`

Keystore Location

The keystores for Oracle GoldenGate Monitor must be stored in the following specific directories of the server and agent installation locations:

- The keystore with certification for the agent must be stored in the `dircrt` subdirectory off the installation directory of the Oracle GoldenGate instance. The default name is `jagentKeyStore`.
- The keystore and certificate for the Monitor server must be stored in the `cert` subdirectory off the installation directory of the Oracle GoldenGate Monitor Server. The default name is `monitorKeyStore`.

By default, the keystore file is created in the directory from which the Keytool is run. Either run the Keytool from the specified locations, or move the keystore after it is created.

You can change the default names of the keystores by setting the `jagent.keystore.file`, `jagent truststore.file`, `monitor.keystore.file`, or `monitor.truststore.file` properties. If the keystore and truststore files are set to the same name, that one file will be used to store both your certificates and trusted certificates from other entities.

Keystore Password

The keystore is protected by a password created when you create the certificates that identify the Oracle GoldenGate Monitor Server or Agent. This password must also be added to the Oracle Wallet (or to the `password.properties` file for IBM z/OS.)

> **Note:** Oracle GoldenGate Monitor passwords are stored in the Oracle Wallet for all supported platforms except IBM z/OS. For z/OS, they are stored in the `password.propterties` file, which is in the `cfg` subdirectory of the installation location for the Oracle GoldenGate Monitor Server or Agent.

For Oracle GoldenGate Monitor Server:

- Add the keystore password by using the WebLogic Scripting Tool command `updateCred()`, as described in Section 2.3.1, "Changing Passwords."
- If you are using a separate truststore, also add that password, also by using the `updateCred()`.

For each Oracle GoldenGate agent:

- Add the password for the Oracle GoldenGate agent, as explained in Section 3.3, "Updating Oracle GoldenGate Monitor Agent Passwords."
- If you are using a separate truststore, also add the truststore password by using the `-truststore` option of the utility.

5.3 Creating Self Signed Certificates

The Java Keytool can be used to create the certificates needed to verify Oracle GoldenGate Monitor Server and the Oracle GoldenGate agents. The Keytool stores these certificates in the keystore.

5.3.1 Establishing a Self-signed Certificate for Monitor Server

You need to first create the self-signed certificate that identifies the Oracle GoldenGate Monitor Server and then export it into the Oracle GoldenGate Monitor Agent truststore.

Creating the Certificate to verify Monitor Server

Follow these steps to create the certificate. Refer to the Keytool for Java documentation for more detail on the commands and options.

1. Navigate to the Oracle GoldenGate Monitor Server installation directory and then to the `cert` subdirectory. This is the location of the Monitor Server keystore.
2. Create the certificate used by Monitor to verify itself and to encrypt the communication.

   For example:

   ```
   keytool -genkeypair -keystore monitorkeystore -keyalg rsa -alias monalias
   -storepass keystorepw -keypass serverpw
   ```

   The option `-keystore` identifies the name of the keystore repository. If you do not use the `monitorkeystore` default name, ensure that you update the `monitor.properties` file with the name you select.

   `-keyalg` identifies the encrypting algorithm, which must be `rsa`.

   `-alias` specifies an identifier for the new keystore entry that will be created; *`monalias`* in this example.

   `-storepass` identifies the password that protects the keystore. `-keypass` identifies the password that protects the private key of the generated key pair identifying the Monitor Server. If you do not provide a password for one of these, you will be prompted to enter it.

   > **Note:** Record the keystore password you select so you can add it to Oracle GoldenGate Monitor Server or Agent as explained in Section 5.2.2, "Creating the Keystore."

The program will prompt you to answer the following questions:

```
What is your first and last name?
What is the name of your organizational unit?
What is the name of your organization?
What is the name of your City or Locality?
What is the name of your State or Province?
What is the two-letter country code for this unit?
```

After you enter your answers, the program will display them and ask you to confirm they are correct. This information becomes part of the certificate.

3. (Optional) Verify the certificate creation by listing the entry that was created using a command similar to:

   ```
   keytool -list -alias monalias -keystore monitorkeystore
   ```

4. Export the Oracle GoldenGate Monitor Server certificate from the *monitorkeystore* keystore to a file. For example:

   ```
   keytool -exportcert -alias monalias -file moncert.crt -keystore monitorkeystore
   storepass keystorepw
   ```

   The option `-file` identifies the file that will hold the *monalias* certificate after the export. `-keystore` identifies the repository that contains the `alias` that is to be exported.

   Optionally you can generate a Certificate Signing Request (CSR) with the `-certreq` option and send the certificate to a third party CA for signing.

5. (Optional) Print the certificate information. For example:

   ```
   keytool -printcert -file moncert.crt
   ```

6. Copy the Oracle GoldenGate Monitor Server certificate file created by the Keytool `exportcert` command (*moncert.crt* in the example) to the Oracle GoldenGate agent *installation location*/`dircrt` directory.

Importing the Server Certificate to the Agent Keystore

Perform the following steps to import the Oracle GoldenGate Monitor Server certificate to the agent truststore.

1. Navigate to the location of the Oracle GoldenGate agent keystore (*installation location*/`dircrt`).

2. Import the certificate to the Oracle GoldenGate agent keystore. For example:

   ```
   keytool -importcert -alias agentalias -file moncert.crt -keystore
   jagentkeystore storepass agentkeystorepw
   ```

   This reads the certificate from the *moncert.crt* file and stores it in the *jagentkeystore*.

   Optionally, you can use the *-trustcerts* option to import additional certificates for the chain of trust from a system-wide keystore of CA certificates. For example, the following command will trigger Keytool to attempt to establish a trust path from the *moncert.crt* up to a self-signed certificate.

   ```
   keytool -importcert -trustcacerts -alias agentalias -file moncert.crt -keystore
   jagentkeystore -storepass agentkeystorepw
   ```

3. (Optional) List the certificates that have been created in the keystore to verify the import. For example:

```
keytool -list -keystore jagentkeystore storepass agentkeystorepw
```

5.3.2 Establishing a Self-signed Certificate for the Agent

You need to create a self-signed certificate to verify the Oracle GoldenGate Monitor Agent and import the certificate to the Oracle GoldenGate Monitor Server.

5.3.2.1 Creating the Certificate to Verify the Agent

Follow these steps to create the certificate for the agent. Refer to the Keytool documentation for detail on the commands and options.

1. Navigate to the location of the Oracle GoldenGate agent keystore (*installation location*/dircrt).
2. Generate the agent certificate using a Keytool command similar to the following:

   ```
   keytool -genkeypair -alias agentalias -keystore jagentkeystore
   ```

   In this example we are using the same keystore, *jagentkeystore*, that we used for the trusted certificate from the Oracle GoldenGate Monitor Server.

   Since we did not enter a keystore password, the Keytool will prompt the user to enter it.
3. (Optional) List the certificates in the keystore to verify the creation. For example:

   ```
   keytool -list -keystore jagentkeystore -storepass agentkeystorepw
   ```
4. Export the certificate that verifies the agent into a file. For example:

   ```
   keytool -exportcert -alias agentalias -file agentclient.crt -keystore
   jagentkeystore -storepass agentkeystorepw
   ```
5. (Optional) Print the certificate information.

   ```
   keytool -printcert -file agentclient.crt
   ```
6. Copy the certificate file (*agentclient.crt* in this example) to the Oracle GoldenGate Monitor Server *installation location*/cert directory.

5.3.2.2 Importing the Agent Certificate to the Monitor Server

Follow these steps to import the certificate into the Oracle GoldenGate Monitor Server truststore.

1. Navigate to the location of the Oracle GoldenGate Monitor Server keystore (*installation location*/cert).
2. Import the agent certificate into the Oracle GoldenGate Monitor Server keystore. For example:

   ```
   keytool -importcert -alias agentalias -file agentclient.crt -keystore
   monitorkeystore -storepass keystorepw
   ```
3. (Optional) List the certificate information to verify the import.

   ```
   keytool -list -keystore monitorkeystore -storepass keystorepw
   ```

5.4 Using Certificate Authority (CA) Documents

The Java Keytool can be used to request that a certificate be signed by a trusted third party. Once the Certificate Authority (CA) signs the certificate, it can be imported into a keystore or truststore to provide identification and validation.

5.4.1 Generating a Certificate Signing Request

Follow these steps to generate a Certificate Signing Request (CSR). In this example we are using different names for the keystores to avoid limiting the example to either the Oracle GoldenGate Monitor Server side or the Agent side.

1. Generate the CSR using a Keytool command similar to the following.

   ```
   keytool -certreq -v -alias certalias -file cert.csr -keypass keypassword
   -storepass keystorepw -keystore keystore.jks
   ```

   This command will create a file named *cert.csr* as specified in the `-file` option. The `-v` option signifies verbose mode, which will output more information.

2. Submit the CSR *cert.csr* to the third party that is to sign the certificate.

5.4.2 Importing the Certificate

Once you get the signed certificate back from the third party CA, import both the signed certificate and the certificate of the CA that signed it into your keystore.

1. Convert the CA certificates and the signed certificate into PEM format and store them in the directory of the keystore. Each certificate, including those in a chain, must be stored individually.

2. Import the top (or root) certificate in the chain, which must be the self-signed certificate of the CA, using a command similar to the following:

   ```
   keytool -importcert -v -noprompt -trustcacerts -alias rootCA -file rootCA.pem
   -keystore keystore.jks -storepass keystorepw
   ```

   In this example, *rootCA* is the alias of the certificate and *rootCA.pem* is the file that contains the top certificate in the chain.

   If the import is successful, Keytool will display the message *Certificate was added to keystore*.

   > **Note:** For some third party Certification Authorities, there may be two CA certificates in the chain: the root certificate and an intermediate CA certificate. In this case, the intermediate certificate should be imported directly after the root certificate.

3. Repeat the import for each of the certificates (`.pem` files) in the chain. Create a different `alias` for each one.

4. After all of the certificates have been successfully imported, import the signed reply certificate using a command similar to the following:

   ```
   -keytool -importcert -v-alias certalias -file cert.pem -keystore keystore.jks
   -keypass keypass -storepass keystorepw
   ```

   Use the same `alias` that you used when generating the certificate and the CSR.

If the import was successful, Keytool will display the message *`Certificate reply was installed in the keystore`*.

5. List all of the CA certificates that were imported.

   ```
   keytool -list -keystore keystore.jks -storepass keystorepw
   ```

6. Export the public certificate using a command similar to the following, which will create the file *`cert.cer`*.

   ```
   keytool -exportcert -alias certalias -file cert.cer -keystore keystore.jks
   -storepass keystorepw
   ```

   Repeat the export for each of the certificates.

7. To import a certificate and its CA certificates to the keystore, first store all of the certificates in separate files in `.pem` format. Then import all of the certificates, starting with the top or root certificate, giving each one a different alias. For example to import the root certificate:

   ```
   keytool -importcert -v -noprompt -trustcacerts -alias IDrootCA -file
   IDrootCA.pem -keystore keystore.jks -storepass storepasspw
   ```

   Repeat the import for each of the certificates.

6

Understanding Instance Level Security

This chapter describes how to implement instance-level security in Oracle GoldenGate Monitor 12*c* (12.1.3). It contains the following sections:

- Section 6.1, "Overview"
- Section 6.2, "Instance Access Rules"
- Section 6.3, "Instance Access Behavior"
- Section 6.4, "User Instance Matrix"

6.1 Overview

In addition to the current functional level of security, instance level security is available in Oracle Golden Gate Monitor 12.1.3 (12c) in order to restrict individual user access to different hosts/instances.

6.2 Instance Access Rules

The following instance access rules apply:

- A Super Administrator will always have access to all the Oracle GoldenGate agent instances, and can always assign instances directly to users belongs to Operator/Power Operator and Administrator roles.
- Administrators are not authorized to make any changes on the instances mapped to Super Administrators & other Administrators. Administrators can map and un-map the instances (that is, instances that are accessible by the current logged in Administrator) only for Operator and Power Operators.
- Administrators and Operators/Power Operators can have shared instance access; that is, multiple users can have access to common/same instances from the instance pool; for example, Pool [1,2,3,4,5] Ad1 -->[1,2,3], Ad2-->[3,4,5]
- A Super Administrator will always have the full/combined set of instances accessible to Operator/Power Operators, which are granted from different Administrators.A Super Administrator can override the instances assigned to Operator/Power Operators by an Administrator; for example: A user logs in as a Super Administrator and clicks on the user Opr1. The Super Administrator should see that only instances 1, 2, 3, 4,5 are enabled because the user Admin2 has access to instances 3,4,5 and had previously assigned 3,4,5 instance access to user Opr1.
- Administrators can reassign the instances deleted by the Super Administrator to Operators/Power Operators, provided the Administrator has access to those instances.

- If the Administrator gets deleted, the instances assigned to any Operator or Power Operators will remain as is. In the absence of a deleted Administrator, the instances owned by Operators or Power Operators will be administrated by the Super Administrator.
- A user can be assigned multiple roles. Oracle GoldenGate Monitor will consider the highest role for that user; for example, Admin1 can be both a Super Administrator and an Administrator.

Role Assignment Example

The following scenario shows how the preceding rules are applied during role/instance assignment. The roles used in this example are:

- SA: Super Admin
- JI: Jagent Instance
- Ad1, Ad2: Administrators
- Opr1, Opr2: Operators
- PowerOpr1,PowerOpr2: Power Operators.

The available Oracle GoldenGate agent Instances are: {1, 2, 3, 4, 5}

These steps illustrate how roles are assigned specific instances:

1. SA: Has access to JI 1,2,3,4,5
2. SA: Assigns JI 1,2,3 to Ad1
3. SA: Assigns JI 3,4,5 to Ad2
4. Ad1: Assigns JI 1,2,3 to Opr1
5. Ad2: Assigns JI 3,4,5 to Opr1
6. Ad1: Assigns JI 1,2,3 to PowerOpr1
7. Ad2: Assigns JI 3,4,5 to PowerOpr1

6.3 Instance Access Behavior

When the user tries to access the instance that are not assigned to that him or her, the user will see a message saying that he or she does not have access to the instance. The same behavior applies on the solutions/Views that are part of a specific instance.

6.4 User Instance Matrix

Logged In User	EDIT USER Instance Enabled/Disabled	EDIT USER Instance Enabled/Disabled	EDIT USER Instance Enabled/Disabled	EDIT USER Instance Enabled/Disabled
Logged in as Super Administrator	Super Administrator Instance(s) Disabled	Administrator Instance(s) Enabled	Operator Instance(s) Enabled	Power Operator Instance(s) Enabled
Logged in as Administrator	Super Administrator Instance(s) Disabled	Administrator Instance(s) Disabled	Operator Instance(s) Enabled	Power Operator Instance(s) Enabled

Note: Instance level security is not available for Alerts generations. Users can create alerts for the instance objects that are not accessible by that user.

7 Commands and Parameters

This chapter describes the parameters set in the Oracle GoldenGate instance to enable monitoring, and the commands used to create and alter the Oracle GoldenGate Monitor data store and to start the Oracle GoldenGate Monitor Agent that handles monitoring data.

This chapter includes the following sections:

- Section 7.1, "Parameters"
- Section 7.2, "GGSCI Commands"

7.1 Parameters

Oracle GoldenGate parameters are used to configure, run, and manage Oracle GoldenGate processes. The parameters included here apply to monitoring by Oracle GoldenGate Monitor Server or Oracle Enterprise Manager.

7.1.1 ENABLEMONITORAGENT

Valid for

`GLOBALS`

Use the `ENABLEMONITORAGENT` parameter to enable the Oracle GoldenGate agent for Oracle GoldenGate release 11.1.1.1. For more information on this parameter, consult the *Oracle GoldenGate Windows and UNIX Reference Guide* for the 11.1.1.1 release.

> **Note:** `ENABLEMONITORAGENT` is deprecated for Oracle GoldenGate release 11.2.1 and later.

Syntax

```
ENABLEMONITORAGENT
```

7.1.2 ENABLEMONITORING

Valid for

`GLOBALS`

Use the `ENABLEMONITORING` parameter to enable monitoring for Oracle GoldenGate. Monitoring is enabled for Extract, Replicat, and Manager processes within the instance of Oracle GoldenGate to which it is applied.

ENABLEMONITORING activates collection of monitoring points providing status and other information on the Oracle GoldenGate processes to Oracle GoldenGate Monitor or Oracle Enterprise Manager.

> **Note:** ENABLEMONITORING is a valid parameter for Oracle GoldenGate release 11.2.1 and later.

Syntax

```
ENABLEMONITORING
```

7.1.3 AUTOSTART JAGENT

Valid for

Manager

Use the JAGENT option of the AUTOSTART parameter to automatically start the Oracle GoldenGate agent when Manager starts up.

You can use multiple AUTOSTART statements in the same parameter file.

Syntax

```
AUTOSTART {{EXTRACT | REPLICAT | ER} group_name | JAGENT}
```

Example

```
AUTOSTART JAGENT
```

7.2 GGSCI Commands

The Oracle GoldenGate Software Command Interface (GGSCI) is the command interface between users and Oracle GoldenGate functional components. For Oracle GoldenGate release 11.2.1 and later, commands are used to set up and control the interface between Oracle GoldenGate and monitoring by Oracle GoldenGate Monitor Server or Oracle Enterprise Manager.

> **Note:** Some commands or options in this section are marked as valid only for Oracle GoldenGate 11.2.1.0.7 and later; the others are valid for 11.2.1 and later.

7.2.1 CREATE DATASTORE

Use CREATE DATASTORE to create an Oracle GoldenGate Monitor data store in the Oracle GoldenGate installation directory. The data store holds monitoring information supplied by the Oracle GoldenGate Extract, Replicat, and Manager processes. This is a required step to use monitoring.

Syntax

```
CREATE DATASTORE [ MMAP | SHM [ID number] ]
```

MMAP indicates that the data store should use memory mapped files for interprocess communications. This is the default for Windows platforms. The MMAP option is valid with Oracle GoldenGate release 11.2.1.0.7 and later.

> **Note:** Do not use MMAP if you are running on a shared network file system such as Network File System (NFS), ASM Cluster File System (ACFS), or an Oracle Database File System (DBFS).

SHM indicates that the data store should use System V shared memory for interprocess communications. This is the default for non-Windows platforms. The optional ID specifies that *number* should be used as the System V shared memory key. If ID *number* is not entered, a default key will be assigned starting with 1000 and incrementing by 1 for each assignment. The SHM option is not available on Windows platforms and is valid only with Oracle GoldenGate release 11.2.1.0.7 and later.

Examples

```
CREATE DATASTORE
CREATE DATASTORE MMAP
CREATE DATASTORE SHM
CREATE DATASTORE SHM ID 1000
```

7.2.2 ALTER DATASTORE

Use ALTER DATASTORE to change the memory model used for interprocess communications by the Oracle GoldenGate Monitor data store. Before using this command, stop all Oracle GoldenGate processes, including Manager. This command is valid with Oracle GoldenGate release 11.2.1.0.7 and later.

Syntax

```
ALTER DATASTORE [ MMAP | SHM [ID number] ]
```

MMAP indicates that the data store should use memory mapped files for interprocess communications.

SHM indicates that the data store should use System V shared memory for interprocess communications. The optional ID specfices that *number* should be used as the System V shared memory key. If ID *number* is not entered, the key will be selected by GGSCI. This option is not available on Windows platforms.

Examples

```
ALTER DATASTORE MMAP
ALTER DATASTORE SHM
ALTER DATASTORE SHM ID 1000
```

7.2.3 DELETE DATASTORE

Use the DELETE DATASTORE command to remove the Oracle GoldenGate Monitor data store from the Oracle GoldenGate installation directory. Before using this command, stop all Oracle GoldenGate processes, including Manager. This command is valid with Oracle GoldenGate release 11.2.1.0.7 and later.

Syntax

```
DELETE DATASTORE [!]
```

! (exclamation point character) bypasses the prompt that confirms the intent to remove the data store.

Examples

```
DELETE DATASTORE
DELETE DATASTORE !
```

7.2.4 INFO DATASTORE

Use the INFO DATASTORE command to display information about the memory model that is being used for interprocess communications by the Oracle GoldenGate Monitor data store. This command is valid with Oracle GoldenGate release 11.2.1.0.7 and later.

Syntax

```
INFO DATASTORE
```

7.2.5 REPAIR DATASTORE

Use the REPAIR DATASTORE command to repair the data store for the Oracle GoldenGate installation. It checks that all Extract and Replicat processes are registered and attempts to resolve any internal consistency errors. Use REPAIR DATASTORE to apply required updates when upgrading from a previous version of the data store.

Before using this command, stop all Oracle GoldenGate process, including Manager.

Syntax

```
REPAIR DATASTORE
```

7.2.6 INFO JAGENT

Use INFO JAGENT to display whether or not the Jagent is running.

Syntax

```
INFO JAGENT
```

Example

The INFO request will respond that the Oracle GoldenGate agent is running or is down.

```
INFO JAGENT
```

```
JAgent is DOWN!
```

7.2.7 START JAGENT

Use START JAGENT to start the agent process. To confirm that it has started, use the INFO JAGENT or STATUS JAGENT command.

> **Note:** You do not need to restart an agent process to detect added or deleted processes.

Syntax

```
START JAGENT
```

Example

```
START JAGENT
```

```
GGCMD JAGENT started.
```

7.2.8 INFO JAGENT

Use `STATUS JAGENT` to determine whether or not the agent is running.

Syntax

```
STATUS JAGENT
```

Example

```
STATUS JAGENT

JAgent is running.
```

7.2.9 STOP JAGENT

Use `STOP JAGENT` to stop the agent process. To confirm that it has stopped, use the `INFO JAGENT` or `STATUS JAGENT` command.

Syntax

```
STOP JAGENT [!]
```

Where:

`!`(exclamation point) bypasses the prompt to confirm the request to stop the agent.

8 Properties

This chapter describes property files delivered to the cfg subdirectory when Oracle GoldenGate Monitor and Oracle GoldenGate core are installed. These files contain settings that control the monitoring process. Some property values are preset based on the release of the software and some are set by the installer based on user entries.

This chapter includes the following sections:

- Section 8.1, "Monitor Server Properties"
- Section 8.2, "Agent Properties"

8.1 Monitor Server Properties

The monitor.properties file describes the characteristics of Oracle GoldenGate Monitor processing. It includes properties to define the relationship with the JMX server, types of alert notifications to be used, and the timing for connection attempts. Many of these property values are initially set based on user entries during installation.

> **Note:** After you change a property value, you must restart Monitor Server to activate it.

8.1.1 Restricted Properties

Certain Oracle GoldenGate Monitor properties can cause the system to malfunction if changed. These properties are designated as *restricted*. You should *not* change the preset values for restricted properties.

Restricted properties include:

```
monitor.jmx.internal.mbeans.enabled
monitor.supported.agent.metadata.version
monitor.jpa.connection.driver_class
monitor.jpa.connection.url
eclipselink.target-database
eclipselink.weaving
eclipselink.ddl-generation
```

8.1.2 JMX Server Properties

These properties enable the JMX server, identify the user name, and register the name and port of the JMX server host.

8.1.2.1 monitor.jmx.server.enabled

Use `monitor.jmx.server.enabled` to enable or disable the JMX server. The JMX server must be enabled to allow the Jagent to register with the Oracle GoldenGate Monitor Server. The value is initially set to `true` to allow the agent to register.

Default

```
true
```

Syntax

```
monitor.jmx.server.enabled={true | false}
```

8.1.2.2 monitor.jmx.server.host

Use `monitor.jmx.server.host` to specify the computer name of the Oracle GoldenGate Monitor installation. Set this to the fully qualified host name of the server for the Oracle GoldenGate Monitor installation. This must match the entry for monitor.server in the agent's Config.properties file. The value is initially set by the installer based on user entries.

Syntax

```
monitor.jmx.server.host=host_name
```

8.1.2.3 monitor.jmx.server.port

Use `monitor.jmx.server.port` to specify the JMX server port number. The value is initially set by the installer based on user entries.

Syntax

```
monitor.jmx.server.port=port_number
```

8.1.2.4 monitor.jmx.server.user

Use `monitor.jmx.server.user` to specify the user name to use when communicating with the JMX server. The value is initially set by the installer based on user entries.

Syntax

```
monitor.jmx.server.user=user_name
```

8.1.3 Communication

You must select whether the communication will use Secure Sockets Layer (SSL) or not. If you use SSL, you need to identify the storage for the keys and certificates.

8.1.3.1 monitor.ssl

Use `monitor.ssl` to specify whether or not the communication from Oracle GoldenGate Monitor will use SSL.

Note: The `monitor.ssl` setting for Oracle GoldenGate Monitor Server must match the `agent.ssl` property settings in the `Configuration.properties` file of all agents that communicate with the server.

Default

```
false
```

Syntax

```
monitor.ssl=[true | false]
```

8.1.3.2 monitor.keystore.file

Use `monitor.keystore.file` to identify the file that stores the key pairs and certificates used for SSL authentication.

Default

```
monitorKeyStore
```

Syntax

```
monitor.keystore.file=keystore_filename
```

8.1.3.3 monitor. truststore.file

Use `monitor.truststore.file` to store trusted certication authority (CA) certificates that verify the identity of other clients or servers.

Defalt

```
jagentKeyStore
```

Syntax

```
monitor.truststore.file=truststore_filename
```

8.1.4 Alert Notification

These properties enable or disable the types of alerts and store information needed for communication to the e-mail server.

8.1.4.1 monitor.smtp.from

Use `monitor.smtp.from` to specify the sender name for Oracle GoldenGate Monitor communications generated from the e-mail server. The value is initially set by the installer based on user entries.

Syntax

```
monitor.smtp.from=sender_name
```

8.1.4.2 monitor.smtp.host

Use `monitor.smtp.host` to specify the host name for the e-mail server. The value is initially set by the installer based on user entries.

Syntax

```
monitor.smtp.host=email_host_name
```

8.1.4.3 monitor.smtp.port

Use `monitor.smtp.port` to specify the port for sending e-mail alerts. The value is initially set by the installer based on user entries.

Syntax

```
monitor.smtp.port=port_number
```

8.1.4.4 monitor.smtp.secure

Use `monitor.smtp.secure` to specify whether the SMTP server is in secure mode. The value is initially set by the installer based on user entries.

Syntax

```
monitor.smtp.secure={true | false}
```

8.1.4.5 monitor.smtp.user

If the SMTP server is in secure mode, specify the user authorized to log in. The value is initially set by the installer based on user entries.

Syntax

```
monitor.smtp.user=user_name
```

8.1.4.6 monitor.smtp.alerts.enabled

Use `monitor.smtp.alerts.enabled` to specify whether e-mail alerts are enabled. The value is initially set by the installer based on user entries.

Syntax

```
monitor.smtp.alerts.enabled={true | false}
```

8.1.4.7 monitor.snmp.alerts.enabled

Use `monitor.snmp.alerts.enabled` to specify whether SNMP alerts are enabled. The value is initially set by the installer based on user entries.

Syntax

```
monitor.snmp.alerts.enabled={true | false}
```

8.1.4.8 monitor.cli.alerts.enabled

Use `monitor.cli.alerts.enabled` to specify whether command-line interface alerts are enabled. The value is initially set by the installer based on user entries.

Syntax

```
monitor.cli.alerts.enabled={true | false}
```

8.1.5 Connection Properties

These properties define characteristics of the connections. They can be changed, but it is recommended that you first consult with Oracle Support. For more information go to `http://support.oracle.com`.

8.1.5.1 monitor.db.connection.initial_size

Use `monitor.db.connection.initial_size` to set the Initial number of database connections in the database connection pool. The default is 5.

Syntax

```
monitor.db.connection.initial_size=5
```

8.1.5.2 `monitor.db.connection.max_active`

Use `monitor.db.connection.max_active` to set the maximum number of database connection can be created. The default is 50.

Syntax

```
monitor.db.connection.max_active=50
```

8.1.5.3 `monitor.db.connection.max_idle`

Use `monitor.db.connection.max_idle` to set the maximum number of database connection that can be idle. The default is 5.

Syntax

```
monitor.db.connection.max_idle=5
```

8.1.5.4 `monitor.default_agent_connection.max_attempts`

Use `monitor.default_agent_connection.max_attempts` to specify the number of unsuccessful connections before the process will stop attempting to connect. An entry of 0 or a negative number specifies no limit on the number of times the connection should be tried. The value is initially set to 10.

Syntax

```
monitor.default_agent_connection.max_attempts=number
```

8.1.5.5 `monitor.default_agent_connection.interval`

Use `monitor.default_agent_connection.interval` to specify the number of seconds to wait between each unsuccessful attempt to connect. The value is initially set to 30.

Syntax

```
monitor.default_agent_connection.interval=seconds
```

8.1.5.6 `monitor.default_agent_connection.reconnect_interval`

Use `monitor.default_agent_connection.reconnect_interval` to specify the number of seconds to wait after an existing connection is broken before an attempt is made to reconnect. The value is initially set to 5.

Syntax

```
monitor.default_agent_connection.reconnect_interval=seconds
```

8.1.6 Repository properties

The repository database is selected during the installation of Oracle GoldenGate Monitor. Contact Oracle Support if you need to change the repository after installation. For more information on contacting support, go to `http://support.oracle.com`.

8.1.6.1 `monitor.jpa.connection.user`

Use `monitor.jpa.connection.user` to specify the repository database user name. The value is initially set by the installer based on user entries.

Syntax

```
monitor.jpa.connection.user=user_name
```

8.1.7 Configuration Management Properties

These properties set the timeout value and the threshold number of events to trigger processing.

8.1.7.1 monitor.cm.event.timeout

Use monitor.cm.event.timeout to specify the time in milliseconds for the solution discovery process to wait between inquiries for new agents registered with the Oracle GoldenGate Monitor Server. The solution discovery process starts if a new agent is found. The value is initially set to 2000 milliseconds.

Syntax

```
monitor.cm.event.timeout=milliseconds
```

8.1.7.2 monitor.cm.event.max.size

Use `monitor.cm.event.max.size`to specify the threshold number of events that triggers the solution discovery process not to wait `monitor.cm.event.timeout` seconds before processing the remaining events. The value is initially set to 1000 events.

Syntax

```
monitor.cm.event.max.size=number_events
```

8.1.7.3 monitor.events.dispatcher.threads_size

Use `monitor.events.dispatcher.threads_size` to specify the number of threads that will be used by the events dispatcher process. The value is initially set to 30.

Syntax

```
monitor.events.dispatcher.threads_size=number
```

8.2 Agent Properties

The `Config.properties` file configures an Oracle GoldenGate agent for communication with the Oracle GoldenGate Monitor Server or Oracle Enterprise Manager (Oracle GoldenGate release 11.2.1 or later). It contains preset properties delivered with the Oracle GoldenGate core installation. Values such as host server names, ports, and users must be reset to valid values for your system.

Some of the properties have default values that are used when a value for the property is not defined in the property file. Default values are not substituted for invalid entries. A message is written to the `jagent.log` and the `ggserr.log` when a default value is used.

Note: After you change a property value, you must restart the agent to activate it.

8.2.1 Agent Definition

You must set the type of agent that will be used to monitor the Oracle GoldenGate instance.

8.2.1.1 agent.type.enabled

Use agent.type.enabled to specify whether monitoring will be done in Oracle GoldenGate Monitor or Oracle Enterprise Manager.

The Oracle GoldenGate agent polls Manager at configurable intervals to collect monitoring points data.

- Setting agent.type.enabled to OGGMON causes the Oracle GoldenGate agent to send the data to the Oracle GoldenGate Monitor Server.
- Setting agent.type.enabled to OEM causes the Oracle GoldenGate agent to supply monitoring points data when polled by an Oracle Enterprise Manager agent or ODI.

This setting is also required when integrating with Oracle Data Integrator. If you are using both GoldenGate Monitor and Oracle Data Integrator at the same time, you will need to set up two agents, one with agent.type.enabled set to Oracle GoldenGate Monitor and one set to OEM.

When agent.type.enabled is set to OEM, the Oracle Enterprise Manager agent will connect to the Oracle GoldenGate agent through the Remote Method Invocation (RMI) connector, so you must enter the jagent.rmi.port.

Note: This property is valid for the 11.2.1 release of Oracle GoldenGate and later.

Syntax

```
agent.type.enabled={OGGMON | OEM}
```

8.2.1.2 jagent.rmi.port

Use jagent.rmi.port to specify the port number for the RMI connector. This is the port that will be used by the Enterprise Manager agent when connecting to the Oracle GoldenGate agent.

This property is required when agent.type.enabled is set to OEM.

Note: This property is valid for the 11.2.1 release of Oracle GoldenGate and later.

Default

```
5559
```

Syntax

```
agent.rmi.port=port_number
```

8.2.2 Communication

You must select whether the communication will use Secure Sockets Layer (SSL) or not. If you use SSL, you need to identify the storage for the keys and certificates.

Note: Oracle GoldenGate agents running on IBM z/OS can only use SSL . Additionally, all agents communicating with a GoldenGate Monitor Server must all use the same connection protocol.

8.2.2.1 jagent.ssl

Use jagent.ssl to specify whether or not the communication from the agent will use SSL.

> **Note:** If the jagent.backward.compatility property is set to true, Oracle GoldenGate Monitor assumes SSL is not enabled and ignores the setting for jagent.ssl.

> **Note:** The jagent.ssl setting for all agents that communicate with Oracle GoldenGate Monitor Server must match the server's setting for the monitor.ssl property in the monitor.properties file.

Default

```
false
```

Syntax

```
jagent.ssl=[true | false]
```

8.2.2.2 jagent.keystore.file

Use jgent.keystore.file to identify the file that stores the key pairs and certificates used for SSL authentication.

Default

```
jagentKeyStore
```

Syntax

```
jagent.keystore.file=keystore_filename
```

8.2.2.3 jagent.truststore.file

Use jagent.truststore.file to store trusted certification authority (CA) certificates that verify the identity of other clients or servers.

Default

```
jagentKeyStore
```

Syntax

```
jagent.truststore.file=truststore_filename
```

8.2.3 Monitoring Targets

You need to identify the names and ports for the computers where the Oracle GoldenGate Manager and Agent reside. You also set backward compatibility and define the agent credentials.

8.2.3.1 jagent.host

Use jagent.host to specify the host name of the computer where the Oracle GoldenGate agent is running. This is the host name of the Oracle GoldenGate instance being monitored and should be the same name used in the RMTHOST parameter by any remote GoldenGate Extract processes that connect to this Oracle GoldenGate Monitor instance. This will ensure that Solution Discovery works properly.

This property is required. If a valid value is not entered, Manager writes an error to `ggserr.log` during start up and the agent is not initialized successfully.

Syntax

```
jagent.host=ogg_host_name
```

Note: If a remote trail is specified in the Extract parameter file using `RMTTRAIL`, then the host name specified for the `RMTHOST` parameter must match the value set for the Java `jagent.host` entry. When you have a remote Extract connecting to the local Oracle GoldenGate Monitor instance running the Oracle GoldenGate agent, the fully qualified host name specified in the parameter file must be the same as the fully qualified name used for `jagent.host`.

8.2.3.2 `jagent.jmx.port`

Use `jagent.jmx.port` to specify the JMX port of the agent.

Default

```
5555
```

Syntax

```
jagent.jmx.port=port_number
```

8.2.3.3 `mgr.host`

Use `mgr.host` to specify the name or IP address of the computer where the Oracle GoldenGate Manager is running. Together, `mgr.host` and `mgr.port` identify the Oracle GoldenGate instance to the Oracle GoldenGate agent.

If this property is not entered, the system will assume that the agent is local to the Oracle GoldenGate instance and determine the value by default.

Note: This property is not used for 11.2.1 and earlier releases since the agent must be local to the Oracle GoldenGate instance.

Syntax

```
mgr.host=ogg_host_name
```

8.2.3.4 `mgr.port`

Use `mgr.port` to specify the port of the Oracle GoldenGate Manager. If this property is not entered, the system will assume that the agent is local to the Oracle GoldenGate instance and determine the value by default.

Note: This property is not used for 11.2.1 and earlier releases since the agent must be local to the Oracle GoldenGate instance.

Syntax

```
mgr.port=manager_port
```

8.2.3.5 monitor.host

Use `monitor.host` to specify the host computer name of the Oracle GoldenGate Monitor Server installation. Use the fully qualified host name. This must match the entry for `monitor.jmx.server.host` in the `monitor.properties` file.

This property is required. If a valid value is not entered, Manager writes an error to `ggserr.log` during start up and the agent is not initialized successfully.

Syntax

```
monitor.host=monitor_host_name
```

8.2.3.6 monitor.jmx.port

Use `monitor.jmx.port` to specify the JMX port of the Oracle GoldenGate Monitor Server installation. Initially set this to the value entered for the port on the JMX Server Configuration screen during the Oracle GoldenGate Monitor Server installation.

Default

```
5502
```

Syntax

```
monitor.jmx.port=port_number
```

8.2.3.7 monitor.jmx.username

Use `monitor.jmx.username` to specify the user name for the JMX connection to the Oracle GoldenGate Monitor Server. Initially set this to the value entered for the user name on the JMX Server Configuration screen during the Oracle GoldenGate Monitor installation.

This property is required. If a valid value is not entered, Manager writes an error to `ggserr.log` during start up and the agent is not initialized successfully.

Syntax

```
monitor.jmx.username=user_name
```

8.2.3.8 jagent.username

Use `jagent.username` to specify the agent user name for the JMX connection to the Oracle GoldenGate agent. When the agent registers, it passes this name to the Oracle GoldenGate Monitor Server.

This property is required. If a valid value is not entered, Manager writes an error to `ggserr.log` during start up and the agent is not initialized successfully.

Syntax

```
jagent.username=user_name
```

8.2.3.9 jagent.backward.compatibility

Set `jagent.backward.compatibility` to `true` to activate backward compatibility that allows monitoring of Oracle GoldenGate 11.1.1.1.1 instances. The value is initially set to `false`.

> **Note:** This property applies only when `agent.type.enabled=OGGMON`. It is ignored when `agent.type.enabled=OEM`.

`jagent.backward.compatibility` is valid for all supported platforms except IBM z/OS.

If `jagent.backward.compatibility` is set to `true`, Oracle GoldenGate Monitor assumes SSL is not enabled and `jagent.ssl` is ignored.

Syntax

```
jagent.backward.compatibility={true | false}
```

8.2.4 Polling Properties

You can set the polling intervals. These polling interval properties default to the indicated default number of seconds if nothing is entered. An error message is generated if the entered number of seconds is negative or greater than 2147483647.

8.2.4.1 `interval.regular`

Use `interval.regular` to specify the polling interval used for monitoring points in the `Regular Default Polling Group`. The value is in seconds.

Default

60 seconds

Syntax

```
interval.regular=seconds
```

8.2.4.2 `interval.quick`

Use `interval.quick` to specify the polling interval used for monitoring points in the `Quick Default Polling Group`. The value is in seconds.

Default

30 seconds

Syntax

```
interval.quick=seconds
```

8.2.4.3 `reg.retry.interval`

Use `reg.retry.interval` to specify the interval to wait before retrying an initial agent registration when an exception occurs.

Default

60 seconds

Syntax

```
reg.retry.interval=seconds
```

8.2.4.4 `instance.query.initial.interval`

Use `instance.query.initial.interval` to specify the interval that Jagent will wait to register if Manager is the only running process. If there are still no other processes after this interval, the agent will proceed with the registration.

Default

15 seconds

Syntax

```
instance.query.initial.interval=seconds
```

8.2.4.5 `incremental.registration.quiet.interval`

Use `incremental.registration.quiet.interval` to specify the interval that the agent will wait before registration of a new process.

Default

5 seconds

Syntax

```
incremental.registration.quiet.interval=seconds
```

8.2.4.6 `maximum.message.retrieval`

Use `maximum.message.retrieval` to specify the maximum number of messages to retrieve from the core Oracle GoldenGate instance when the Oracle GoldenGate agent starts up.

Default

500 messages

Syntax

```
maximum.message.retrieval=number
```

8.2.4.7 `message.polling.interval`

Use `message.polling.interval` to set the interval for agent to poll `ggserr.log` for new messages.

Default

5 seconds

Syntax

```
message.polling.interval=seconds
```

8.2.4.8 `status.polling.interval`

Use `status.polling.interval` to set the interval for the agent to poll for the status of new and existing processes.

Default

5 seconds

Syntax

```
status.polling.interval=seconds
```

8.2.4.8.1 Monitor Server Properties The `monitor.properties` file describes the characteristics of Oracle GoldenGate Monitor processing. It includes properties to define the relationship with the JMX server, types of alert notifications to be used, and the timing for connection attempts. Many of these property values are initially set based on user entries during installation.