Oracle® Business Intelligence Enterprise Edition Deployment Guide

Version 10.1.3.2

December 2006

ORACLE®

Part Number: B40058-01

Contents

Chapter 1: What's New in This Release

Chapter 2: Overview of Oracle BI Enterprise Deployment

Chapter 3: Clustering, Load Balancing, and Failover in Oracle Business Intelligence

Chapter 4: Deploying Oracle Business Intelligence for High Availability

Chapter 9: Other Deployment-Related Topics

Chapter 10: Integrating Oracle Internet Directory With Oracle Business Intelligence

Chapter 11: Enabling Oracle Single Sign-On for Oracle Business Intelligence

1 What's New in This Release

Oracle Business Intelligence Enterprise Edition consists of components that were formerly available from Siebel Systems as Siebel Business Analytics Platform, with a number of significant enhancements.

The *Oracle Business Intelligence Enterprise Edition Deployment Guide* is part of the documentation set for Oracle Business Intelligence Enterprise Edition. This guide contains information on how to effectively plan and perform the installation and configuration of the Oracle Business Intelligence platform (also called Oracle BI) under various deployment options.

Oracle recommends reading the Oracle Business Intelligence Enterprise Edition Release Notes before installing, using, or upgrading the Oracle BI infrastructure. The Oracle Business Intelligence Enterprise Edition Release Notes are available at the following locations:

- On the Oracle Business Intelligence Enterprise Edition CD-ROM.
- On the Oracle Technology Network at http://www.oracle.com/technology/documentation/bi_ee.html (to register for a free account on the Oracle Technology Network, go to http://www.oracle.com/technology/about/index.html).

What's New in *Oracle Business Intelligence Enterprise Edition Deployment Guide*, Version 10.1.3.2

Table 1 lists changes described in this version of the documentation to support release 10.1.3.2 of the software.

Table 1. New Product Features in *Oracle Business Intelligence Enterprise Edition Deployment Guide*, Version 10.1.3.2

Topic	Description
All topics	This guide is new for Oracle Business Intelligence 10.1.3.2.

2 Overview of Oracle BI Enterprise Deployment

The Oracle Business Intelligence infrastructure components can be deployed across your enterprise.

The Oracle Business Intelligence components can be installed in a distributed architecture in a multi-server environment. Oracle Business Intelligence provides for high availability through the use of native clustering, failover, and load balancing capabilities. The Oracle Business Intelligence SSL Everywhere feature allows you to secure communications across all Oracle Business Intelligence components. Oracle Business Intelligence allows you to enable Single Sign-On for your deployment. The Oracle Business Intelligence infrastructure allows integration with the Oracle Middleware products such as Oracle Internet Directory and Oracle Single Sign-On.

The Oracle Business Intelligence components consist of:

- Oracle Business Intelligence Presentation Services

  The Oracle Business Intelligence Presentation Services provides the framework and interface for presentation of Business Intelligence data to web clients. It maintains a Presentation Catalog service on the file system for the customization of this presentation framework. It is a stand-alone process and integrates with the Oracle Business Intelligence Presentation Services Plug-in from which it receives web client requests. It communicates with the Oracle Business Intelligence Server using ODBC over TCP/IP.

- Oracle Business Intelligence Server

  The Oracle Business Intelligence Server is a stand-alone process that maintains the logical data model which it provides to BI Presentation Services via ODBC. Metadata is maintained for the data model in a local proprietary file called the repository file (rpd). On the back-end, the BI Server connects to customer data stores via data source adaptors.

- Oracle Business Intelligence Scheduler

  The Oracle Business Intelligence Scheduler is an extensible scheduling application for scheduling reports to be delivered to users at specified times. It is the engine behind the Oracle Business Intelligence Delivers feature.

- Oracle Business Intelligence Publisher

  The Oracle Business Intelligence Publisher generates highly-formatted, pixel-perfect enterprise reports.

- Oracle Business Intelligence Java Host

  The Oracle Business Intelligence Javahost provides services to BI Presentation Services for Charts, Gauges and PDFs. The services are provided based on request-response model.

- Oracle Business Intelligence Presentation Services Plug-in

  The Oracle Business Intelligence Presentation Services Plug-in is the entry point for web client requests to BI Presentation Services. There are two types of BI Presentation Services Plug-ins. For Oracle Business Intelligence that is serviced by J2EE application servers, the BI Presentation Services Plug-in is a Java Servlet. For Oracle Business Intelligence where the web server used is Microsoft Internet Information Services (IIS), the Oracle BI Presentation Services Plug-in is an ISAPI Plug-in.

NOTE: This guide assumes that you are familiar with the components of Oracle Business Intelligence infrastructure, their functionality, and the process to install and configure these components.

This guide describes how to deploy Oracle Business Intelligence in an enterprise. The enterprise deployment options described in this guide are shown in Table 2 on page 12.

Table 2. Oracle BI Deployment Options

Deployment Option	See Chapter ...
High Availability with Clustering, Load Balancing, and Failover of Oracle BI Components	Chapter 3, “Clustering, Load Balancing, and Failover in Oracle Business Intelligence” Chapter 4, “Deploying Oracle Business Intelligence for High Availability”
Data Security - Enabling Secure Communications across Oracle Business Intelligence Components	Chapter 6, “Enabling Secure Communication in Oracle Business Intelligence”
Enabling Single Sign-On	Chapter 8, “Implementing Single Sign-On Products With Oracle Business Intelligence” Chapter 11, “Enabling Oracle Single Sign-On for Oracle Business Intelligence”
Security Firewall	Chapter 9, “Other Deployment-Related Topics”
Integrating with Oracle Middleware Products	Chapter 10, “Integrating Oracle Internet Directory With Oracle Business Intelligence” Chapter 6, “Enabling Secure Communication in Oracle Business Intelligence””

3 Clustering, Load Balancing, and Failover in Oracle Business Intelligence

This chapter describes the clustering, load balancing, and failover capabilities offered by Oracle Business Intelligence.

The Oracle BI components are supported in a many-to-many architecture. End user web requests can be directed to one of many BI Presentation Services servers. In turn, each BI Presentation Services can take advantage of the availability of multiple BI Servers. The BI Cluster Server feature allows multiple BI Servers to be deployed. BI Schedulers participate in the cluster in an active-passive configuration. A Cluster Controller serves as the entry point to the clustered servers. The server metadata is contained in the repository file (.RPD) that is local to each BI Server. One BI Server is designated as a Master. Online changes to the RPD file are made on the Master BI Server and these changes are replicated to other members of the cluster. A cluster-aware cache capability offers support for a common query cache that is visible to all BI Servers in the cluster.

Clustering capability for the Presentation layer allows for the deployment of a multi-server environment to better manage large volumes of users and to provide high availability. Multiple BI Presentation Services instances in the cluster can either share a common Presentation Catalog on a network storage device, or the catalog may be replicated across each BI Presentation Services instance. Native load balancing and failover capabilities are offered for the components of the Presentation layer.

Oracle BI Cluster Server Components

This section describes the components that comprise the BI Cluster Server feature.

- “Oracle Business Intelligence Cluster Controller” on page 13
- “Clustered BI Servers” on page 14
- “Master BI Server” on page 14
- “BI Scheduler” on page 14
- “Cluster Manager” on page 14

Oracle Business Intelligence Cluster Controller

The BI Cluster Controller is a process that serves as the first point of contact for new requests from BI Presentation Services and other clients. The Cluster Controller determines which BI Server in the cluster to direct the request to based on BI Server availability and load. It monitors the operation of servers in the cluster, including the BI Scheduler instances. The Cluster Controller is deployed in active-passive configuration:

- Primary Cluster Controller

  This controller is the active cluster controller.

- Secondary Cluster Controller

  This controller is the secondary cluster controller. It assumes the role of the primary cluster controller if the primary controller is unavailable.

Clustered BI Servers

The BI Cluster Server feature supports up to 16 BI Servers in a network domain to act as a single server. BI Servers in the cluster share requests from multiple Oracle BI clients.

Master BI Server

A clustered Oracle Business Intelligence Server is designated as the Master BI Server. The Oracle Business Intelligence Administration Tool connects to the master BI Server for online repository changes.

BI Scheduler

BI Scheduler instances participate in the Cluster Server feature in active-passive mode. The active BI Scheduler instance processes jobs and executes iBot requests. The inactive BI Schedulers remain idle and do not process jobs until called on to take over in the event of an active Scheduler failure

Cluster Manager

The Cluster Manager is available in the Administration Tool when a repository is open in online mode. The Cluster Manager enables or quiesces Oracle BI Server clustered instances, and activates Oracle BI Scheduler clustered instances.

For more information on the Cluster Manager, refer to the *Oracle Business Intelligence Server Administration Guide*.

NOTE: All components of the BI Cluster Server feature must reside on the same Local Area Network (LAN). Multi-NIC is not supported for clustered deployments.

Clustering of Oracle BI Components of the Presentation Layer

Multiple BI Presentation Services instances may be installed and configured to participate in the Oracle BI deployment to service a large volume of users and to provide for high availability. Note that the multiple BI Presentation Services are not controlled by the BI Cluster Controller. The BI Presentation Services instances in the Oracle BI deployment may either share a common Presentation Catalog on a shared file system or the Presentation Catalog may be replicated across the BI Presentation Services instances.

NOTE: Refer to the *Oracle Business Intelligence Presentation Services Administration Guide* for detailed information on replicating the Presentation Catalog.

The Oracle BI Javahost component provides services to BI Presentation Services for Charts, Gauges and PDFs based on a request-response model. BI Javahost is installed along with each instance of BI Presentation Services, and by default BI Presentation Services communicates requests to its local BI Javahost instance. The clustering capability of the Presentation layer offers the ability to cluster the BI Javahost instances installed along with each instance of BI Presentation Services so that requests to BI Javahost are load balanced to the cluster.

Native load balancing and failover capability is provided for the components of the Presentation layer. This capability supports load balancing and failover for the following component connections:

- BI Presentation Services Plug-in to multiple BI Presentation Services instances
- BI Presentation Services to multiple BI Javahost instances
- BI Scheduler to multiple BI Presentation Services instances
- BI Scheduler to multiple BI Javahost instances

Communication Between Oracle BI Components in a Clustered Environment

This section describes the lines of communication between the Oracle BI components deployed in a clustered environment. Figure 1 on page 16 depicts Oracle BI components deployed in a clustered environment. The following components are described:

- "Web Server" on page 16
- "BI Presentation Services Plug-In" on page 17
- "BI Presentation Services" on page 17
- "BI Cluster Controller" on page 17
- "Oracle BI Servers" on page 18
- "BI Scheduler" on page 18
- "BI Javahost" on page 19

v

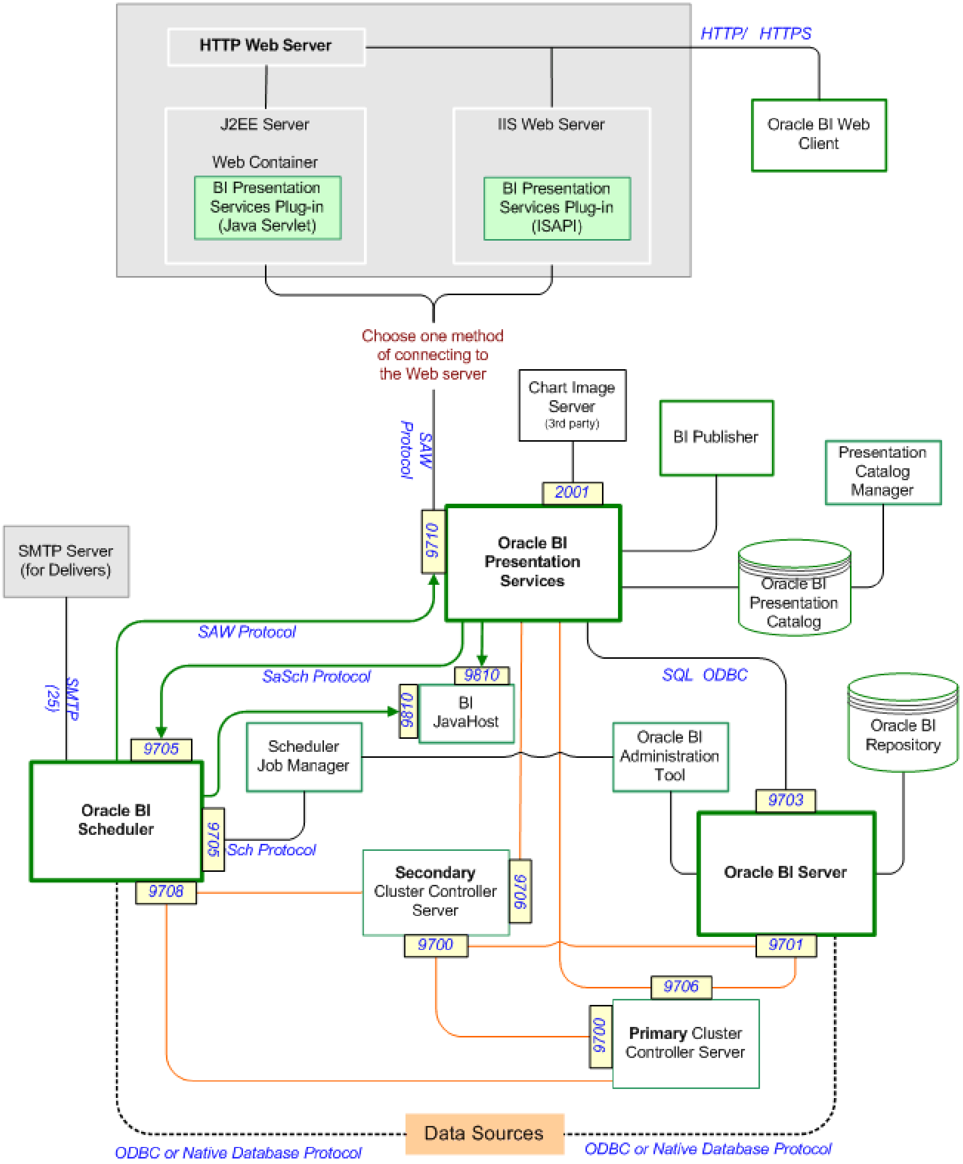
HTTP Web Server
J2EE Server
Web Container
BI Presentation Services Plug-in (Java Servlet)
IIS Web Server
BI Presentation Services Plug-in (ISAPI)
HTTP/ HTTPS
Oracle BI Web Client
Choose one method of connecting to the Web server
SAW Protocol
Chart Image Server (3rd party)
2001
BI Publisher
Presentation Catalog Manager
9710
Oracle BI Presentation Services
Oracle BI Presentation Catalog
SMTP Server (for Delivers)
SMTP (25)
SAW Protocol
SaSch Protocol
9810
BI JavaHost
9810
SQL ODBC
Oracle BI Repository
9705
Scheduler Job Manager
Oracle BI Administration Tool
Oracle BI Scheduler
9705
Sch Protocol
9703
Oracle BI Server
Secondary Cluster Controller Server
9706
9708
9700
9701
9706
9700
Primary Cluster Controller Server
Data Sources
ODBC or Native Database Protocol
ODBC or Native Database Protocol

Figure 1. Oracle BI Components in a Clustered Environment

Web Server

Load balanced Web Servers are the entry points for web client requests to Oracle Business Intelligence.

- For Internet Information Services (IIS), the BI Presentation Services Plug-in (ISAPI Plug-in) is deployed on all instances of the web server.

- In the case of J2EE Application Servers, such as Oracle Application Server, multiple HTTP Servers can be load balanced and serve as the entry point for Oracle BI session requests. Multiple J2EE containers with BI Presentation Services Plug-in (Java Servlet) deployed in them serve to direct these requests to BI Presentation Services instances.

For a list of supported Web Servers and J2EE Application Servers, refer to the *Oracle Business Intelligence System Requirements and Supported Platform Guide*.

BI Presentation Services Plug-In

BI Presentation Services Plug-ins route session requests to BI Presentation Services instances using native protocol. The connections are load balanced using native load balancing capability.

BI Presentation Services

BI Presentation Services receives requests from BI Presentation Services Plug-in on the RPC Listener port (9710) set in the instanceconfig.xml configuration file. Although an initial user session request can go to any BI Presentation Services in the cluster, each user is then bound to a specific BI Presentation Services instance.

- Communication with BI Servers

  For the processing of end-user requests, BI Presentation Services must communicate with the BI Servers. In a clustered environment, the first point of contact to the BI Servers is through the BI Cluster Controller. BI Presentation Services communicates with the Cluster Controller via the BI ODBC data source that is configured for the clustered environment to identify the Primary and Secondary Cluster Controllers and the ports they listen on. BI Presentation Services obtains from the Cluster Controller the BI Server instance to connect to. The connection to the BI Server is established via the BI ODBC, and subsequent requests in the same session go directly from the BI Presentation Services to this assigned BI Server. The ODBC session between BI Presentation Services and the BI Server is stateful and affinity must be maintained for the lifetime of the session.

- Communication with BI Scheduler

  BI Presentation Services must be informed that communication is to occur with a clustered Scheduler. This is specified in the instanceconfig.xml file along with the Primary and Secondary Cluster Controller host names and the ports they listen on. BI Presentation Services first contacts the Cluster Controller, which relays the active BI Scheduler instance to BI Presentation Services. BI Presentation Services then establishes a session with the Scheduler instance.

- Communication with BI Javahost

  Each BI Presentation Services instance is configured to communicate with multiple BI Javahost instances in a cluster. The requests to the BI Javahost instances are load balanced using native load balancing capability.

BI Cluster Controller

The Cluster Controller is the first point of contact for a new request and session from BI Presentation Services and other clients. The Primary and Secondary Cluster Controllers listen on CLIENT_CONTROLLER_PORT (9706), which is set in the NQClusterConfig.INI file.

The NQSClusterConfig.INI file contains the list of BI Servers participating in the cluster. The Cluster Controller connects to BI Servers on the MONITOR_SERVER_PORT (9701) that is configured in the same file. Each BI Server listens on MONITOR_SERVER_PORT (9701) and relays the number of sessions back to the Cluster Controller. The Cluster Controller determines which BI Server in the cluster to direct a request to based on BI Server availability and load.

The Cluster Controller serves as the first point of contact for requests to BI Scheduler. A list of the Scheduler instances that participate in the cluster is configured in the NQSClusterConfig.INI file. It determines the active BI Scheduler instance to which the client then connects.

The Cluster Controllers monitor each other's life cycle on MONITOR_CONTROLLER_PORT (9700). This port is configured in the NQSClusterConfig.INI file.

Oracle BI Servers

Multiple Oracle BI Servers can be installed and configured to create a BI Server cluster. The Cluster Controller dispatches requests from clients such as Presentation Servers to an active member of this cluster. The BI Server listens on RPC_SERVICE_OR_PORT (9703) configured in the NQSConfing.INI file for client requests.

- Master BI Server

  The Master BI Server is the server that the BI Administration Tool connects to in order to perform online metadata changes in the RPD file. These metadata changes are then propagated out to the other servers.

  The Administration Tool uses a BI ODBC DSN that is configured for the clustered environment. It is directed to the Master BI Server via the Cluster Controller.

BI Scheduler

The BI Scheduler instances operate on an active-passive model. Only one BI Scheduler is active and processing requests at any one time. The BI Scheduler listens on port 9708 for Cluster Controller communication and on port 9705 for client requests. These ports are set in the configuration file for the Scheduler, namely, instanceconfig.xml in <ClusterPort> and <PortString> respectively.

- Communication With BI Presentation Services

  BI Scheduler communicates with BI Presentation Services for jobs such as iBots that deliver alerts and reports to end users.

  Since no cluster controller exists for the BI Presentation Services instances, the list of BI Presentation Services instance must be configured in <Web_Server> in the Scheduler instanceconfig.xml. Connections for each unique user session of the iBot are load balanced using native capability in round robin fashion.

- Communication With BI Javahost

  Scheduler is configured to communicate with the BI Javahost instances in the cluster. Round robin load balancing is done for Java jobs and Javahost extensions to iBots

BI Javahost

BI Javahost receives requests from BI Presentation Services and BI Scheduler on port 9810 configured in <Port> in the config.xml configuration file. Requests to multiple BI Javahost can be load balanced using the native load balancing capability.

Failover Mechanisms for Oracle BI Components

This section describes the failover process for BI components in a cluster:

- "BI Presentation Services Failure" on page 19
- "BI Server Failure" on page 19
- "Master BI Server Failure" on page 20
- "BI Scheduler Failure" on page 20
- "Cluster Controller Failure" on page 20

BI Presentation Services Failure

- Web Clients

  Although an initial user session request can go to any BI Presentation Services, each user is then bound to a specific BI Presentation Services instance. Loss of that Presentation server will disconnect the session, and an error is relayed back to the browser. Any work in progress during the loss of the server that was not saved to disk is lost. The user must re-login to establish a new connection to an available BI Presentation Services. If user login is taking place via a Single Sign-On system such as Oracle Single Sign-On (SSO) this relogin takes place automatically. The new BI Presentation Services session will create a new BI Server session.

  NOTE: When a BI Presentation Services instance fails, there is a small interval of time before the system recognizes that the instance has failed and before users are migrated to a new BI Presentation Services instance. There may be some loss of session state.

- iBots

  An error will be relayed to the BI Scheduler which will log the failure and then retry the job. The retry will establish a new connection to an available BI Presentation Services

BI Server Failure

When BI Server failure occurs, an ODBC error is sent back to the client.

- BI Presentation Services

  Each web user of Oracle BI has requests served by one BI Server. If this BI Server becomes unavailable, the end user may see an error, but a browser refresh will cause a new session to be established with an available BI Server.

- Administration Tool

  Administration Tool will relay the ODBC error when the BI Server that it is connecting to becomes unavailable, and then will close the connection. The Administrator will have to re-connect.

- iBots

  When BI Server failure occurs, the error will be relayed to the Scheduler, which logs the failure and retries the job. This will cause a connection to be established with an available BI Server.

- 3rd Party Clients

  3rd Party Clients use ODBC to connect to the BI Server. When BI Server failure occurs, the error will be relayed and the session closed and re-opened according to the ODBC standard.

Master BI Server Failure

If the Master BI Server is unavailable, online metadata changes cannot be performed. This is an administration operation and does not impact runtime availability. If the Master BI Server is permanently unavailable, one of the other Servers must be appointed as the new master. This will require reconfiguration of all the servers.

BI Scheduler Failure

The BI Scheduler is monitored and managed by the Cluster Controller. If the BI Scheduler is unavailable, the Cluster Controller will determine the next BI Scheduler instance to activate. If the previous primary Scheduler becomes available again, the primary role will not revert.

When the active BI Scheduler fails, any open client connections will not receive an error as the Scheduler protocol is stateless and will seamlessly fail over.

- iBots

  iBot executions maintain state in the Scheduler tables. When the next instance of Scheduler becomes active, it will read the state of all job instances that were in progress, and execute them. An iBot will only deliver to those recipients that it did not deliver to prior to the failure of the primary instance.

- Java, Command Line, or Script Job

  The jobs will be re-executed from the beginning with a new job instance.

  NOTE: Any job instance can be manually re-run from the Job Manager. For an iBot, this only delivers to those users that did not have successful deliveries. For example, if the mail server goes down half-way through an iBot execution, the re-run of the instance will only deliver to those recipients who did not receive email due to the mail server crash.

Cluster Controller Failure

The Cluster Controller supports detection of BI Server or BI Scheduler failures and failover for clients of failed servers.

The Cluster Controllers work on an active-passive model. All clients first attempt to connect to the Primary Cluster Controller. In the case where the Primary Cluster Controller is unavailable, clients will then connect to the Secondary Cluster Controller. The Secondary Cluster Controller then directs requests to BI Servers based on load and availability and to the active BI Scheduler instance. If the Primary later becomes available, all requests will then go to the Primary again.

The Secondary Cluster Controller monitors the session count on each BI Server just like the Primary, but does not dictate the active Scheduler unless the Primary Cluster Controller is down.

The Primary and Secondary Cluster Controllers monitor each other's life cycle. This is susceptible to a "Split-Brain" failure if the communication is down between the Cluster Controller instances, but each is up and can communicate with the other clients. In these cases, BI Servers are not effected, but the Scheduler may have two active instances at once. In rare cases, this may lead to double execution of jobs. When the line of communication comes back up, the Primary Cluster Controller will dictate to the cluster that only one Scheduler should be active. The possibility of a Split-Brain failure to occur is minimized by the fact that the Cluster components must exist on the same Local Area Network (LAN) and Multi-NIC is not supported for clustered deployments.

If both Cluster Controllers are unavailable, BI Presentation Services will return an error to any new user attempting to login. Existing sessions will not be affected.

Shared Files and Directories

The BI components deployed in a clustered environment must share certain files and directories as described below. A shared storage device such as NAS or SAN may be used.

NOTE: On Windows, the BI services must run under a domain account in order to access network shares. Do not use the LocalSystem account.

- Presentation Catalog

  The BI Presentation Services instances in a cluster share a common Presentation Catalog. The Presentation Catalog should be placed on a shared NAS or SAN device. All instances of BI Presentation Services must have read and write access to the share.

  Because the Presentation Catalog consists of a large number of heavily accessed small files, there are two important considerations for the shared file system:

  - File Limits

    The Presentation Catalog can consist of thousands of files. In many cases this may exceed file limits for shared file systems. Check the storage vendor documentation for instructions on extending the file limit.

  - Snapshots

    Backup activity such as snapshots may adversely affect the performance of Presentation Catalog files which are small, dynamic files. Ensure that snapshot activity is at a reasonable level that maximizes performance without impacting availability.

- Repository Publishing Directory

  This directory is shared by all Oracle BI Servers participating in a cluster. It holds the master copies of repositories edited in online mode. The clustered Oracle BI Servers examine this directory upon startup for any repository changes.

  The Master BI Server must have read and write access to this directory. All other BI Servers must have read access.

- Cluster-Aware Cache

  The cluster-aware cache is a query cache that is shared by all BI Servers participating in a cluster. For more information, see the topic About Cluster-Aware Cache in this chapter.

  All BI Servers must have read and write access to the global cache directory.

- Scheduler Scripts

  A network share for the Scheduler scripts must be created. The Scheduler servers must have read and write access to this share.

  The following information applies to deployments with BI server components on Linux or UNIX platforms that access the above-mentioned shared files and directories on a NAS device from Network Appliance. For environments with BI server components on Linux or UNIX that use the NTFS security style, the recommended Network Appliance Data ONTAP storage operating system version is 6.3.1 or better.

  Linux or UNIX machines saving to an NTFS qtree in Data ONTAP versions 6.0.3 through 6.3 may see permission errors when trying to save designs. The following setting may be used that works to silently ignore attempts to set UNIX permission on NTFS qtrees after the design file is saved:

  ```
  options cifs.ntfs_ignore_unix_security_ops on
  ```

About the Cluster-Aware Cache

The Oracle BI Server maintains a local, disk-based cache of query result sets called the query cache. The query cache allows a BI Server to potentially satisfy many query requests without accessing back-end databases. This reduction in communication costs can dramatically decrease query response time. Query cache entries become obsolete as updates occur on the back-end databases and must be purged periodically. For more information on the query cache, refer to the *Oracle Business Intelligence Server Administration Guide*.

In a clustered environment, Oracle BI Servers can be configured to access a shared cache that is referred to as the cluster-aware cache. This cluster-aware cache, residing on a shared file system storage device, stores seeding and purging events as well as the result sets associated with the seeding events. The seeding and purging events are sorted by time and stored on the shared storage as a logical event queue. Individual BI Server nodes push to and pull from the logical event queue.

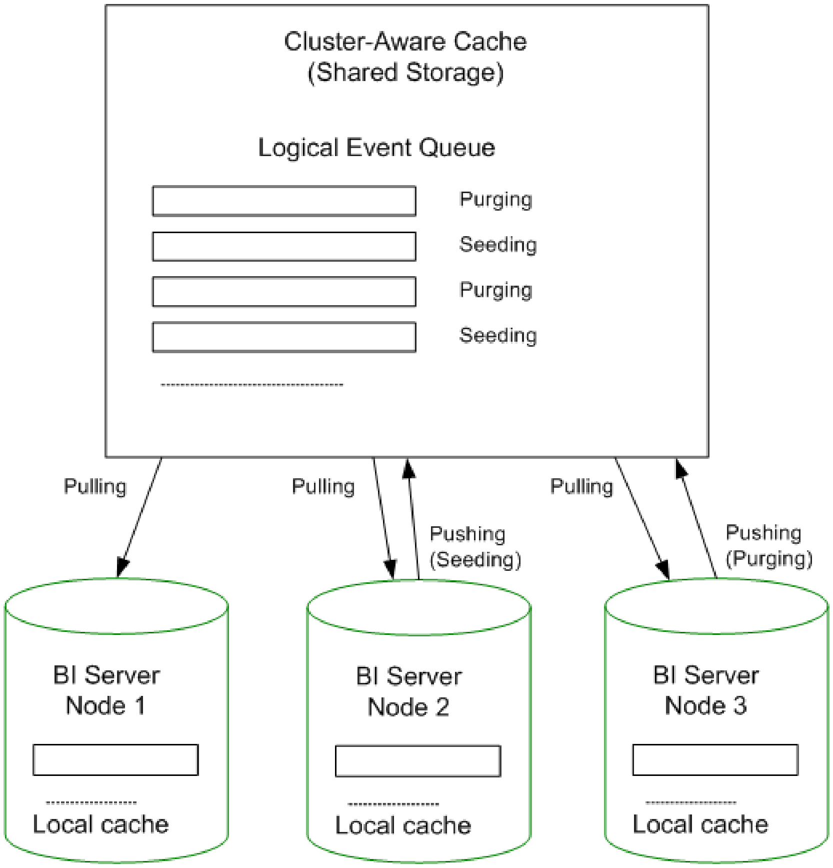

Figure 2. Cluster-Aware Caching

Figure 2 on page 23 shows three BI Server nodes sharing a global cache. The cluster-aware cache stores seeding or purging events held in a logical event queue. The arrows from Node 2 and Node 3 to the shared cache show BI Server Node 2 pushing a seeding event to the queue and BI Server Node 3 pushing a purging event to the queue. The arrows from the shared storage to each BI Server node show each node pulling from the common location. This occurs on a periodic basis and allows participating BI Server nodes to obtain updates to the logical event queue made by other BI Servers.

A BI Server node processes a seeding or purging event locally first in its caching system. It then pushes the event to the global cache on the shared storage. During the push event, the active BI Server node locks the logical event queue on the shared storage and then pushes in the seeding or purging event. In the case of conflict between seeding and purging, for example, one node wants to seed a query and another node wants to purge the same query, the event that comes in last will win.

The logical event queue in the global cache on the shared storage is composed of seeding and purging events from individual BI Server nodes. The queue is sorted according to the timestamp of the events. Hence, clocks on all BI Server nodes participating in cluster must be synchronized.

Each BI Server node polls the global cache on a periodic basis for new cache entries. This polling frequency is configurable. A snapshot of the queued logical events on the shared storage are pulled back to the node and a local logical event queue is constructed and then processed.

NOTE: The process of populating or purging seeded caches across all BI Server nodes that participate in the cluster does not occur in real time, and the elapse of the process is affected by multiple factors, such as the predefined polling interval, network bandwidth, and CPU loads.

As the query cache result set tends to get large, network bandwidth may pose a constraint. Therefore, the following must be chosen carefully:

- The set of cache that qualify for seeded cache
- The time interval for BI nodes to pick up seeded caches from shared storage (to avoid network congestion)

The cluster-aware cache parameters are configured in the NQSConfig.INI file for each BI Server node that participate in the cluster. For more information about configuring these parameters, see "Setting Parameters in the NQClusterConfig.INI File" on page 26.

A seeding or purging procedure is submitted to a specific BI Server node, as described in the chapter on query caching in the *Oracle Business Intelligence Server Administration Guide*. If that BI Server is a node in a BI cluster and the global cache parameters have been defined in BI Server configuration files, the seeding or purging events are propagated across all BI Server nodes that participate in the same clustered environment.

Configuration of BI Components for Clustering, Load Balancing, and Failover

This section describes the configuration of the NQSConfig.INI file to enable clustering, load balancing, and failover.

Setting Parameters in the NQSConfig.INI File

BI Server machines are configured for participation in a BI cluster by setting parameters in the NQSConfig.INI file. This file is located in the following directory on the BI Server machine.

To access the NQSConfig.INI file

- For Windows, access OracleBI_HOME\server\Config
- For Linux or UNIX, access OracleBI_HOME/server/Config

Setting the Cluster Participation Parameters

You must configure the CLUSTER_PARTICIPANT and RPC_SERVICE_OR_PORT parameters.

To set the Cluster Participation parameters

- Set the parameter CLUSTER_PARTICIPANT to YES for a BI Server to join the BI Cluster.
- Set the parameter RPC_SERVICE_OR_PORT to the desired port that the BI Server will listen on for client requests.

  The default port number is 9703. When the BI Server is a cluster participant, the following line must be commented out as shown:

  ```
  # SERVER_HOSTNAME_OR_IP_ADDRESSES = "ALLNICS";
  Multi-NIC is not supported for clustered deployments.
  ```

Setting the Repository Publishing Directory

To allow online modifications to be made to the repository, the following parameters must be set.

To set the repository publishing directory

- Set the parameter REPOSITORY_PUBLISHING_DIRECTORY to the path to the shared file system for the Repository Publishing Directory for all BI Servers participating in the cluster.
- Set the parameter REQUIRE_PUBLISHING_DIRECTORY to YES to make the repository publishing directory available, in order for the BI Server to start up and join the cluster.

Setting Cluster-Aware Caching Parameters

Caching is enabled by default for BI Servers. In addition to the caching-related parameters that are set for the BI Sever for the local query cache, you must set the cluster-aware cache parameters.

NOTE: A copy of the NQSConfig.INI file must reside on all BI Server machines that are part of the BI cluster.

To set cluster-aware caching parameters

- Set the parameter GLOBAL_CACHE_STORAGE_PATH to specify the following:
  - Location of the shared file system storage for the global cache that stores seeding and purging events.
  - Capacity of storage, depending on the maximum number of entries allowed for the global cache and the average size of each entry.
- Set the parameter MAX_GLOBAL_CACHE_ENTRIES to the maximum number of entries that are allowed in the global cache store.
- Set the parameter CACHE_POLL_SECONDS to specify the interval in seconds at which the BI Server will pull from the logical event queue in order to synchronize with other server nodes in the cluster.

- Set the parameter CLUSTER_AWARE_CACHE_LOGGING to turn on logging for the shared cache.
  - Set to YES (enable logging) only for debugging purposes.
  - Entries are made to the NQQuery.log file:
    - Under Windows, located in the directory OracleBI_HOME\server\Log
    - Under Linux and UNIX, located in the directory OracleBI_HOME/server/Log

Setting Parameters in the NQClusterConfig.INI File

This section describes the configuration of the NQClusterConfig.INI file to enable clustering, load balancing, and failover.

Accessing the NQClusterConfig.INI File

BI Cluster Controller, BI Server and BI Scheduler component instances use settings in the NQClusterConfig.INI file for operation in a BI cluster. This file is located in the following directory:

To access the NQClusterConfig.INI file

- Under Windows, access OracleBI_HOME\server\Config
- Under Linux or UNIX, access OracleBI_HOME/server/Config

Enabling Cluster Controller

Use the following procedure to enable the cluster controller.

To enable the cluster controller

- Set the parameter ENABLE_CONTROLLER to YES to allow the BI Server or BI Scheduler node to be controlled by the Cluster Controller for cluster operations.

Designating the Primary and Secondary Cluster Controllers

Use the following procedure to designate the primary and secondary cluster controllers.

To designate the cluster controllers

- Set the parameter PRIMARY_CONTROLLER to the machine hosting the Primary Cluster Controller.
- Set the parameter SECONDARY_CONTROLLER to the machine hosting the Secondary Cluster Controller.

Identifying the Servers Participating in the Cluster

Use the following procedure to identify the servers that are participating in the cluster

To identify the servers participating in the cluster

- Set the parameter SERVERS by entering, between double quotes, a comma-separated list of BI Server hostnames.

  NOTE: Do not use Fully Qualified Domain Names (FQDNs).

- Set the parameter SCHEDULERS to identify the BI Scheduler servers that will participate in the cluster by entering, between double quotes, a comma-separated list of Scheduler hostnames, RPC ports, and monitor ports. For example:

  ```
  SCHEDULERS = "scheduler1:<rpc port>:<monitor port>", "scheduler2:<rpc
  port>:<monitor port>";
  ```

  where:

  - <rpc port> is the port on which Scheduler listens for BI Presentation Services and Jpb Manager connections.

    This is the port number specified in the Server Port Number parameter that is set during initial configuration of the BI Scheduler. The default port is 9705.

    Refer to the chapter on configuring the BI Scheduler component in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

  - <monitor port> is the Scheduler port used by the Cluster Controller for life cycle monitoring.

    This must match the port number set for the Cluster Monitor Port parameter during configuration of BI Scheduler for participation in the BI cluster. The default cluster monitor port is 9708. Refer to the topic "Configuring BI Scheduler" on page 30.

Designating the Master BI Server

Use the following procedure to designate the master BI server

To designate the master BI server

- Set the parameter MASTER_SERVER by entering, between double quotes, the host name of the Master BI Server machine.

Enabling Cluster Communication and Operation

Use the following procedure to enable cluster communication and operation.

NOTE: A copy of the NQClusterConfig.INI file configured for clustering must reside on all machines that host either a Cluster Controller, BI Server or BI Scheduler component that participates in the cluster.

To enable cluster communication and operation

- Set the parameter SERVER_POLL_SECONDS to the frequency of heartbeat messages between the Cluster Controller and the server nodes in the cluster. The default is 5 seconds,

- Set the parameter CONTROLLER_POLL_SECONDS to the frequency of heartbeat messages between the Cluster Controllers. The default is 5 seconds.
- Set the parameter CLIENT_SERVER_PORT port number to be the same as that of RPC_SERVICE_OR_PORT in the NQSConfig.INI file. This is the port that is used by the BI Server for client requests. The default is 9703.
- Set the parameter CLIENT_CONTROLLER_PORT port number to be used by the BI ODBC DSN for communication with the Cluster Controller. The default is 9706.
- Set the parameter MONITOR_CONTROLLER_PORT to the port used by Cluster Controllers for Cluster Controller to Cluster Controller communication. The default is 9700.
- Set the parameter MONITOR_SERVER_PORT to the port that is used by Cluster Controller for life cycle monitoring of BI Servers.

Configuring BI Presentation Services

The BI Presentation Services component communicates with other BI components. To enable communication in a clustered deployment, BI Presentation Services must be configured to point to clustered instances of the other BI components.

BI Presentation Services must be configured to communicate with BI Scheduler instances via the Primary and Secondary Cluster Controllers. BI Presentation Services must also be configured to point to the Javahost cluster. In addition, BI Presentation Services must be configured to use the Presentation Catalog on the network share.

This configuration is done by setting parameters in the BI Presentation Services configuration file, instanceconfig.xml. This file is located in the following directory:

- Under Windows, OracleBIData_HOME\web\Config
- Under Linux or UNIX, OracleBIData_HOME/web/Config

The configuration must be done for all instances of BI Presentation Services in the BI deployment.

NOTE: BI Presentation Services communicates with BI Servers via the BI ODBC Client Data Source. The BI ODBC Data Source must be configured to communicate with the Primary and Secondary Cluster Controllers as described in "Modifying BI ODBC Data Sources For Communication With BI Cluster" on page 36.

Communication with BI Scheduler

Communication with the clustered Scheduler instances occurs through the Cluster Controllers. The configuration file must identify the Primary Cluster Controller and Secondary Cluster Controller in the Alerts section of the instanceconfig.xml file:

```
<Alerts>
   <ScheduleServer
   ccsPrimary="<Primary Cluster Controller>"

    ccsPrimaryPort="<CLIENT_CONTROLLER_PORT>" ccsSecondary="<Secondary Cluster
    Controller>" ccsSecondaryPort="<CLIENT_CONTROLLER_PORT>"/>
```

```
</Alerts>
```

- where:
- ccsPrimary is set to the Primary Cluster Controller machine identified by the PRIMARY_CONTROLLER parameter in the NQClusterConfig.INI file.
- ccsSecondary is set to the Secondary Cluster Controller machined identified by the SECONDARY_CONTROLLER parameter in the NQClusterConfig.INI file.
- ccsPrimaryPort and ccsSecondaryPort are set to the port identified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file. The default is 9706.

Communication with BI Javahost Cluster

The communication of BI Presentation Services with the Javahost cluster is enabled by identifying the Javahost instances and the listening ports in the instanceconfig.xml file. This is done by specifying the JavaHostProxy element and Hosts sub-elements. The Host element contains one or more Host sub-elements that identify specific instances of BI Javahost and port. If these elements are not set, BI Presentation Services will connect to a single Javahost on the default listening port on the local machine

```
<ServerInstance>

.
.
   <JavaHostProxy>

   <Hosts>
      <Host address="<Javahost Machine1>" port="9810" />
      <Host address="<Javahost Machine2>" port="9810" />
   </Hosts>

   </JavaHostProxy>
.
.
</ServerInstance>
```

The default Javahost port is 9810, and can be obtained from the Port element in the config.xml file on the machine where Javahost is installed. The config.xml file is located in the following directory:

- Under Windows, OracleBI_HOME\web\javahost\config
- Under Linux or UNIX, OracleBI_HOME/web/javahost/config

When two or more Host elements are uniquely identified, load balancing of requests to the Javahost cluster automatically takes effect.

The JavaHostProxy node has an optional element LoadBalancer that contains the sub-element Ping.

The following table shows the attributes of the LoadBalancer/Ping and Hosts/Host elements.

Element	Attribute	Attribute Description
LoadBalancer/Ping	keepAliveMaxFailures	Specifies the number of ping failures required before the host is declared dead. Default is 5.
	keepAliveFrequencySecs	Specifies the ping frequency in seconds. Default is 20.
Hosts/Host	Address	Identifies the Javahost instance.
	Post	Identifies the port number. Default: 9810.

Identifying the Shared Presentation Catalog

If you are using a shared Presentation Catalog on a storage device, you must point BI Presentation Services to the shared location for the Presentation Catalog.

To identify the shared Presentation catalog

- Modify the <CatalogPath> element to point to the shared Presentation Catalog.

  For example:

  ```
  <CatalogPath>\\FS-HOST\OracleBIData\web\catalog\customCatalog</CatalogPath>
  ```

  where customCatalog is the name of the shared Presentation Catalog.

  When multiple BI Presentation Services instances are deployed, the following elements and their values must be specified in the isntanceconfig.xml file:

  ```
  <Catalog>
     AccountIndexRefreshSecs>120</AccountIndexRefreshSecs>
     <AccountCacheTimeoutSecs>180</AccountCacheTimeoutSecs>
     <CacheTimeoutSecs>120</CacheTimeoutSecs>
     <CacheCleanupSecs>600</CacheCleanupSecs>
     <PrivilegeCacheTimeoutSecs>180</PrivilegeCacheTimeoutSecs>
  </Catalog>
  ```

  NOTE: The above settings manage when BI Presentation Services cache is updated from disk in environments with multiple BI Presentation Services instances.

Configuring BI Scheduler

BI Scheduler must first be configured following instructions in the chapter on configuring BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

- For steps that require configuration of BI Presentation Services, you must perform the configuration on all instances of BI Presentation Services in your deployment. For example, one of the steps requires you to add BI Scheduler Administrator credentials to the BI Presentation Services Credential Store. You must update the credential store for each instance of BI Presentation Services to store the BI Scheduler credentials, or copy the credential store with updated credentials to each BI Presentation Services machine. The instanceconfig.xml file for each BI Presentation Services must specify the location of the credential store.
- The BI Scheduler instances must be configured to participate in the cluster. The Scheduler instances must be configured to communicate with:
  - Multiple BI Presentation Services instances
  - Multiple BI Javahost instances
- The Scheduler instances must be configured to use the shared location for the Scheduler scripts.

Configuration of BI Scheduler is done either through Job Manager, installed with the Scheduler component on Windows, or by using the schconfig utility on Windows, Linux, or UNIX. The schconfig utility is located in the following directory:

- Under Windows, OracleBI_HOME\server\bin
- Under Linux or UNIX, OracleBI_HOME/setup on Linux or UNIX.

Configuration settings are saved to the Scheduler instanceconfig.xml file located in the following directory

- Under Window, OracleBIData_HOME\scheduler\config
- Under Linux or UNIX, OracleBIData_HOME/scheduler/config on Linux or UNIX.

To configure the BI Scheduler parameters for cluster participation

- In the Advanced tab, set the following parameters for Scheduler to participate in a BI cluster:
  - Check the box Participant in Cluster.
  - Set Cluster Monitor Port to 9708.

  NOTE: The Cluster Monitor Port defaults to 9708. If this port number is changed, you will need to change the <monitor port> port number specified in the SCHEDULER parameter in the NQClusterConfig.INI file.

Communication with Multiple BI Presentation Services in the Cluster

Since communications to BI Presentation Services instances in a cluster are not controlled via the Cluster Controllers, the list of BI Presentation Services instances must be specified. This is done either through Job Manager or the schconfig utility. When more than one BI Presentation Services instance is specified, load balancing of requests to the multiple BI Presentation Services instances automatically takes effect.

- In the iBots tab of the Scheduler Configuration window in Job Manager, provide the comma-separated list of the BI Presentation Services instances:

```
OBI Presentation Server = <BI Presentation Services Host1>:9710, <BI Presentation
Services Host2>:9710
```

- From the Delivers Configuration Menu when running the schconfig utility, selection 3 - Configure iBots, select the 1-Saw Machine name parameter and set it to the following:

  ```
  <BI Presentation Services Host1>:9710, <BI Presentation Services Host2>:9710
  ```

  The default port that BI Presentation Services uses to listen to RPC calls is 9710.

Communication with BI Javahost Cluster

Communication with the multiple BI Javahost instances in a cluster is enabled for the Scheduler instance by specifying the list of BI Javahost instances. This is done either through Job Manager or the schconfig utility. When more than one BI Javahost is specified, load balancing of requests automatically effective.

- In the Java Extension tab of the Scheduler Configuration window in Job Manager, provide the comma-separated list of BI Javahost instances:

  ```
  Java Host Servers = <BI Javahost Machine1>:<Port>, <BI Javahost Machine2>:<Port>
  ```

- From the Delivers Configuration Menu when running the schconfig utility, select 5 - Configure Java Extension, Select the 1 - Java Host Server parameter and set it to the following:

  ```
  <BI Javahost Machine1>:<Port>, <BI Javahost Machine2>:<Port>
  ```

The default port that Javahost listens on is 9810. This port is configured in the <Port> element in the config.xml configuration file for BI Javahost. For more information, see “Configuring Oracle BI Javahost for Communication Over SSL” on page 117.

Share for Scheduler Scripts

BI Scheduler must be directed to use the network share for the Scheduler Scripts. This is done either through Job Manager or the schconfig utility

- In the General tab of the Scheduler Configuration window in Job Manager, set the Scheduler Script Path and Default Script Path fields to network shares.
- From the Scheduler Configuration menu when running the schconfig utility, select 2 - General. Select the 1 - Scheduler Script Path to the following:

  ```
  <Shared file system location for Scheduler scripts>
  ```

- From the Scheduler Configuration menu when running the schconfig utility, select 2 - General. Select the 2 - Default Script Path to the following:

  ```
  <Shared file system location for default scripts>
  ```

Copy default and custom Scheduler scripts to the shared file system locations.

Configuring BI Presentation Services Plug-in

BI Presentation Services Plug-in must be configured to direct requests to the BI Presentation Services instances in the deployment.

BI Presentation Services consists of two types:

- For a Microsoft IIS web server, the BI Presentation Services is an ISAPI Plug-in.

  The configuration process for this type of BI Presentation Services Plug-in is described in the topic "Configuring BI Presentation Services Plug-in (ISAPI Plug-in)" on page 33.

- For J2EE based application servers, the BI Presentation Services Plug-in is a Java Servlet deployed in a web container.

  The configuration process for this type of BI Presentation Services Plug-in is described in the topic "Configuring BI Presentation Services Plug-in (Java Servlet)" on page 35.

Configuring BI Presentation Services Plug-in (ISAPI Plug-in)

The instances of BI Presentation Services that the BI Presentation Services Plug-in can direct requests to is specified in the ServerConnectInfo element the in the isapiconfig.xml. The isapiconfig.xml file is located in OracleBIDATA_HOME\web\config.

The top level element, ServerConnectInfo, contains the following elements:

- LoadBalancer

  Contains the Ping element.

- Hosts

  Contains one or more Host elements. Each Host element identifies a specific instance of BI Presentation Services and port.

The attributes for the LoadBalancer, Ping, and Host elements are shown in the following table.

Element	Attribute	Attribute Description
LoadBalancer	autoRoute	Specifies whether to automatically redirect requests to another instance of Oracle BI Presentation Services if the current server fails: ■ True. Automatically redirects requests. ■ False. Does not automatically redirect requests. (The default setting is false).
	encryptHostID	Specifies whether to encrypt the value of the cookie used for session binding. ■ True. Encrypts the cookie value. (The default setting is true.) ■ False. Does not encrypt the cookie value.
LoadBalancer/Ping	keepAliveMaxFailures	Specifies the number of ping failures required before the host is declared dead. The default is 5.
	keepAliveFrequencySecs	Specifies the ping frequency in seconds. The default is 20.
Hosts/Host	Address	Identifies the BI Presentation Services instance.
	Port	Identifies the port number that BI Presentation Services is listening on. The default is 9710

When more than one Host element is specified, load balancing of requests to the multiple BI Presentation Services instances is automatically enabled.

The following is an example of a ServerConnectInfo entry.

```
<?xml version="1.0" encoding="utf-8" ?>
<WebConfig>
   <ServerInstance>
      <ServerConnectInfo>
         <LoadBalancer autoRoute="true"/>
            <Hosts>
               <Host address="BI Presentation Services Machine1" port="9710"/>
               <Host address="BI Presentation Services Machine2" port="9710"/>
            </Hosts>
      </ServerConnectInfo>
   </ServerInstance>
</WebConfig>
```

NOTE: This configuring must be performed for all BI Presentation Services Plug-in instances in the deployment.

Configuring BI Presentation Services Plug-in (Java Servlet)

The instances of BI Presentation Services that the BI Presentation Services Plug-in can direct requests to is specified in the web.xml file for the Java Servlet. The default version of this file is located in OracleBI_HOME\web\app\WEB-INF on Windows and OracleBI_HOME/web/app/WEB-INF on Linux or UNIX.

The following table contains parameters for the Java Servlet.

Connection	Element Description
oracle.bi.presentation.Sawservers	Identifies the Oracle BI Presentation Services instances that requests can be directed to. The value of this element is a list of host:port pairs, with each pair identifying a BI Presentation Services instance.
oracle.bi.presentation.sawconnect. loadbalance.AlwaysKeepSessionAffiliation	Controls whether requests belonging to the same session can be redirected to another instance of Oracle BI Presentation Services if the current Oracle BI Presentation Services instance score is too low:· ■ Y. Allows redirection of requests. ■ N. Disallows redirection of requests.

The following entry is an example of a web.xml file.

```
<init-param>
   <param-name>oracle.bi.presentation.Sawservers</param-name>
   <param-value>server1:port;server2:port2;server3:port</param-value>
   <param-
name>oracle.bi.presentation.sawconnect.loadbalance.AlwaysKeepSessionAffliation
   </param-name>
   <param-value>Y</param-value>
</init-param>
```

Where server:port identifies the BI Presentation Services instance. The default port that BI Presentation Services listens on is 9710.

NOTE: This configuration must be performed on all machines where the BI Presentation Services Plug-in Java Servlet has been deployed.

Modifying BI ODBC Data Sources For Communication With BI Cluster

The BI ODBC Data Sources must be modified for communication to occur with the BI cluster, depending on your environment:

- In environments with a single Oracle BI Server, the BI ODBC Data Source points to the BI Server instance.
- In an environment where multiple BI Server instances participate in a BI cluster, the BI ODBC Data Source must point to the Primary and Secondary Controllers.

The following components use the BI ODBC Data Source:

- BI Presentation Services
- BI Administration Tool

Modify the BI ODBC DSN on the following machines:

- All machines that host BI Presentation Services.
- The machine that hosts the BI Administration Tool used to connect to the clustered environment.

After the BI ODBC client is configured to communicate with the BI cluster, the Administration Tool will connect to the repository on the Master BI Server.

Modifying the BI ODBC Data Source under Windows

By default, the BI ODBC Data Source Name (DSN) is AnalyticsWeb. The DSN is modified using the ODBC Data Source Administrator control panel, as shown in the following procedure.

To modify the ODBC data source under Windows

1. On the ODBC Data Source Administrator Systems DSN tab, select the AnalyticsWeb DSN.
2. Click the Configure button to open the Oracle BI Server DSN Configuration window.
3. Check the box "Is this a clustered DSN" and enter the names of the Primary Cluster Controller and Secondary Cluster Controller machines in the Primary Controller and Secondary Controller text boxes respectively.
4. Set the Controller Port field as appropriate.

   The default value is 9706. This port value must match the port number set for the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file.

Modifying the odbc.ini file under Linux or UNIX

On Linux and UNIX machines, the odbc.ini file is located in the OracleBI_HOME/setup directory.

- Modify odbc.ini as follows:

```
IsClusteredDSN=Yes
PrimaryCCS=BI-CCS-01
PrimaryCCSPort=9706
SecondaryCCS=BI-CCS-02
SecondaryCCSPort=9706
Regional=No
```

- Set the PrimaryCCS parameter to the Primary Cluster Controller host.
- Set the SecondaryCCS parameter to the Secondary Cluster Controller host.
- Set the ports to the port number specified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file. The default is 9706.

Best Practices for Setting Up an Oracle BI Clustered Environment

This section provides some general guidelines on setting up and configuring your Oracle BI clustered environment.

It is recommended that every machine in the BI deployment be set up with the same path structure. If the primary paths for OracleBI_HOME, OracleBIData_HOME and OracleBITemp_HOME are, for example, D:\OracleBI, D:\OracleBIData and D:\OracleBIData\tmp respectively, then all machines should

Configure the NQSConfig.INI and NQClusterConfig.INI files, Scheduler instanceconfig.xml file, and BI Presentation Services instanceconfig.xml file for one instance each of BI Server, BI Scheduler and BI Presentation Services. Ensure proper functioning of Oracle BI. Then copy the appropriate configuration files to the other machines.

Troubleshooting an Oracle BI Clustered Environment

Log files for Oracle BI components help you trouble shoot issues that may occur in your BI deployment after enabling the clustering, load balancing and failover capabilities of Oracle BI. To effectively diagnose issues, understand the lines of communication that occur as described in the topic Communication Between Oracle BI Components in a Clustered Environment in this chapter.

Review the log files for the BI components for every instance in the cluster. The table below shows the log file and its location for each BI component. Log files will record any client-side failures that may have occurred due to misconfiguration. While some Failover events are not logged, the Cluster Controller log file will record crashes of any BI Scheduler or BI Server instances. Review the Event Viewer log on Windows and syslog on Linux or UNIX systems.

Review the log files after initial start up. If an BI Server or BI Scheduler instance has not been configured correctly and as expected by the Cluster Controller, then the instance, though it may not shut down, will not be added to the cluster. The log files will record such failures.

The following table lists the log files for BI Components.

BI Components	Log File	Log File Location
BI Server	NQServer.log	■ Windows OracleBI_HOME\server\Log§ ■ Linux or UNIX OracleBI_HOME/server/Log
BI Cluster Controller	NQCluster.log	■ Windows OracleBI_HOME\server\Log§ ■ Linux or UNIX OracleBI_HOME/server/Log
BI Scheduler	NQScheduler.log	■ Windows OracleBI_HOME\server\Log§ ■ Linux or UNIX OracleBI_HOME/server/Log
BI Presentation Services	sawlog*.log (For example, sawlog0.log)	■ Windows OracleBIData_HOME\web\log§ ■ Linux or UNIX OracleBIData_HOME/web/log
BI JavaHost	host*.log.* (For example, jhost0.log.0)	■ Windows OracleBIData_HOME\web\log\javahost ■ Linux or UNIX OracleBIData_HOME/web/log/javahost

Monitoring and Managing BI Servers and BI Schedulers in a Cluster

The Cluster Manager utility in BI Administration Tool allows administrators to monitor and manage operations and activities of BI Scheduler and BI Server instances in the cluster. Note that BI Presentation Services and BI Javahost are not monitored by the Cluster Manager. For more information on using the Cluster Manager and its capabilities, see chapter on clustering BI Servers in the *Oracle Business Intelligence Server Administration Guide*.

Deploying Oracle Business Intelligence Publisher for High Availability

Figure 3 depicts the deployment of BI Publisher for high availability.

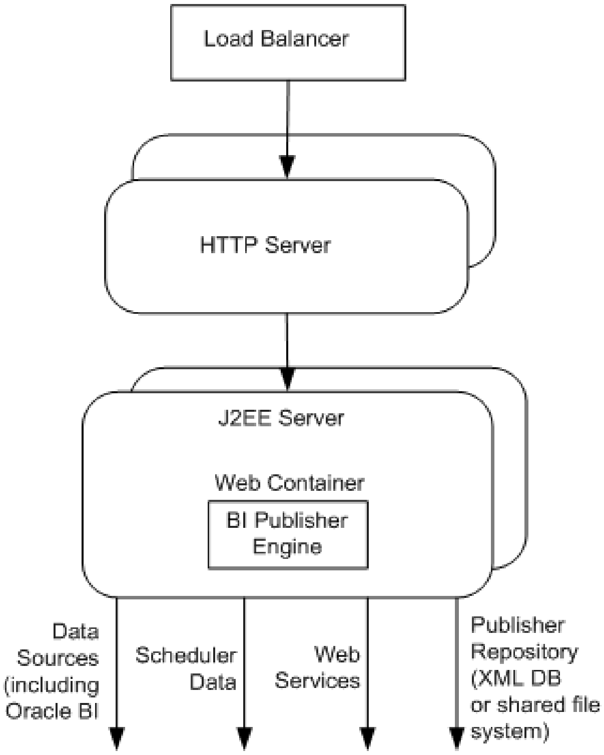

Figure 3. Deployment of BI Publisher for High Availability

BI Publisher J2EE Engine

The BI Publisher Engine is deployed in a web container in a J2EE server. Any number of BI Publisher Engines can be deployed simultaneously, each serving requests. For session state to be preserved in case of failure of any one particular engine, session state replication should be enabled.

Data Sources

Data sources are the raw sources used to compile the report data. This can be any JDBC data source. When integrated with Oracle BI, a JDBC data source named OracleBI EE is configured to point to the Oracle BI Server as a data source. When the multiple BI Servers participate in a BI clustered environment, the Oracle BI EE data source must point to the Primary and Secondary Cluster Controllers.

Repository Services and Scheduler Data

BI Publisher can store its Report metadata in either an XML DB or in a file system. All BI Publisher processes must have access to this Repository. In the case of a file system repository, a shared file system solution such as NAS must be used.

Integrating BI Publisher with Oracle BI Clustered Environment

The integration of BI Publisher with Oracle BI allows users to generate highly formatted reports based on Oracle BI data. The configuration of BI Publisher to enable this integration is described in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

To integrate BI Publisher with Oracle BI deployed in a clustered environment, you must specify some of the parameters differently from what is necessary in a single server environment.

Setting the Oracle BI EE Data Source

The Oracle BI EE Data Source must point to the clustered BI Servers via the Cluster Controllers. Perform this task in the BI Publisher application.

To set the Oracle BI EE data source in BI Publisher

1. In the BI Publisher application, in the Admin tab, click the link JDBC Connection under Data Sources.

2. Update the Oracle BI EE Data Source setting by changing the Connection String parameter to the following:

   ```
   jdbc:oraclebi://<Primary Cluster Controller Host>:9706/PrimaryCCS=<Primary
   Cluster Controller Host>;PrimaryCCSPort=9706;SecondaryCCS=<Secondary Cluster
   Controller Host>;SecondaryCCSPort=9706
   ```

   where:

   - PrimaryCCS parameter is set to the Primary Cluster Controller.
   - SecondaryCCS parameter is set to the Secondary Cluster Controller.
   - PrimaryCCSPort and SecondaryCCSPort parameters are set to the port specified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file. (The default is 9706.)
   - Username and Password fields are set to the Oracle BI Administrator credentials.
   - Database Driver Class field is set to oracle.bi.jdbc.AnaJdbcDriver.

Integrating with BI Presentation Services

The following procedure demonstrates how to specify the values for the BI Publisher URL to connect to Oracle BI. For example, http://bi.mycompany.com:80/analytics/saw.dll. This allows BI Answers requests to be visible in BI Publisher.

Perform this task in the BI Publisher application.

NOTE: The Oracle BI credentials specified in Administrator Username and Administrator Password fields are used to log in.

To specify the values for the BI Publisher URL to connect to Oracle BI

1. In the BI Publisher application, in the Admin tab, click the link Oracle BI Presentation Services under Integration.
2. From the Server Protocol dropdown, select http or https.
3. From the Server Version dropdown, select v4.
4. For the Server field, enter the server host name or Virtual IP for your Oracle BI environment. For example: bi.mycompany.com
5. Enter the port for the server in the Port field. For example, 80.
6. In the Administrator Username and Password fields, specify the Oracle BI Administrator credentials.
7. Set the URL Suffix field to the default value of analytics/saw.dll.

Integrating with BI Server Security

If you have defined BI Server Security as the security model in BI Publisher, you must modify the JDBC connection string to point to the clustered BI Servers via the Cluster Controllers.

To modify the JDBC connection string in BI Publisher

1. Log in to the BI Publisher Enterprise application as administrator.
2. In the Admin tab, go to the Security Configuration page.
3. Modify the Connection String as follows:

   ```
   jdbc:oraclebi://BI-CCS-01:9706/PrimaryCCS=BI-CCS-
   01;PrimaryCCSPort=9706;SecondaryCCS=BI-CCS-02;SecondaryCCSPort=9706
   ```

   where:

   - PrimaryCCS points to the Primary Cluster Controller.
   - SecondaryCCS points to the Secondary Cluster Controller.
   - PrimaryCCSPort and SecondaryCCSPort are set to the port specified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file.

For more information on setting BI Publisher to integrate with BI Server Security, refer to the *Oracle Business Intelligence Publisher User's Guide*.

Integrating BI Publisher in BI Presentation Services User Interface

The Oracle BI Reporting and Publishing feature allows the integration of BI Publisher in the BI Presentation Services user interface. Oracle BI users access the BI Publisher application from the link More Products > BI Publisher.

BI Presentation Services must point to the BI Publisher application URL. This is done by specifying the BI Publisher application URLs in the instanceconfig.xml file for BI Presentation Services under the <AdvancedReporting> tag.

For example:

```
<AdvancedReporting>
.
.
   <ServerURL>http://bi-publisher.mycompany.com/xmlpserver/services/XMLPService
   </ServerURL>
   <WebURL>http://bi-publisher.mycompany.com/xmlpserver</WebURL>
   <AdminURL>http://bi-publisher.mycompany.com/xmlpserver/servlet/admin</AdminURL>
.
.
</AdvancedReporting>
```

where:

- The BI Publisher application is deployed for high availability as described in "Deploying Oracle Business Intelligence Publisher for High Availability" on page 39.
- The Oracle BI Publisher Administrator credentials are stored in the BI Presentation Services Credential Store, as described in the chapter on configuring BI Publisher in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.
- When multiple BI Presentation Services instances participate in the Oracle BI deployment, the BI Publisher Administrator credentials must be stored in the Credential Store for every BI Presentation Services instance.

4 Deploying Oracle Business Intelligence for High Availability

This chapter describes the deployment of Oracle Business Intelligence in a high availability environment. It outlines the installation and configuration procedures for the BI components needed to achieve end-to-end availability. The Oracle BI components are configured to use the native clustering, load balancing and failover mechanisms described in Chapter 3, “Clustering, Load Balancing, and Failover in Oracle Business Intelligence.” BI components are deployed in either an active-active or active-passive mode to maximize availability.

Figure 4 on page 44 shows the deployment of Oracle Business Intelligence for high availability.

- A Load Balancer serves as the entry point and load balances Oracle BI web requests to multiple web servers. Two scenarios for the web tier are shown.
  - The first scenario shows IIS as the web server with BI Presentation Services Plug-in (ISAPI) deployed.
  - The second scenario shows a J2EE based application server with BI Presentation Services Plug-in (Java Servlet) deployed in a web container in the J2EE server. For information on deploying the BI web tier in a De-Militarized Zone (DMZ) and details on communication over firewalls, see Chapter 9, “Other Deployment-Related Topics.”
- Multiple instances of BI Presentation Services, BI Servers and BI Scheduler components are installed and the Oracle BI environment is configured for clustering, load balancing and failover of its components using native capability.
- Figure 4 does not show the lines of communication between the different BI components. See Figure 1 on page 16 and the topic “Communication Between Oracle BI Components in a Clustered Environment” on page 15 for details.

For details of the deployment of BI Publisher component for high availability, see the topic "Deploying Oracle Business Intelligence Publisher for High Availability" on page 63.

v

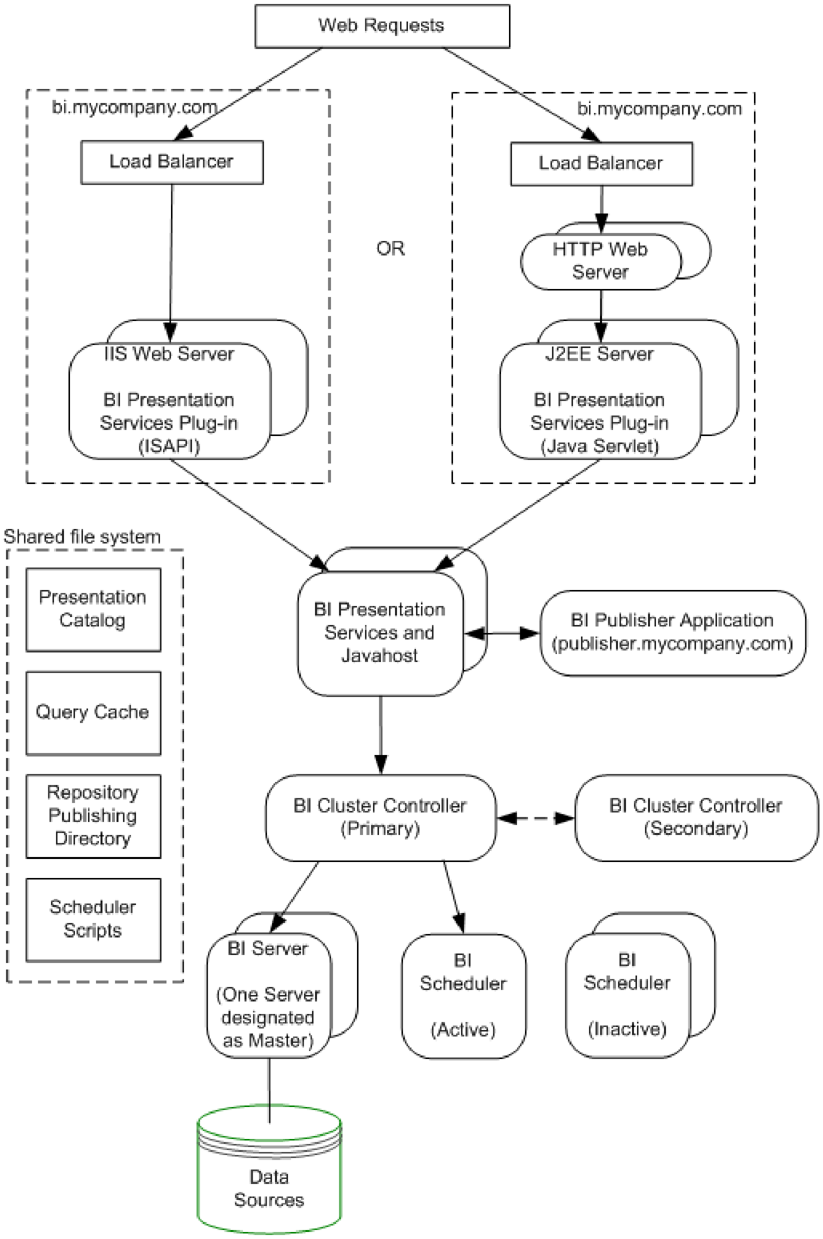
Web Requests
bi.mycompany.com
Load Balancer
IIS Web Server
BI Presentation Services Plug-in (ISAPI)
OR
bi.mycompany.com
Load Balancer
HTTP Web Server
J2EE Server
BI Presentation Services Plug-in (Java Servlet)
Shared file system
Presentation Catalog
Query Cache
Repository Publishing Directory
Scheduler Scripts
BI Presentation Services and Javahost
BI Publisher Application (publisher.mycompany.com)
BI Cluster Controller (Primary)
BI Cluster Controller (Secondary)
BI Server
(One Server designated as Master)
BI Scheduler
(Active)
BI Scheduler
(Inactive)
Data Sources

Figure 4. Deployment of Oracle BI for High Availability

Planning For the Installation

This topic provides general guidelines to help you plan the installation of Oracle BI for high availability.

Review the *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence Suite Enterprise Edition 10.1.3.2*. Also review the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for information on installation requirements.

Determine the number of instances of each Oracle BI component that will form part of the deployment based on your requirements. The maximum number of BI Servers that can participate in a cluster is 16. BI Scheduler instances participate in the cluster in an active-passive configuration. Only one BI Scheduler instance is active and processing requests at a given time; the other instances are passive.

NOTE: For purposes of illustrations, this document describes the installation of two instances of each Oracle BI component.

Determine which BI components will be co-located. For example, the Primary Cluster Controller, one BI Server node and BI Scheduler may be installed on one machine.

NOTE: For the purposes of illustration, this topic describes the installation of Oracle BI components on separate machines.

Identify the machines on which you will deploy the Oracle BI components. Refer to the *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence Suite Enterprise Edition 10.1.3.2* for more information on supported operating systems and hardware requirements.

Identify a shared network location for the Presentation Catalog, Repository Publishing Directory, Cluster-Aware Cache and Scheduler Scripts.

Refer to the *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence Suite Enterprise Edition 10.1.3.2* and to the topic "Shared Files and Directories" on page 51 for requirements and considerations for the shared file systems.

For deployments on Windows, identify a Domain account under which the BI services will run. This Domain account must also have the Log on as a service right. Refer to the appendix Granting the Oracle BI Log On as Service Right for procedures to perform the task of granting the Log on as a service right based on your specific Windows platform.

Note the following requirements:

- All BI Servers participating in the cluster need to be within the same domain and on the same LAN subnet. Geographically separated computers are not supported.
- The clock on each server participating in a cluster must be kept in synchronization. Out-of-sync clocks can skew reporting.

Installation of Oracle BI Components

Refer to the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for information on required third-party software, pre-requisites and installation requirements for Windows and Linux platforms.

NOTE: Determine what installation type best suits your needs by reading the topic on Basic and Advanced types of Oracle BI installation in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

Installing Oracle BI Client Tools (Windows)

BI Client Tools are used to administer and manage BI components. These tools must be installed on a Windows machine. Refer to the *System Requirements and Supported Platforms for Oracle Business Intelligence Suite Enterprise Edition* for information on the Windows operating systems that are supported for Client Tools.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. For Oracle BI Client Tools, select Basic installation type.
2. For Oracle BI Client Tools, select the setup type Oracle Business Intelligence Client Tools.

After you perform this procedure, the following components are installed:

- Oracle Business Intelligence ODBC Driver
- Oracle Business Intelligence JDBC Driver
- Oracle Business Intelligence Administration Tool
- Oracle Business Intelligence Client
- Oracle Business Intelligence Catalog Manager
- Oracle Business Intelligence Job Manager

Installing Oracle BI Cluster Controllers

Install Oracle BI Cluster Controller on the machines that you have identified to host this component by following the installation steps identified in this topic. For high availability, install two instances of Cluster Controller on different machines. One instance will serve as the Primary Cluster Controller; the other as the Secondary Cluster Controller.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. For Oracle BI Cluster controller, select either Basic or Advanced installation type, depending on your deployment
2. For Oracle BI Cluster controller, select the setup type Custom.
   - Select the feature Oracle Business Intelligence Cluster Controller for installation.

- If you are co-locating other BI components on this machine (for example, BI Server, BI Scheduler, or both) select the other desired components for installation.

3. On Windows machines, the Oracle BI Services screen is part of the installation.
   - Enter a domain account to run the BI services. Do not specify a LocalSystem account.
   - Select the start up type for the services—either manual or automatic.

NOTE: For reference purposes, this topic assumes that BI Cluster Controllers have been installed on machines BI-CCS-01 and BI-CCS-02.

Installing Oracle BI Server

Install BI Server instances on each of the machines that you have identified to host this component by following the installation steps identified in this topic. You can install a maximum of 16 BI Servers to participate in the BI Cluster.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. For Oracle BI Server, select either Basic or Advanced installation type, depending on your deployment

   NOTE: If the Advanced installation type has been chosen for installing the BI Server, then the Systems Management component for BI Server will be deployed in the Oracle Application Server installed on the machine. If the Basic installation type has been chosen for installing the BI Server, then the Systems Management component for BI Server will be deployed in OC4J.

2. For Oracle BI Server, select the setup type Custom.
   - Select the feature Oracle Business Intelligence Server for installation.
   - If you are co-locating other BI components on this machine (for example, BI Scheduler) select the other desired components for installation.

3. On Windows machines, the Oracle BI Services screen is part of the installation.
   - Enter a domain account to run the BI services. Do not specify a LocalSystem account.
   - Select the start up type for the services—either manual or automatic.

The installed BI Servers will be configured to participate in the BI Cluster by setting parameters in the NQSConfig.INI and NQClusterConfig.INI files on each machine that hosts the BI Server.

NOTE: For reference purposes, this topic assumes that two BI Servers have been installed on machines BI-SERVER-01 and BI-SERVER-02.

Installing Oracle BI Scheduler

Install BI Scheduler instances on each of the machines that you have identified to host this component by following the installation steps identified in this topic. The BI Scheduler component participates in the BI Cluster in active-passive mode. Install BI Scheduler on two machines, one will be identified as the active node and the other as the passive node.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. For Oracle BI Scheduler, select either Basic or Advanced installation type, depending on your deployment

   NOTE: If the Advanced installation type has been chosen for installing the BI Scheduler, then the Systems Management component for BI Scheduler will be deployed in the Oracle Application Server installed on the machine. If the Basic installation type has been chosen for installing the BI Scheduler, then the Systems Management component for BI Scheduler will be deployed in OC4J.

2. For Oracle BI Scheduler, select the setup type Custom.

   - Select the feature Oracle Business Intelligence Scheduler for installation.
   - If you are co-locating other BI components on this machine (for example, BI Server) select the other desired components for installation.

3. On Windows machines, the Oracle BI Services screen is part of the installation.

   - Enter a domain account to run the BI services. Do not specify a LocalSystem account.
   - Select the start up type for the services—either manual or automatic.

4. Complete the additional configuration steps. Refer to chapter on configuring Oracle Business Intelligence Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for details.

The installed BI Scheduler instances will be configured to participate in the BI Cluster. This is described in the topic "Configuring BI Scheduler" on page 57.

NOTE: For reference purposes, this topic assumes that two BI Schedulers have been installed on machines BI-SCHEDULER-01 and BI-SCHEDULER-02.

Installing Oracle BI Presentation Services and Oracle BI Javahost

Install BI Presentation Services and BI Javahost on each of the machines that you have identified to host these components following their installation.

NOTE: BI Javahost is installed along with BI Presentation Services by the BI Installer. There is no separate component option for BI Javahost.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1. On Windows 2003, ensure that you have Data Execution Prevention (DEP) disabled as described on the first screen of the installer. If DEP is not disabled prior to running the installer, the charting server installation fails.

2 For Oracle BI Presentation Services, select either Basic or Advanced installation type, depending on your deployment.

NOTE: If the Advanced installation type has been chosen for installing the BI Presentation Services, then the Systems Management component for BI Presentation Services will be deployed in the Oracle Application Server installed on the machine. If the Basic installation type has been chosen, then the Systems Management component for BI Presentation Services will be deployed OC4J.

3 For Oracle BI Presentation Services, select the setup type Custom.

- Select the feature Oracle Business Intelligence Presentation Services for installation.
- If you are co-locating other BI components on this machine (for example, BI Presentation Services Plug-in) select the other desired components for installation.

4 On Windows machines, the Oracle BI Services screen is part of the installation.

- Enter a domain account to run the BI services. Do not specify a LocalSystem account.
- Select the start up type for the services—either manual or automatic.

The installed BI Presentation Services instances are configured to participate in the BI Cluster by setting parameters in the instanceconfig.xml file on each machine that hosts BI Presentation Services. This is described in the topic "Configuring BI Presentation Services" on page 55.

NOTE: For reference purposes, this topic assumes that two BI Presentation Services instances have been installed on machines BI-PS-01 and BI-PS-02.

Installing Oracle BI Presentation Services Plug-in

Use one of the following procedures to install the Oracle BI Presentation Services Plug-in depending on your web server:

- "Installing BI Presentation Services Plug-in (ISAPI)" on page 49
- "Installing BI Presentation Services Plug-in for J2EE Application Servers" on page 50

Installing BI Presentation Services Plug-in (ISAPI)

Use this procedure to install BI Presentation Services Plug-in (ISAPI) on all machines hosting IIS instances that will service Oracle Business Intelligence.

Oracle BI components are installed using the Oracle Business Intelligence Installer.

1 For Oracle BI Presentation Services, select either Basic or Advanced installation type, depending on your deployment.

NOTE: If the Advanced installation type has been chosen for installing the BI Presentation Services, then the Systems Management component for BI Presentation Services will be deployed in the Oracle Application Server installed on the machine. If the Basic installation type has been chosen, then the Systems Management component for BI Presentation Services will be deployed OC4J.

2 For Oracle BI Presentation Services Plug-in, select the setup type Custom.

- Select the feature Oracle Business Intelligence Presentation Services Plug-in for installation.
- If you are co-locating other BI components on this machine (for example, BI Presentation Services) select the other desired components for installation.

3. Select Microsoft IIS on the Application Server Selection screen.
4. On the BI Presentation Services Connection Details screen, enter the host name and port for a BI Presentation Services instance. The BI Presentation Services Plug-in will be later configured to communicate with all instances of BI Presentation Services in the deployment.

The installed ISAPI Plug-in instances will be configured to communicate with the BI Presentation Services instances by setting parameters in the isapiconfig.xml file on each machine that hosts the ISAPI Plug-in. This is described in the topic "Configuring BI Presentation Services Plug-in" on page 59.

NOTE: This topic refers to the two machines on which ISAPI Plug-ins have been installed as BI-ISAPI-01 and BI-ISAPI-02.

Installing BI Presentation Services Plug-in for J2EE Application Servers

When the BI Presentation Services component is installed, analytics.ear and analytics.war files are installed in OracleBI\web on Windows and OracleBI/web on Linux or UNIX. Deploy the appropriate file to the J2EE container on the Application Server of your choice. See Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence Enterprise Edition 10.1.3.2 for a list of supported Application Servers. Perform this deployment on all instances of the J2EE container in your web cluster.

To install the BI Presentation Services Plug-in for J2EE Application Servers

- BI Presentation Services Plug-in (Java Servlet) is deployed by the installer into either an OC4J standalone instance or in Oracle Application Server OC4J instance.
  - BI Presentation Services Plug-in (Java Servlet) deployed in a standalone OC4J:
    - Select the Basic installation type and choose OC4J when prompted for the Application Server.
    - Select Custom setup type.
    - Choose the Oracle BI Presentation Services Plug-in feature.
  - BI Presentation Services Plug-in (Java Servlet) deployed into an OC4J instance in the Oracle Application Server:
    - Select the Advanced installation type.
    - Select Custom setup type.
    - Choose the Oracle BI Presentation Services Plug-in feature.

Shared Files and Directories

The BI components deployed in a clustered environment must share certain files and directories as described below. A shared storage device such as NAS or SAN may be used. Refer to the topic "Shared Files and Directories" on page 21 for important considerations for the shared disks.

Presentation Catalog

- Create a network share for the Presentation Catalog. All instances of BI Presentation Services in the cluster must have read and write access to this share.
- Place the Presentation Catalog on the network share.
- Reference the shared Presentation Catalog as \\FS-HOST\OracleBIData\web\catalog\customCatalog, where *customCatalog* an example of the catalog name.

Repository Publishing Directory

- Create a shared directory for the Repository Publishing Directory. The Master BI Server must have read and write access to this directory. All other BI Servers must have read access.
- Reference the shared network share for the Repository Publishing Directory as \\FS-HOST\OracleBIData\ClusterRpd.

Cluster-Aware Cache

- Create a shared directory for the global cache. All BI Servers must have read and write access to this directory.
- Reference the shared global cache as \\FS-HOST\OracleBIData\ClusterCache.

Scheduler Scripts

- Create network shares for the Scheduler scripts. The Scheduler servers must have read and write access to this share.
- Reference the shared Scheduler scripts as \\FS_HOST\OracleBI\Server\Scripts\Common and \\FS_HOST\OracleBI\Server\Scripts\Scheduler.
- Copy default and custom Scheduler scripts from to the corresponding network shares created for the Scheduler scripts.

Configuration of Oracle BI Components for Clustering, Load Balancing, and Failover

Use the procedures in this topic to configure the Oracle BI components for clustering, load balancing, and failover.

Pre-Configuration Tasks

Before configuring the Oracle BI components for clustering and load balancing, perform the following tasks.

To perform pre-configuration tasks

1. Identify the Cluster Controller to serve as the Primary Cluster Controller.

   For example, BI-CCS-01.

2. Identify the Cluster Controller instance to serve as the Secondary Cluster Controller. For example, BI-CCS-02.
3. Identify the BI Server instance to serve as the Master BI Server. For example, BI-SERVER-01.
4. Copy the repository file (RPD) to the machines hosting the BI Servers.
   - The repository file must be copied to OracleBI_HOME\server\Repository.
   - On Linux or UNIX, the file must be copied to OracleBI_HOME/server/Repository.
5. Before performing the configurations, shut down all BI services or processes.

   Restart the BI services or processes after configuration is complete.

Setting Parameters in the NQSConfig.INI File

This topic shows how to use the NQSConfig.INI file to configure your deployment. The NQSConfig.INI file is located as follows:

- For Windows: OracleBI\server\Config
- For Linux or UNIX: OracleBI/server/Config

To set parameters in the NQSConfig.INI file

1. Open the NQSConfig.INI file for editing.
2. In the Repository section of the NQSConfig.INI, define your repository by setting the logical repository name and file name pair. For example:

   ```
   Star = < Custom rpd filename>, DEFAULT;
   ```

3 Set Cache parameters. Caching is enabled by default.

To use the *cluster-aware caching* capability, set the cluster-aware cache parameters for the BI Sever. In the Query Result Cache section of the NQSConfig.INI file, uncomment and set the following parameters:

- GLOBAL_CACHE_STORAGE_PATH. Set "<*path to shared storage for cache*>" and <*Size*>.
- MAX_GLOBAL_CACHE_ENTRIES. Set <*Max number of entries*>.
- CACHE_POLL_SECONDS. Set <*Polling interval in seconds*>.

4 In the Server Section, uncomment the parameter CLUSTER_PARTICIPANT and set it to YES.

5 When the BI Server is a cluster participant, comment out the parameter SERVER_HOSTNAME_OR_IP_ADDRESSES = "ALLNICS";

6 Set the parameter RPC_SERVICE_OR_PORT to the desired port that the BI Server will listen on.

NOTE: The default port is 9703.

7 In order for online modifications to be made to the repository, uncomment and set the following parameters:

- REPOSITORY_PUBLISHING_DIRECTORY. Set "<path to shared network location>".
- REQUIRE_PUBLISHING_DIRECTORY. Set to YES.

  For example:

```
[ REPOSITORY ]
Star = custom.rpd, DEFAULT;

[ CACHE ]
ENABLE = YES:

// Cluster-aware cache
GLOBAL_CACHE_STORAGE_PATH = "\\FS-HOST\OracleBI\ClusterCache" 700 MB;
MAX_GLOBAL_CACHE_ENTRIES = 1000;
CACHE_POLL_SECONDS = 300;
CLUSTER_AWARE_CACHE_LOGGING = NO;

[ SERVER ]

# SERVER_HOSTNAME_OR_IP_ADDRESSES = "ALLNICS"
CLUSTER_PARTICIPANT = YES;
REPOSITORY_PUBLISHING_DIRECTORY = "\\FS-HOST\OracleBIData\ClusterRpd";
REQUIRE_PUBLISHING_DIRECTORY = YES;
```

Setting Parameters in the NQClusterConfig.INI File

This topic shows how to use the NQSClusterConfig.INI file to configure your deployment. The NQSClusterConfig.INI file is located as follows:

- For Windows: OracleBI\server\Config
- For Linux or UNIX: OracleBI/server/Config

To set parameters in the NQClusterConfig.INI file

1. Open the NQClusterConfig.INI file for editing.
2. Set the parameter ENABLE_CONTROLLER to YES to enable clustering.
3. Identify the Primary and Secondary Cluster Controllers:
   - Set the parameter PRIMARY_CONTROLLER to the machine hosting the Primary Cluster Controller.
   - Set the parameter SECONDARY_CONTROLLER to the machine hosting the Secondary Cluster Controller.
4. Set the parameter SERVERS by entering a comma-separated list of the BI Server hostnames.

   NOTE: Do not specify Fully Qualified Domain Names for the BI Servers.
5. Set the parameter MASTER_SERVER by entering the hostname of the Master BI Server machine.
6. Set the parameter SCHEDULERS for the Scheduler servers participating in the cluster.

   ```
   SCHEDULERS = "scheduler1:<rpc port>:<monitor port>", "scheduler2:<rpc
   port>:<monitor port>";
   ```

   where:

   - <rpc port> is the Server Port Number that is set during initial configuration of the BI Scheduler. The default port is 9705. Refer to the chapter on configuring the BI Scheduler component in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.
   - <monitor port> is the Cluster Monitor Port that is set during configuration of the BI Scheduler for participation in the BI cluster. The default is 9708. Refer to the topic "Configuring BI Scheduler" on page 57.

7 The following parameters are set to the default values shown. Modify the parameter values as required for your deployment.

Parameter	For...	Default
SERVER_POLL_SECONDS	BI Server to Cluster Controller polling frequency	5
CONTROLLER_POLL_SECONDS	Cluster Controller to Cluster Controller polling frequency	5
CLIENT_SERVER_PORT	Set to same as RPC_SERVICE_OR_PORT in NQSConfig.INI	9703
CLIENT_CONTROLLER_PORT	Port used by clustered ODBC for communication with Cluster Controller	9706
MONITOR_CONTROLLER_PORT	Port used by Cluster Controllers for Cluster Controller to Cluster Controller communication	9700
MONITOR_SERVER_PORT	Port used by Cluster Controller for life cycle monitoring of BI Servers	9701

For example:

```
#  NQClusterConfig.INI

[Cluster]

ENABLE_CONTROLLER = YES;
PRIMARY_CONTROLLER   = "BI-CCS-01";
SECONDARY_CONTROLLER = "BI-CCS-02";
SERVERS = "BI-SERVER-01","BI-SERVER-02";
MASTER_SERVER = "BI-SERVER-01";
SERVER_POLL_SECONDS = 5;
CONTROLLER_POLL_SECONDS = 5;
CLIENT_SERVER_PORT = 9703;
CLIENT_CONTROLLER_PORT = 9706;
MONITOR_CONTROLLER_PORT = 9700;
MONITOR_SERVER_PORT = 9701;
SCHEDULERS = "BI-SCHEDULER-01:9705:9708","BI-SCHEDULER-02:9705:9708";
```

Configuring BI Presentation Services

BI Presentation Services must be configured to communicate with BI Scheduler instances via the Primary and Secondary Cluster Controllers. BI Presentation Services must also be configured to point to the Javahost cluster. In addition, BI Presentation Services must be configured to use the Presentation Catalog on the network share.

The BI Presentation Services is configured by setting parameters in the configuration file instanceconfig.xml. The instanceconfig.XML file is located in the following directories:

- Under Windows: OracleBIData_HOME\web\config
- Under Linux or UNIX: OracleBIData_HOME/web/config

Use the following procedure to configure BI Presentation Services on each machine that hosts BI Presentation Services.

To configure BI Presentation Services

1. Open the configuration file instanceconfig.xml for editing.
2. Locate the <Alerts> element.

   Configure for communication with the clustered Scheduler instances as follows:

   ```
   <Alerts>
      <ScheduleServer
      ccsPrimary="BI-CCS-01" ccsPrimaryPort="9706" ccsSecondary="BI-CCS-02"
   ccsSecondaryPort="9706"/>
   </Alerts>
   ```

   where:

   - ccsPrimary is set to value of the PRIMARY_CONTROLLER parameter in the NQClusterConfig.INI file.
   - ccsPrimaryPort is set value of the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file.
   - ccsSecondary is set to value of the SECONDARY_CONTROLLER parameter in the NQClusterConfig.INI file.
   - ccsSecondaryPort is set value of the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file.
3. Under the ServerInstance tag, create the JavaHostProxy element.
4. Set the JavaHostProxy element attributes and values to point to the Javahost cluster:

   ```
   <ServerInstance>
   .
   .
   <JavaHostProxy>
      <Hosts>
         <Host address="BI-PS-01" port="9810" />
         <Host address="BI-PS-02" port="9810" />
      </Hosts>

   </JavaHostProxy>
   .
   .
   </ServerInstance>
   ```

   where:

   - BI-PS-01 and BI-PS-02 are the machines that host the BI Javahost component. BI Javahost was installed along with the installation of BI Presentation Services.

- The Hosts element contains Host sub-elements that identify the Javahost and port pairs. (The default Javahost port is 9810.)
  - Use the values in the Port element in the config.xml file on the machine where Javahost is installed.
  - The config.xml file is located in OracleBI_HOME\web\javahost\config (Windows) and OracleBI_HOME/web/javahost/config (Linux).

5 Modify the <CatalogPath> element to point to the shared Presentation Catalog:

```
<CatalogPath>\\FS-HOST\OracleBIData\web\catalog\customCatalog</CatalogPath>
```

6 Under the ServerInstance element, create the Catalog sub-element with the following attributes and values:

```
<Catalog>
   <AccountIndexRefreshSecs>120</AccountIndexRefreshSecs>
   <AccountCacheTimeoutSecs>180</AccountCacheTimeoutSecs>
   <CacheTimeoutSecs>120</CacheTimeoutSecs>

   <CacheCleanupSecs>600</CacheCleanupSecs>
   <PrivilegeCacheTimeoutSecs>180</PrivilegeCacheTimeoutSecs>
</Catalog>
```

These settings manage when BI Presentation Services cache is updated from disk in environments with multiple BI Presentation Services instances.

7 Save changes to the file.

Configuring BI Scheduler

Before configuring the BI Scheduler for participation in the BI Cluster, you must configure BI Scheduler following the steps described in the chapter on configuring BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

The integration of BI Scheduler with BI Presentation Services requires the BI Scheduler Administrator credentials be added to the BI Presentation Services Credential Store.

- You must update the credential store for every instance of BI Presentation Services in your deployment to store the BI Scheduler credentials, or copy the credential store with updated credentials to each BI Presentation Services machine.
- The instanceconfig.xml file for each BI Presentation Services must specify the location of the credential store.

The Scheduler instances participate in the BI cluster in active-passive mode. That is, only one Scheduler instance processes and executes Scheduler jobs at a given time.

Use the following procedure to configure BI Scheduler on each machine that hosts this component.

Configuring BI Scheduler Installed on Windows

Use the BI Scheduler Job Manager to configure Scheduler for participation in a BI cluster.

To configure BI Scheduler installed on Windows

1. From the Windows Start menu select Programs > Oracle Business Intelligence > Job Manager.
2. In Job Manager, select File > Configuration Options
3. In the Scheduler > Advanced tab of the Scheduler Configuration window, check the "Participant in Cluster" check box. The Cluster Monitor Port defaults to 9708. Change this port number as needed.
4. In the Scheduler > General tab, set the Scheduler Script Path and Default Script Path to network shares. For example:

   ```
   Scheduler Script Path = \\FS-HOST\OracleBI\server\Scripts\Scheduler
   Default Script Path = \\FS-HOST\OracleBI\server\Scripts\Common
   ```

5. In the iBots tab of the Scheduler Configuration window, provide a comma-separated list of the BI Presentation Services instances. For example:

   ```
   OBI Presentation Server = BI-PS-01:9710, BI-PS-02:9710
   ```

   where 9710 is the default port on which BI Presentation Services listens to RPC calls.
6. In the Java Extension tab of the Scheduler Configuration window, provide the comma-separated list of BI Javahost instances. For example:

   ```
   Java Host Servers = BI-PS-01:9810, BI-PS-02:9810
   ```

   where the default Javahost port is 9810.

Configuring BI Scheduler Installed on Linux or UNIX

The Scheduler configuration options are set using schconfig, a console-based application. On the machines where Scheduler instances are installed, in the directory OracleBI_HOME/setup, run the command schconfig:

```
. sa-init.sh
schconfig
```

To configure BI Scheduler installed on Linux or UNIX

1. From the Delivers Configuration choices that appear, select 1 - Configure Scheduler.

2 For each Scheduler Configuration Menu choice shown in the following table, select the listed parameter and configure as shown.

Configuration Menu	Choice	Parameter	Setting
Scheduler	3 - Advanced	3 - Participant in Cluster	True
		4 - Cluster Monitor Port	9708 note The Cluster Monitor Port must match the <monitor port> port number specified in the SCHEDULER parameter in the NQClusterConfig.INI file.
	2 - General	1 - Scheduler Script Path	Shared network location for Scheduler scripts
		2 - Default Script Path	Shared network location for default scripts
Delivers	Configure iBots	1 - Saw Machine Name	BI-PS-01:9710, BI-PS-02:9710. Use a comma-separated list of BI Presentation Services hosts and ports. 9710 is the default port on which BI Presentation Services listens to RPC calls.
Java Extension		1 - Java Host Server	BI-PS-01:9810; BI-PS-02:9810. Use a comma-separated list of BI Javahost instances. 9810 is the default Javahost port.

3 Select 0 to quit and save changes when prompted.

4 Select 0 to quit the utility

Configuring BI Presentation Services Plug-in

BI Presentation Services must be configured to communicate with multiple BI Presentation Services instances.

BI Presentation Services consists of two types:

- For J2EE based application servers, the BI Presentation Services Plug-in is a Java Servlet deployed in a web container.

  The configuration process for this type of BI Presentation Services Plug-in is described in the topic "Configuring BI Presentation Services Plug-in (Java Servlet) for Deployments Using J2EE Based Application Servers" on page 60.

- For a Microsoft IIS web server, the BI Presentation Services is an ISAPI Plug-in.

  The configuration process for this type of BI Presentation Services Plug-in is described in the topic "Configuring BI Presentation Services Plug-in (ISAPI Plug-in) For Deployments Using IIS" on page 61.

Configuring BI Presentation Services Plug-in (Java Servlet) for Deployments Using J2EE Based Application Servers

Follow the steps in this procedure to configure BI Presentation Services Plug-in to communicate with the multiple BI Presentation Services instances.

For the Java servlet, configure the Plug-in in the web.xml file, located as follows:

- Under Windows: ORACLE_HOME\j2ee\bianalytics\applications\analytics\analytics\WEB-INF
- Under Linux or UNIX: ORACLE_HOME/j2ee/bianalytics/applications/analytics/analytics/WEB-INF

To configure the BI Presentation Services Plug-in Java Servlet

1. Open the web.xml file for editing.
2. Locate the following param-name and param value pairs:

   ```
   <init-param>
      <param-name>oracle.bi.presentation.sawserver.Host</param-name>
      <param-value><BI Presentation Services></param-value>
   </init-param>

   <init-param>
      <param-name>oracle.bi.presentation.sawserver.Port</param-name>
      <param-value>9710</param-value>
   </init-param>
   ```

3. Replace the current values with the following values:

   ```
   <init-param>
      <param-name>oracle.bi.presentation.Sawservers</param-name>
      <param-value>BI-PS-01:9710;BI-PS-02:9710</param-value>
   </init-param>
   ```

   where:

   - BI-PS-01 and BI-PS-02 are machines that host the BI Presentation Services instances.
   - BI Presentation Services listens on port 9710 to RPC calls from the BI Presentation Services plug-in.

   NOTE: The parameter name oracle.bi.presentation.Sawservers is case-sensitive on Linux and UNIX systems.

4. Save changes to the file.
5. Copy the web.xml file to OracleBI_HOME\web\app\WEB-INF on Windows and to OracleBI_HOME/web/app/WEB-INF on Linux.
6. Restart your Java Servlet container.

Configuring BI Presentation Services Plug-in (ISAPI Plug-in) For Deployments Using IIS

Follow the steps in this procedure to configure BI Presentation Services Plug-in to communicate with the multiple BI Presentation Services instances.

For Microsoft IIS, configure the Plug-in in the isapiconfig.xml file, located in the directory OracleBIData_HOME\web\config.

To configure the BI Presentation Services Plug-in for IIS

1. Open the isapiconfig.xml file for editing.
2. Locate the entry similar to the following:

   ```
   <ServerConnectInfo address="localhost" port="9710"/>
   ```

3. Replace this entry with the following lines:

   ```
   <ServerConnectInfo>
      <LoadBalancer autoRoute="true"/>
         <Hosts>
            <Host address="BI-PS-01" port="9710"/>
            <Host address="BI-PS-02" port="9710"/>|
         </Hosts>
   </ServerConnectInfo>
   ```

4. Save changes to the file.
5. Restart IIS.

Modifying BI ODBC Data Sources For Communication With BI Cluster

The BI ODBC Data Source Names (DSNs) must be modified for communication to occur with the BI cluster.

The following components use the BI ODBC to connect to the BI environment:

- BI Presentation Services
- BI Administration Tool

Modify the BI ODBC DSN on the following machines:

- All machines that host BI Presentation Services.
- The machine that hosts the BI Administration Tool used to connect to the clustered environment.

After the BI ODBC client is configured to communicate with the BI cluster, the Administration Tool will connect to the repository on the Master BI Server.

Modifying the BI ODBC Data Source under Windows

By default, the BI ODBC Data Source Name (DSN) is AnalyticsWeb. The DSN is modified using the ODBC Data Source Administrator control panel, as shown in the following procedure.

To modify the ODBC data source under Windows

1. On the ODBC Data Source Administrator Systems DSN tab, select the AnalyticsWeb DSN.
2. Click the Configure button to open the Oracle BI Server DSN Configuration window.
3. Check the box "Is this a clustered DSN."
4. In the Primary Controller text box, enter the name of the Primary Cluster Controller:

   ```
   BI-CCS-01
   ```

5. In the Secondary Controller text box, enter the name of the Secondary Cluster Controller:

   ```
   BI-CCS-02
   ```

6. Set the Controller Port field as appropriate.

   The default value is 9706. This port value must match the port number set for the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file.

Modifying the odbc.ini file under Linux or UNIX

On Linux and UNIX machines, the odbc.ini file is located in the OracleBI_HOME/setup directory.

To modify the ODBC data source under Linux

1. Open the odbc.ini file for editing.
2. Make the following modifications:

   ```
   IsClusteredDSN=Yes
   PrimaryCCS=BI-CCS-01
   PrimaryCCSPort=9706
   SecondaryCCS=BI-CCS-02
   SecondaryCCSPort=9706
   Regional=No
   ```

   where:

   - Set the PrimaryCCS parameter to the Primary Cluster Controller host.
   - Set the SecondaryCCS parameter to the Secondary Cluster Controller host.
   - Set the ports to the port number specified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file. The default is 9706.

3. Save changes to the file.

Deploying Oracle Business Intelligence Publisher for High Availability

Figure 5 on page 63 shows the deployment of BI Publisher for High Availability. Multiple BI Publisher Engines deployed in J2EE based application servers may be installed. In a clustered environment, the BI Publisher instances share a report repository, which may be either an XML database or a shared file system.

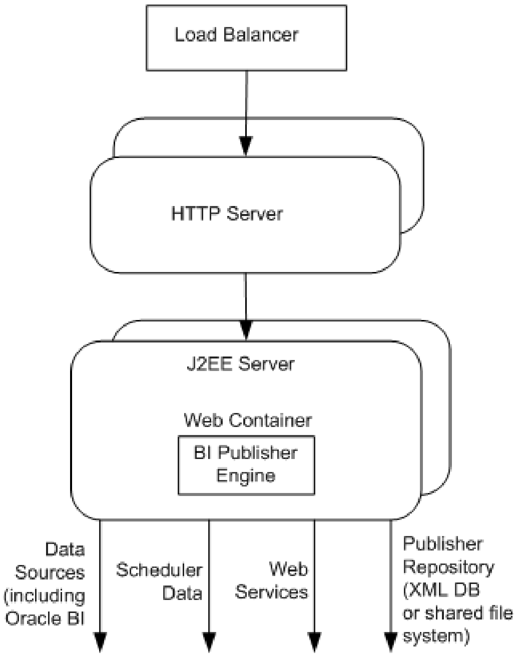

Figure 5. Deployment of BI Publisher for High Availability

Installing BI Publisher

Install multiple instances of BI Publisher Engine as required for your deployment. If you are using Oracle Application Server or standalone OC4J as your application server, use the BI installer to deploy the BI Publisher engine. If you are using any other J2EE based application server, you must deploy the BI Publisher Engine manually using the xmlpserver.ear or xmlpserver.war files provided in the Oracle Business Intelligence Enterprise Edition media pack.

To install BI Publisher on Standalone OC4J

1. Run the BI Installer.
2. Select the Basic Installation Type.

3. Select Oracle Business Intelligence Publisher option on the set up type screen.
4. Enter values for prompts on all other installer screens.
5. Complete the installation.

   NOTE: The BI Installer installs a standalone OC4J and deploys the BI Publisher engine in this container. Repeat the installation on all machines identified to host the BI Publisher engine.

To install BI Publisher on Oracle Application Server

1. Run the BI Installer.
2. Select the Advanced Installation Type.

   NOTE: Oracle Application Server must be installed on the machine.

3. Select Oracle Business Intelligence Publisher option on the set up type screen.
4. Enter values for prompts on all other installer screens.
5. Complete the installation.

   NOTE: The BI Installer deploys the BI Publisher engine into the Oracle Application Server. Repeat the installation on all machines identified to host the BI Publisher engine.

Deploying BI Publisher Engine on J2EE Based Application Servers

The Oracle Business Intelligence Enterprise Edition media pack provides the .ear and .war files needed to deploy the BI Publisher engine. These files are located on the installation CD in the following folders:

- xmlpserver.war file: \Server_Ancillary\Oracle_Business_Intelligence_Publisher\generic
- xmlpserver.ear file: \Server_Ancillary\Oracle_Business_Intelligence_Publisher\oc4j

For instructions on how to deploy these files in your J2EE based application server, refer to the file install.pdf (located in the folder \Server_Ancillary\Oracle_Business_Intelligence_Publisher).

Configuring BI Publisher to Use a Shared File System Repository

The BI Publisher instances in a cluster must share a report repository. A shared file system may be used.

- Create a directory on the shared file system that will serve as the root directory for the Publisher repository.

  For example, \Publisher\repository

- Copy the repository files and folders to this shared directory.

If you have installed BI Publisher using the Oracle Business Intelligence installer, the BI Publisher repository files and folder are located in the following directories:

- Windows: OracleBI_HOME\xmlp\XMLP
- Linux or UNIX: OracleBI_HOME/xmpl/XMLP

Use the following procedure to configure BI Publisher to access the repository on the shared file system. Perform this configuration on all instances of BI Publisher. Once you have completed this task, you can perform all additional configuration tasks to the clustered BI Publisher application.

To configure BI Publisher to access the repository on the shared file system

1. Log in to the BI Publisher Enterprise application using the URL http://<server>/xmlpserver.

   The default username and password is Administrator/Administrator.

2. Navigate to the Admin tab. Under System Maintenance, click on the Server Configuration link.
3. In the Report Repository section on the Server Configuration tab of the System Maintenance page, select File System as the Repository Type.
4. In the Path field, enter the path to the shared file system.

   Click Apply.

5. Restart the BI Publisher application for the change to take effect.

Integrating BI Publisher in BI Presentation Services User Interface

In order for users to access a clustered BI Publisher application from the More Products > BI Publisher link in the BI Presentation Services user interface, BI Presentation Services must be aware of the BI Publisher application URLs for the clustered environment.

The BI Publisher URLs are specified in the BI Presentation Services configuration file instanceconfig.xml, under the <AdvancedReporting> tag.

Open the instanceconfig.xml file for editing. This file is located in OracleBI_HOME\web\config on Windows and in OracleBI_HOME/web/config on Linux or UNIX.

To integrate BI Publisher in BI Presentation Services User interface

1. Modify the AdvancedReporting element as shown below:

```
<AdvancedReporting>
.
.
<ServerURL>http://bi-publisher.mycompany.com/xmlpserver/services/XMLPService</
ServerURL>
<WebURL>http://bi-publisher.mycompany.com/xmlpserver</WebURL>
<AdminURL>http://bi-publisher.mycompany.com/xmlpserver/servlet/admin</AdminURL>
```

```
.
.
</AdvancedReporting>
```

2. Replace http://bi-publisher.mycompany.com with the protocol and Virtual IP for your BI Publisher application.

3. Add the BI Publisher Administrator credentials to the BI Presentation Services Credential Store.

4. Specify the Credential Store in the instanceconfig.xml file for BI Presentation Services.

   Refer to the chapter on configuring BI Publisher in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for more information on these steps.

   NOTE: You must update the credential store for every instance of BI Presentation Services in your deployment to store the BI Publisher credentials, or copy the credential store with updated credentials to each BI Presentation Services machine. The instanceconfig.xml file for each BI Presentation Services must specify the location of the credential store.

Integrating BI Publisher with Oracle BI Clustered Environment

This topic describes the settings for the integration parameters when BI Publisher integrates with an Oracle BI environment deployed in a cluster.

Perform this task in the BI Publisher application.

To set the Oracle BI EE data source in BI Publisher

1. In the BI Publisher application, in the Admin tab, click the link JDBC Connection under Data Sources.

2. On the JDBC tab of the data sources page, select the Oracle BI EE Data Source.

3. Update the Oracle BI EE Data Source setting by changing the Connection String parameter to the following:

   ```
   jdbc:oraclebi://BI-CCS-01:9706/PrimaryCCS=BI-CCS-
   01;PrimaryCCSPort=9706;SecondaryCCS=BI-CCS-02;SecondaryCCSPort=9706
   ```

   where:

   - PrimaryCCS parameter is set to the Primary Cluster Controller.
   - SecondaryCCS parameter is set to the Secondary Cluster Controller.
   - PrimaryCCSPort and SecondaryCCSPort parameters are set to the port specified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file. (The default is 9706.)

4. Set the Username and Password fields to the Oracle BI Administrator credentials.

5. Verify that the Database Driver Class is set to the following:

```
oracle.bi.jdbc.AnaJdbcDriver
```

Click Apply.

Integrating with BI Presentation Services

Perform this task in the BI Publisher application.

NOTE: The Oracle BI credentials specified in Administrator Username and Administrator Password fields are used to log in.

To specify the values for the BI Publisher URL to connect to Oracle BI

1. In the BI Publisher application, in the Admin tab, click the link Oracle BI Presentation Services under Integration.
2. From the Server Protocol dropdown, select http or https.
3. From the Server Version dropdown, select v4.
4. For the Server field, enter the server host name or Virtual IP for your Oracle BI environment. For example: bi.mycompany.com
5. Enter the port for the server in the Port field. For example, 80.
6. In the Administrator Username and Password fields, specify the Oracle BI Administrator credentials.
7. Set the URL Suffix field to the default value of analytics/saw.dll.

NOTE: This allows BI Publisher to access BI Presentation Services by building the URL using <Server Protocol>://<Server>:<Port>/<URL Suffix. Login occurs using the Oracle BI credentials specified in the Administrator Username and Password fields.

Integrating with BI Server Security

If you have defined BI Server Security as the security model in BI Publisher, you must modify the JDBC connection string in the BI Publisher Enterprise application to point to the clustered BI Servers using the Cluster Controllers.

NOTE: This procedure assumes that you have BI Server Security already set up.

For more information on setting BI Publisher to integrate with BI Server Security, refer to the *Oracle Business Intelligence Publisher User's Guide*.

To modify the JDBC connection string in BI Publisher

1. Log in to the BI Publisher Enterprise application as administrator.
2. In the Admin tab, go to the Security Configuration page.
3. Modify the Connection String as follows:

```
jdbc:oraclebi://BI-CCS-01:9706/PrimaryCCS=BI-CCS-
01;PrimaryCCSPort=9706;SecondaryCCS=BI-CCS-02;SecondaryCCSPort=9706
```

where:

- PrimaryCCS points to the Primary Cluster Controller.
- SecondaryCCS points to the Secondary Cluster Controller.
- PrimaryCCSPort and SecondaryCCSPort are set to the port specified in the CLIENT_CONTROLLER_PORT parameter in the NQClusterConfig.INI file.

5 Oracle BI Presentation Services Credential Store

This chapter describes the Oracle Business Intelligence BI Presentation Services Credential Store. This Credential Store is a mechanism by which BI Presentation Services stores and accesses credentials and secrets. The BI Presentation Services Credential Store supports two types of credentials:

- Credentials based on a username and password
- x.509 credentials that use a private key and a public certificate

Certain tasks may require BI Presentation Services to communicate with other BI components (for example, with BI Publisher or BI Scheduler). BI Presentation Services needs to be aware of the user names and passwords that it should use to successfully authenticate against and establish communication with the other components. The credentials that are needed by BI Presentation Services are stored in the BI Presentation Services Credential Store and BI Presentation Services accesses and issues the credentials as appropriate.

The x.509 credentials that consist of private key and public certificate pairs are used by BI Presentation Services for communication when the communication occurs over the Secure Socket Layer (SSL).

Credentials and Aliases

Credentials in the credential store are stored along with unique alias values. BI Presentation Services accesses a credential via the alias that is mapped to the credential. Either through configuration or through hard-coded values, BI Presentation Services is aware of the alias it needs to retrieve for the task it must perform. At runtime, BI Presentation Services queries the credential store for the credential that is mapped to the desired alias. For example, when establishing a connection with the BI Scheduler component, BI Presentation Services must provide the BI Scheduler administrator credentials in order to authenticate successfully with BI Scheduler. The credential that is mapped to the alias "admin" is retrieved from the Credential Store and issued by BI Presentation Services for authentication against BI Scheduler.

Supported Storage Types

The following types of storage are supported for the BI Presentation Services Credential Store:

- **XML File System Store**

  Credentials and secrets may be stored in a file system store that is an XML file. The syntax of this file is defined by BI Presentation Services. The XML file contains nodes that point to files on disk for certificates and private keys. The file may also contain username and password based credentials, with optional encryption support for passwords. A default XML file store called credentialstore.xml is provided with BI Presentation Services.

- **Java Keystore**

  A Java keystore may be used for storage of keys and certificates. BI Presentation Services provides a Java component that can extract the certificates and keys stored within the Java keystore file.

- **Custom Store**

  The BI Presentation Services Credential Store supports storage of credentials and secrets in a custom store. BI Presentation Services accesses the credential data through a text based interchange format.

The XML file credential store, the Java Keystore and the custom store are described in further detail in section Credential Store Storage Types later in the chapter.

Credentials may be stored in multiple stores. At runtime, the BI Presentation Services Credential Store service will extract credentials from all configured locations and build a superset of credentials in an internal credential store for access by the system. For example, some credentials may be loaded from a Java keystore, some from an LDAP-based custom store and some from the BI Presentation Services proprietary XML file format store. In such a situation, the aliases across the distinct storage systems must be unique.

Configuring BI Presentation Services to Identify Credential Stores

BI Presentation Services must be able to identify the credential stores that store the credentials and secrets. The credential store details are specified in the BI Presentation Services configuration file instanceconfig.xml. This file is located in one of the following locations:

- Windows: OracleBIData_HOME\web\config
- Linux or UNIX: OracleBData_HOME/web/config

Credential stores are identified by the CredentialStore node in the instanceconfig.xml file. This top-level node contains one or more CredentialStorage sub-elements, each of which describes the type and location of a credential store. In addition, optional attributes for passphrase, as shown in Table 3 on page 71, may be specified for the CredentialStore node.

When BI Presentation Services encounters an encrypted credential, it needs to know the passphrase to use to decrypt the credential. A passphrase-related attribute may be specified either at the CredentialStore node level, as an attribute of the CredentialStorage sub-node, or may be stored along with the credential in the credential store. When a passphrase attribute is specified at the CredentialStore level, then BI Presentation Services will use it as a default to decrypt any encrypted credential that it encounters if no other passphrase attributes have been specified at lower levels.

For security reasons, it is recommended that the passphrase not be stored within the credential store, since unauthorized access to the credential store will reveal passwords and secrets. Providing the passphrase needed to decrypt a credential in the instanceconfig.xml file allows for enhanced security, because neither the instanceconfig.xml file nor the credential store on its own has enough information to expose the password.

NOTE: You should secure any file where a passphrase has been provided using OS or file system capabilities.

Table 3. CredentialStorage Node Elements and Sub-elements

Attribute Name	Required?	Description
passphrase	No	For encrypted files, this determines the passphrase used to decrypt the file.
passphraseFile	No	Path to a plain text file that contains the passphrase. This file should be suitably protected using OS and file system facilities.
passphraseEnvVar	No	Name of an environment variable that contains the passphrase.
passphraseLoader	No	This value specifies a command line that should be executed to extract the passphrase for the key. The command must result in the passphrase being written out, in plain text format, to the standard output of the executable.

The CredentialStorage Element

The CredentialStorage sub-element describes the type and location of the credential store and various options for the credential store. The attributes of the CredentialStorage node are shown in Table 4 on page 71.

In addition, all the passphrase-related attributes that are shown in Table 3 on page 71 also supported for the CredentialStorage node.

Table 4. CredentialStorage Node Passphrase-Related Attributes

Attribute Name	Required?	Description
type	Yes	This describes the type of credential store. Possible values are JKS, file and custom. JKS refers to a Java Keystore, file to a credential store in proprietary XML file format, and custom to a custom store.
propertyFile	Yes, if type=JKS	This value points to a standard Java property file which contains all additional options necessary for loading the Java credential store. This attribute must be specified if type=JKS.

Table 4. CredentialStorage Node Passphrase-Related Attributes

Attribute Name	Required?	Description
path	Yes, if type=file	This value points to an XML file in proprietary format that describes the credential store. This attribute must be specified if type=file.
commandLine	Yes, if type=custom	This value specifies the command line that should be executed to run the custom credential store loader. This attribute must be specified if type=custom.

The following example of the instanceconfig.xml file identifies two credential stores via two CredentialStorage elements. The first store identified is a Java Keystore of type JKS. The second is the XML file store, credentialstore.xml. For the XML file store, the passphrase "secret" has been provided as an attribute of the CredentialStorage element.

```
<WebConfig>
   <ServerInstance>
<!-- other settings -->

   <CredentialStore>
         <CredentialStorage type="JKS"
      propertyFile="D:\OracleBIData\web\config\jks_props.txt"/>
         <CredentialStorage type="file"
      path="D:\OracleBIData\web\config\credentialstore.xml"
      passphrase="secret"/>
   </CredentialStore>

<!-- other settings -->
   </ServerInstance>
</WebConfig>
```

Credential Store Storage Types

This section contains information about the different storage types that can be used with Oracle Business Intelligence. The following storage types are supported:

- “File System Store (XML File Store)” on page 72
- “Java Keystore” on page 78
- “Custom Store” on page 80

File System Store (XML File Store)

BI Presentation Services supports an XML file store that may contain the following items:

- References to certificates and private keys on disk
- Username and password based credentials embedded inline

The XML file is in proprietary format.

- The namespace for all elements in the XML file is com.siebel.analytics.web.credentialStore/v1.
- The prefix for the namespace is sawcs.
- Different types of XML nodes are used to specify the locations of the various files, and to associate each credential with an alias. The file format supports encrypted key files, as well as encrypted passwords.
- The root node for the XML file is credentialStore. The possible sub-nodes are credential, trustedCertificate and trustedCertificateDir.

A default credential store in XML file format, credentialstore.xml, is provided. credentialstore.xml is located in one of the following directories:

- Windows: OracleBIData_HOME\web\config
- Linux or UNIX: OracleBIData_HOME/web/config

A utility called CryptoTools is provided for the XML file manipulation. CryptoTools is located in one of the following directories:

- Windows: OracleBI_HOME\web\bin
- Linux or UNIX: OracleBI_HOME/web/bin

For more information about the usage and syntax of the CryptoTools utility, see Appendix B, "Using the CryptoTools Utility".

The XML file structure is described in the following topics.

The Credential Element

The credential element defines one of several credentials stored within the credential store file. Two types of credentials are supported:

- X.509 credentials, which are made up of a public certificate and a private key
- Username/password based credentials

Table 5 on page 73 lists the attributes of the credential element.

Table 5. credential Element Attributes

Attribute Name	Required?	Description
type	Yes	Credential type. Possible values: ■ x509 ■ usernamePassword
alias	Yes	This is the alias associated with this credential

Username Password Credentials

A username password credential of type usernamePassword contains two sub-elements: username and password.

- username: This element contains text which identifies a username known to the system. No attributes are specified for this element. The username should be specified as a text node(s) within the element.
- password: The password element supports storage of plain text and encrypted passwords. When storing a password in plain text, the password should be specified as text node(s) within the element. When storing an encrypted password, the schema specified by the W3C XML Encryption Syntax and Processing standard is used.

In addition, the attributes used to specify a decryption passphrase are also permitted. If no passphrase is supplied here and one is required (that is, if the password is found to be encrypted), then any passphrase supplied in general credential store configuration will be used. For example, if a passphrase is supplied on the CredentialStorage node of the instanceconfig.xml file then this passphrase will be used if an encrypted password is encountered. If no passphrase is supplied in the CredentialStorage node of instanceconfig.xml, then BI Presentation Services will look for the passphrase to be supplied in the CredentialStore element of instanceconfig.xml

The following are examples of the usernamePassword credential type in the XML file credential store.

NOTE: In the following example of the usernamePassword credential type in the instanceconfig.xml, the alias for the credential is "impersonation" and the username is "Impersonator". The password is encrypted using a passphrase "password" and the passphrase is stored inline. BI Presentation Services will use the passphrase to decrypt the password.

```
<?xml version="1.0" encoding="utf-8" ?>
<sawcs:credentialStore xmlns:sawcs="com.siebel.analytics.web.credentialStore/v1"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
xmlns:pkcs-5="http://www.rsasecurity.com/rsalabs/pkcs/schemas/pkcs-5#">

<!--
This is a username password credential with an encrypted password that is required.
In this example, the passphrase is shown inline.
-->

   <sawcs:credential type="usernamePassword" alias="impersonation">
      <sawcs:username>Impersonator</sawcs:username>
      <sawcs:password passphrase="password">
         <xenc:EncryptedData>
            <xenc:EncryptionMethod Algorithm="http://www.rsasecurity.com/rsalabs/
pkcs/schemas/pkcs-5#pbes2">
                  <pkcs-5:PBES2-params Algorithm="http://www.rsasecurity.com/
rsalabs/pkcs/schemas/pkcs-5#pbkdf2">
                     <pkcs-5:KeyDerivationFunc>
                        <pkcs-5:Parameters>
                           <pkcs-5:IterationCount>1024</pkcs-5:IterationCount>
                        </pkcs-5:Parameters>
                     </pkcs-5:KeyDerivationFunc>
                     <pkcs-5:EncryptionScheme Algorithm="http://www.w3.org/2001/
04/
xmlenc#tripledes-cbc"/>
```

```
                  </pkcs-5:PBES2-params>
               </xenc:EncryptionMethod>
               <xenc:CipherData>
                  <xenc:CipherValue>Ab76239KdhJiklj8967</xenc:CipherValue>
               </xenc:CipherData>
            </xenc:EncryptedData>
         </sawcs:password>
      </sawcs:credential>

<!--
This is a username password credential with an encrypted password. No passphrase is 
supplied, so any passphrase specified earlier would need to be used.
-->
   <sawcs:credential type="usernamePassword" alias="testuser">
      <sawcs:username>testuser</sawcs:username>
      <sawcs:password>
         <xenc:EncryptedData>
            <xenc:EncryptionMethod Algorithm="http://www.rsasecurity.com/rsalabs/
pkcs/schemas/pkcs-5#pbes2">
                  <pkcs-5:PBES2-params Algorithm="http://www.rsasecurity.com/
rsalabs/pkcs/schemas/pkcs-5#pbkdf2">
                     <pkcs-5:KeyDerivationFunc>
                        <pkcs-5:Parameters>
                           <pkcs-5:IterationCount>1024</pkcs-5:IterationCount>
                        </pkcs-5:Parameters>
                     </pkcs-5:KeyDerivationFunc>
                     <pkcs-5:EncryptionScheme Algorithm="http://www.w3.org/2001/
04/xmlenc#tripledes-cbc"/>
                  </pkcs-5:PBES2-params>
               </xenc:EncryptionMethod>
               <xenc:CipherData>

<xenc:CipherValue>VwEp5qS69bwC8tGl+RmE+l0/1TZc4q0+</xenc:CipherValue>
               </xenc:CipherData>
            </xenc:EncryptedData>
         </sawcs:password>
      </sawcs:credential>

</sawcs:credentialStore>
```

X.509 Credentials

An X.509 credential element contains two sub-elements: key and certificate.

The key element describes a private key. See Table 6 on page 76. All passphrase-related attributes specified in Table 4 on page 71 are also supported for the key element.

Table 6. X.509 Credential Key Attributes

Attribute Name	Required	Default	Description
path	Yes		The file on disk where the private key is stored.
encoding	Yes	Based on filename.	Encoding for the file. May be one of PEM or ASN1.

The **certificate** element describes a public certificate. See Table 7 on page 76.

Table 7. certificate Element Attributes

Attribute Name	Required	Default	Description
path	Yes		The file on disk where the certificate is stored.
encoding	No	Based on filename.	Encoding for the file. May be one of PEM or ASN1.

The **trustedCertificateDir** element describes a directory which contains CA certificates. All valid certificate files within this directory will be read and assumed to be CAs. No alias can be assigned to these certificates, and encoding is inferred from file extensions.

The trustedCertificateDir has one required attribute called path. This attribute specifies the path to the directory containing the CAs. See Table 8 on page 76.

Table 8. trustedCertificate Element Attributes

Attribute Name	Required	Default	Description
alias	No		The (optional) alias for this certificate.
path	Yes		The file on disk where the certificate is stored.
encoding	No	Based on filename.	Encoding for the file. May be one of PEM of ASN1.

The following is an example of a credential store XML file.

NOTE: The credential has an alias of "testuser" and a username of "testuser". The password is encrypted using a passphrase. The passphrase itself has not stored been stored in the file. Therefore, BI Presentation Services will use the passphrase supplied in the instanceconfig.xml file to decrypt the password. If a passphrase has been specified as an attribute of the credentialStorage element that describes this credential store, then that passphrase will be used. Otherwise, the passphrase specified as an attribute of the credential node of the instanceconfig.xml file will be used.

```
<?xml version="1.0" encoding="utf-8" ?>

<sawcs:credentialStore xmlns:sawcs="com.siebel.analytics.web.credentialStore/v1"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
xmlns:pkcs-5="http://www.rsasecurity.com/rsalabs/pkcs/schemas/pkcs-5#">

<!--
For this key, the passphrase is provided inline. Care should be taken to protect
this XML file suitably.
-->
   <sawcs:credential type="x509" alias="obips">
      <sawcs:key
         encoding="pem"
         passphrase="password"
         path="d:/temp/certificates/obips.pem"/>
      <sawcs:certificate encoding="pem" path="d:/temp/certificates/obips.crt"/>
   </sawcs:credential>

<!--
For this key, the passphrase is provided in a file. Care should be taken to protect
the passphrase file suitably.
-->
   <sawcs:credential type="x509" alias="obi_isapi">
      <sawcs:key
         encoding="pem"
         passphraseFile="d:/temp/certificates/obi_isapi_pwd.txt"
         path="d:/temp/certificates/obi_isapi.pem"/>
      <sawcs:certificate encoding="pem" path="d:/temp/certificates/obi_isapi.crt"/
>
   </sawcs:credential>

<!--
For this credential, execute the command line specified by passphraseLoader and
assume that the program writes out the entire passphrase to standard output. If the
program execution results in a non-zero return code, then this is considered an
error, and the output will be ignored. Any leading/trailing whitespace in the
passphrase will be trimmed out. The passphrase may only be composed of printable
ASCII characters.
-->
   <sawcs:credential type="x509" alias="obijavahost">
      <sawcs:key
         encoding="pem"
         passphraseLoader="d:/temp/certificates/getpassphrase.exe"
         path="d:/temp/certificates/ obijavahost.pem"/>
      <sawcs:certificate encoding="pem" path="d:/temp/certificates/
obijavahost.crt"/>
```

```
<!-- Individual CA certificates. -->

   <sawcs:trustedCertificate alias="obica" encoding="pem" path="d:/temp/
certificates/obica.crt"/>
   <sawcs:trustedCertificate alias="verisign" encoding="pem" path="d:/temp/
certificates/verisign.crt"/>
   <sawcs:trustedCertificate alias="thawte" encoding="pem" path="d:/temp/
certificates/thawte.crt"/>

<!--
Directory with CA certificate files. Use file extension to guess encoding and no
alias.-->
   <sawcs:trustedCertificateDir path="d:/temp/certificates/cacerts"/>

</sawcs:credentialStore>
```

Java Keystore

The BI Presentation Services Credential Store may be a standard Java keystore that allows the storage and management of keys and certificates. The default Keystore implementation provided by Java 2 SDK is a flat file called Java Keystore (JKS). The JKS keystore can be managed by the command line keytool utility that ships with the JDK. Refer to the JDK documentation for more information.

In order for BI Presentation Services to load this type of credential store, specify a Java property file with the options shown in Table 9 on page 78.

Table 9. Java Keystore Property File Options

Property	Required	Default	Description
KeyStore	No	Value of system property javax.net.ssl.keyStore.	The file where the private keys and their associated certificates are maintained.
KeyStorePwd	No	Value of system property javax.net.ssl. keyStorePassword	Password to access the credential store.
KeyStoreType	No	JKS	The type of Java based credential store (your Java installation must have support for it). JKS is the default implementation of KeyStore.
KeyAlias			Alias of key and certificate pair that will be used to retrieve the key-certificate pair from the store. If not specified, all keys and certificates present will be extracted.

Table 9. Java Keystore Property File Options

Property	Required	Default	Description
KeyPwd	No	Value of KeyStorePwd	The password for the specific key and certificate to be retrieved from the store.
TrustStore		Value of system property javax.net.ssl.trustStore.	The file where the trusted CA certificates are maintained.
TrustStoreType		JKS	The type of Java based store this is (your Java installation must have support for it).
TurstStorePwd			Password to access trust store (if necessary).

The following example is of a Java Property file that BI Presentation Services uses.

NOTE: The double back-slashes are required for file locations on Windows.

```
# The file where the private keys and their associated certificates are maintained
KeyStore = D:\\jks\\private.keystore

# Password to access KeyStore
KeyStorePwd = password

# What type of keystore this is (your Java runtime must support it)
KeyStoreType = JKS

# The alias of the key/certificate you wish to retrieve from the store
KeyAlias = obips

# The password for the key/certificate you wish to retrieve
# Defaults to the value of KeyStorePwd
KeyPwd = obips

# The file where the trusted CA certificates are maintained
TrustStore = D:\\jks\\trust.keystore

# What type of trust store this is (your Java runtime must support it)
# TrustStoreType = JKS

# Password to access TrustStore (if necessary)
# TrustStorePwd =
```

Custom Store

The BI Presentation Services Credential Store supports the storage of keys and secrets in custom storage systems. Administrators must write a program or shell script that extracts the credential data from the custom store and writes them to standard output in a well-defined text based interchange format. This custom loader must be specified in the commandLine attribute of the CredentialStorage element in the instanceconfig.xml file. At initialization of the BI Presentation Services Credential Store, the custom executable or script is launched and the Credential Store service creates an anonymous pipe to access the standard output of the custom program, thus ensuring that sensitive data is not written to any temporary file on disk.

The command line that should be executed to run the custom credential store loader should only consist of the path to the executable or script and any necessary arguments. Any redirection of standard output will fail as BI Presentation Services needs standard output to read the results from. Redirection of standard input is also not supported.

The custom executable or script must extract credentials from the store of choice and write them to standard output in a well-defined text based format. Industry standard PEM (unencrypted) format must be used for X.509 keys and certificates. A proprietary format defined by BI Presentation Services must be used for username/password based credentials. The format is described below.

Stream Structure

The general format of the stream is plain text. A rough grammar is below.

NOTE: Certificates and private keys having the same alias will get grouped into one credential.

```
entity (EOL entity)*
entity      = x509entity | upwdentity
x509entity   = x509type ":" alias EOL x509body
upwdentity   = "Username Password" ":" alias EOL upwdbody
x509type   = "Key Certificate(PEM)" | "Private Key(PEM)" | "CA Certificate(PEM)"
alias      = Sequence of printable ASCII characters, excluding whitespace.
EOL      = "\r\n" | "\n"
x509body   = PEM encoded contents
upwdbody   = username EOL password
username   = Sequence of printable ASCII characters excluding EOL
password   = Sequence of printable ASCII characters excluding EOL
```

Stream Example

The following example contains entries for:

- An X.509 credential (key and certificate) with alias "obiserver"
- An X.509 CA certificate with alias "obica"
- A username password credential for "testuser"

```
Key Certificate(PEM):obiserver
-----BEGIN CERTIFICATE-----
MIIEOTCCA6KgAwIBAgIBCTANBgkqhkiG9w0BAQQFADBXMQ8wDQYDVQQDEwZTQVcg
Q0ExCzAJBgNVBAYTAlVTMRIwEAYDVQQIEwlNaW5uZXNvdGExDzANBgNVBAoTBlNp
ZWJlbDESMBAGA1UECxMJQW5hbHl0aWNzMB4XDTA2MDYwMjE1NTYwNFoXDTA3MDYw
```

```
MjE1NTYwNFowYTELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAk1OMRQwEgYDVQQHEwtC
bG9vbWluZ3RvbjEPMA0GA1UEChMGT3JhY2xlMQswCQYDVQQLEwJCSTERMA8GA1UE
AxMISmF2YUhvc3QwggG3MIIBLAYHKoZIzjgEATCCAR8CgYEA/X9TgR11EilS30qc
Luzk5/YRt1I870QAwx4/gLZRJmlFXUAiUftZPY1Y+r/F9bow9subVWzXgTuAHTRv
8mZgt2uZUKWkn5/oBHsQIsJPu6nX/rfGG/g7V+fGqKYVDwT7g/bTxR7DAjVUE1oW
kTL2dfOuK2HXKu/yIgMZndFIAccCFQCXYFCPFSMLzLKSuYKi64QL8Fgc9QKBgQD3
4aCF1ps93su8q1w2uFe5eZSvu/o66oL5V0wLPQeCZ1FZV4661FlP5nEHEIGAtEkW
cSPoTCgWE7fPCTKMyKbhPBZ6i1R8jSjgo64eK7OmdZFuo38L+iE1YvH7YnoBJDvM
pPG+qFGQiaiD3+Fa5Z8GkotmXoB7VSVkAUw7/s9JKgOBhAACgYAYbpdoPpMmgPeF
yw+TySZWxJFnOGEF4Qdu36Iogfixv2keMo+/c18hXnT2gs2wM4fDfegKzQMQ164Z
lBkiUtNoI1wRuwNpAm/adgr6ndewJl/mCWzHLtbdvu0v8HOjrRf8xvYBmLokDREI
wsnmBCmJ516eNE/dpOHFjm7a5OCidqOB8TCB7jAMBgNVHRMBAf8EAjAAMB0GA1Ud
DgQWBBRp7oqPTp08t7AXJnLn2gK3JT65pjB/BgNVHSMEeDB2gBRMIr9mpa2SZddi
dva/y449nmhVHaFbpFkwVzEPMA0GA1UEAxMGU0FXIENBMQswCQYDVQQGEwJVUzES
MBAGA1UECBMJTWlubmVzb3RhMQ8wDQYDVQQKEwZTaWViZWwxEjAQBgNVBAsTCUFu
YWx5dGljc4IBATALBgNVHQ8EBAMCBeAwEQYJYIZIAYb4QgEBBAQDAgZAMB4GCWCG
SAGG+EIBDQQRFg94Y2EgY2VydGlmaWNhdGUwDQYJKoZIhvcNAQEEBQADgYEAV1eJ
DfkWe2sXnU0ozMWXnBXPE6wpXsjDwEqdwc8ELK7GjUZmpbggZtkzGo9lR53SZ10S
41FoSqoBfYR7CijnMCpvEUhAq1/EpMHd3dqqm9o=
-----END CERTIFICATE-----
Private Key(PEM):obiserver
-----BEGIN PRIVATE KEY-----
MIIBSwIBADCCASwGByqGSM44BAEwggEfAoGBAP1/U4EddRIpUt9KnC7s5Of2EbdS
PO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/xfW6MPbLm1Vs14E7gB00b/JmYLdrmVCl
pJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E+4P208UewwI1VBNaFpEy9nXzrith
1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/BYHPUCgYEA9+GghdabPd7L
vKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT+ZxBxCBgLRJFnEj6EwoFhO3
zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx+2J6ASQ7zKTxvqhRkImo
g9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoEFgIUHiP33wBqFCtvYwfEcPskjJY4kIc=
-----END PRIVATE KEY-----
CA Certificate(PEM):obica
-----BEGIN CERTIFICATE-----
MIIDGTCCAoKgAwIBAgIBATANBgkqhkiG9w0BAQUFADBXMQ8wDQYDVQQDEwZTQVcg
Q0ExCzAJBgNVBAYTAlVTMRIwEAYDVQQIEwlNaW5uZXNvdGExDzANBgNVBAoTBlNp
ZWJlbDESMBAGA1UECxMJQW5hbHl0aWNzMB4XDTA2MDEzMDIzMDAzMFoXDTE2MDEy
ODIzMDAzMFowVzEPMA0GA1UEAxMGU0FXIENBMQswCQYDVQQGEwJVUzESMBAGA1UE
CBMJTWlubmVzb3RhMQ8wDQYDVQQKEwZTaWViZWwxEjAQBgNVBAsTCUFuYWx5dGlj
czCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAn5uiQ/YRu76ABfnKyHvIDOUt
dbZpwGiqA7VVRQcr/RJN5Yi1Nn5W2jF/n3C3MFnHX93u1V4+kevwoN631FaARwcg
PjaTtqzCbQRsqY+QBCRdB1I/+sifsgIqnOOO/tZCl9iVVU0IhZArDQeTX+9/6/bc
DgQWBBRMIr9mpa2SZddidva/y449nmhVHTB/BgNVHSMEeDB2gBRMIr9mpa2SZddi
dva/y449nmhVHaFbpFkwVzEPMA0GA1UEAxMGU0FXIENBMQswCQYDVQQGEwJVUzES
MBAGA1UECBMJTWlubmVzb3RhMQ8wDQYDVQQKEwZTaWViZWwxEjAQBgNVBAsTCUFu
YWx5dGljc4IBATALBgNVHQ8EBAMCAYYwEQYJYIZIAYb4QgEBBAQDAgAHMB4GCWCG
SAGG+EIBDQQRFg94Y2EgY2VydGlmaWNhdGUwDQYJKoZIhvcNAQEFBQADgYEAT5rs
vNppAZCxjKmZxpyGlPZwCXHtV8y0RIAH42gs1uE4HkhMElkiirKbYdvQq1o8t8P6
DFjGCdXBABgLUx69Uwcv/1K8LHtnLCSYoiNf/ka8Y8qaIv74wnGAAysfLm8gISP7
YxoqnJKqVbCVqtyKR29RTA+YAwNPpbvw083qpRw=
-----END CERTIFICATE-----
Username Password:testuser
testuser
testpassword
```

6 Enabling Secure Communication in Oracle Business Intelligence

The SSL Everywhere feature of Oracle Business Intelligence allows communications that occur between the different BI components to be made secure. This chapter describes how to configure Oracle BI components to communicate over the Secure Socket Layer.

SSL (Secure Socket Layer) is a secured communication protocol mainly used in communications over the TCP/IP network protocol suite. By default, components of Oracle BI communicate with each other using TCP/IP. To support secured network communication, the Oracle BI components need to be configured for SSL.

Oracle BI components can communicate only through one protocol at a time. To enable SSL, you must configure each of the Oracle Business Intelligence components listed below to communicate over SSL. You must configure all instances of these components that occur in your Oracle BI deployment.

The Oracle BI components that are enabled for communication over SSL are:

- Oracle BI Server
- Oracle BI Presentation Services
- Oracle BI JavaHost
- Oracle BI Presentation Services Plug-in (Java Servlet or ISAPI)
- Oracle BI Scheduler
- Oracle BI Job Manager
- Oracle BI Cluster Controller
- Oracle BI Server Clients (For example, Oracle BI ODBC Client)

SSL requires the server to possess a public key and a private key for session negotiation. The public key is made available through a server certificate. The certificate also contains information that identifies the server. The private key is protected by the server.

The SSL Everywhere feature supports mutually-authenticated SSL and server-only authentication. Mutual authentication requires the two BI components involved in communications to both posses certificates that identify the entities. Server-only authentication mode requires only the BI component acting as the server to possess a certificate.

NOTE: It is assumed that readers are familiar with Public Key Cryptography mechanisms and the use of certificates and keys for data encryption and authentication. This guide does not explain these concepts.

Process for Enabling Secure Communication for Oracle BI Components

The process for enabling secure communication for the components of Oracle Business Intelligence consists of the following:

- Generating certificates and keys
- Configuring SSL parameters for each Oracle BI component in order for communication to occur over SSL.

NOTE: To configure your Web Server or Application Server to use the HTTPS protocol, consult your vendor documentation for instructions.

The Oracle Application Server can be set to use HTTPS protocol during installation using the Advanced option. For more information on configuring the Oracle HTTP Server for SSL, see the chapter on enabling SSL for Oracle HTTP Server in the *Oracle HTTP Server Administrator's Guide 10g (10.1.3.1.0)*. The SSL Configuration Tool allows configuration of the Oracle Application Server for HTTPS after the installation of Oracle Application Server. Refer to the *Oracle Application Server Administrator's Guide* 10g Release 3 (10.1.3.1.0) for more information.

Creating Certificates and Keys

For secure communication to occur between Oracle BI components, the BI component acting as the server must possess a public key and a private key for session negotiation. A server certificate provides the public key and server identity information to the client Oracle BI component. If client authentication is to be enabled, then the client Oracle BI component must possess a client certificate and private and public keys.

Public and private keys may be generated using toolkits such as OpenSSL. The tools also generate a certificate request to be signed by a commercial Certificate Authority (CA) such as Verisign (http://www.verisign.com) or Thawte (http://www.thawte.com). The CA issues a certificate and signs it using its private key.

To configure SSL you will require a server certificate (issued and signed by a trusted CA), a server public key and a private key. If client components are to be authenticated, then you will require a client certificate (issued and signed by a CA), a client public key and private key. The supported file formats are .pem, .cer (PEM encoding) and .der.

Oracle Business Intelligence provides an executable called openssl along with a configuration file that can be used to create certificate requests and keys. The openssl executable is based on OpenSSL. OpenSSL uses the PEM file format to store certificates and keys. The certificate request can be submitted to an outside CA. For testing purposes, the certificate requests may be signed using the root Certificate Authority generated by the executable.

For BI components such as BI Javahost and the BI Presentation Services Plug-in (Java Servlet) that are Java based, a Java certificate store must be created that contains all key and certificate data.

The following are the steps to producing a certificate:

- "Generating Certificates and Keys Using openssl" on page 85
- "Creating the Certificate Authority (CA) Certificate" on page 85
- "Generating the Hash Version of the CA Certificate File" on page 87
- "Generating Server Certificate and Server Private Key" on page 87
- "Creating the Client Certificate and Client Private Key" on page 89
- "Creating Passphrase Files or Passphrase-Producing Programs for Server and Client Keys" on page 90
- "Creating the Java Keystore" on page 91

Generating Certificates and Keys Using openssl

Before using the openssl executable provided with Oracle BI to generate certificates and keys, a directory structure must be set up as shown in the following procedure.

To generate certificates and keys using openssl

1. Create a directory where all the certificates will be located. For example, D:\ssl. This directory will be referred to as $DIR.
2. Create the following folders under $DIR:

   ```
   private
   newcerts
   demoCA
   ```

3. Copy the openssl.exe and openssl.cnf file to $DIR.

   NOTE: The openssl.exe and openssl.cnf files are located in the OracleBI_HOME\server\Bin directory and also in the OracleBI_HOME\web\bin directory.

4. Create an empty file called .oid under $DIR\demoCA.
5. Create an empty file called index.txt under $DIR.
6. Create a file called serial under $DIR. In this file, input any number that has an even number of digits. For example, numbers such as 01 or 73 are valid entries, while the numbers 1 and 173 are not.

   NOTE: The openssl executable commands outlined in the following procedures must be executed in the directory $DIR.

Creating the Certificate Authority (CA) Certificate

To create the CA, follow the procedure below.

To create the CA

- Create a Certificate Authority (CA) certificate by running the following command:

  ```
  req -new -x509 -keyout private/cakey.pem -out cacert.pem -config openssl.cnf -
  days $ValidityPeriod
  ```

  For example:

  ```
  OpenSSL> req -new -x509 -keyout private/cakey.pem -out cacert.pem -config
  openssl.cnf -days 365

  This example generates the following dialog:

  Loading 'screen' into random state - done
  Generating a 1024 bit RSA private key
  ..++++++
  .............................................++++++
  writing new private key to 'private/cakey.pem'
  Enter PEM pass phrase:
  Verifying - Enter PEM pass phrase:
  -----
  You are about to be asked to enter information that will be incorporated into your
  certificate request.
  What you are about to enter is what is called a Distinguished Name or a DN.
  There are quite a few fields but you can leave some blank
  For some fields there will be a default value,
  If you enter '.', the field will be left blank.

  -----
  Country Name (2 letter code) [AU]:
  State or Province Name (full name) [Some-State]:
  Locality Name (eg, city) []:
  Organization Name (eg, company) [Some-Organization Pty Ltd]:
  Organizational Unit Name (eg, section) []:
  Common Name (eg, YOUR name) []:
  Email Address []:
  ```

- Make a note of the passphrase that you entered. This passphrase is used when signing a new request.

- Enter a Distinguished Name as prompted. This DN identifies the Certificate Authority.

This generates a Certificate Authority (CA) certificate named cacert.pem. This certificate verifies the certificates signed by the private key. The validity period for the CA certificate generated in the above example is 365 days.

The cakey.pem file stores the private key and is generated in $DIR\private. This key is used to sign certificate requests.

Generating the Hash Version of the CA Certificate File

For enhanced security, hash the cacert.pem file that was generated in the topic "Generating the Hash Version of the CA Certificate File" on page 87.

To generate the hash version of the CA certificate file

- Run the following command:

  ```
  OpenSSL> x509 -hash -in cacert.pem
  ```

  NOTE: When you execute the hash command, you will see a number in the screen. Note this hash value. Rename the cacert.pem file to <hashvalue>.0

Generating Server Certificate and Server Private Key

The following procedures generate the server certificate and server private key that BI components acting as servers must possess. The server certificate and private key is used by Oracle BI Cluster Controller, Oracle BI Server, Oracle BI Scheduler, Oracle BI Presentation Services and Oracle BI Presentation Services Plug-in (ISAPI) components.

Generating Server Certificate Request and Private key

Use the following procedure to generate the server certificate request and private key.

To generate the server certificate request and private key

- Run the following command:

  ```
  req -new -keyout $ServerKeyFilename -out $ServerRequestFilename -days
  $ValidityPeriod -config openssl.cnf
  ```

  For example:

  ```
  OpenSSL> req -new -keyout server-key.pem -out server-req.pem -days 365 -config
  openssl.cnf
  ```

  This example generates the following dialog:

  ```
  Loading 'screen' into random state - done
  Generating a 1024 bit RSA private key
  ............................++++++
  ...........................................................++++++
  writing new private key to 'server-key.pem'
  Enter PEM pass phrase:
  Verifying - Enter PEM pass phrase:
  -----

  You are about to be asked to enter information that will be incorporated into your
  certificate request.
  What you are about to enter is what is called a Distinguished Name or a DN.
  There are quite a few fields but you can leave some blank
  ```

```
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Some-Organization Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, YOUR name) []:
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

- Enter a Distinguished Name as prompted. The Distinguished Name identifies the server.
- Make a note of the passphrase that you entered. This passphrase is needed to decrypt the private key.

The command generates the server private key file called server-key.pem and the certificate request (unsigned server certificate) called server-req.pem.

Creating the Server Certificate

The certificate request created above can be submitted to a commercial CA to generate a server certificate. For testing purposes, the CA generated in the step "Creating the Certificate Authority (CA) Certificate" on page 85 can be used to sign the request, as described in the following procedure.

To create the server certificate

- Run the following command:

```
ca -policy policy_anything -out $ServerCertFilename -config openssl.cnf -infiles
$ServerRequestFilename
```

For example:

```
Openssl>ca -policy policy_anything -out server-cert.pem -config openssl.cnf -
infiles server-req.pem
```

For this example, the following dialog is received:

```
Using configuration from openssl.cnf
Loading 'screen' into random state - done
Enter pass phrase for ./private/cakey.pem:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'US'
stateOrProvinceName   :PRINTABLE:'CA'
localityName          :PRINTABLE:'Redwood Shores'
organizationName      :PRINTABLE:'Oracle'
organizationalUnitName:PRINTABLE:'BI'
```

```
commonName            :PRINTABLE:'Server Certificate'
Certificate is to be certified until Dec 29 07:06:45 2007 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

- When prompted, enter the passphrase for the private key of the CA.

  This is the passphrase that was supplied when creating the private key cakey.pem in the topic "Creating the Certificate Authority (CA) Certificate" on page 85.

This command generates the server certificate named server-cert.pem. The private key of the CA was used to sign the request. The public key is generated and placed in $DIR\newcerts with a filename that reflects the serial number, for example, 01.pem.

The server certificate and private key is used by Oracle BI Cluster Controller, Oracle BI Server, Oracle BI Scheduler, Oracle BI Presentation Services and Oracle BI Presentation Services Plug-in (ISAPI) components.

Creating the Client Certificate and Client Private Key

For mutually-authenticated SSL where the client BI component identity is verified by the server, the client must possess a certificate and private key. Use the following procedures to generate a client certificate and client private key. The client certificate and private key is used by the BI Server Client components such as Oracle BI ODBC client.

To create the client certificate request and private key

- Create the client request and private key by running the following commands:

  ```
  req -new -keyout $ClientKeyFilename -out $ClientRequestFilename -days
  $ValidityPeriod -config openssl.cnf
  ```

  For example:

  ```
  OpenSSL> req -new -keyout client-key.pem -out client-req.pem -days 365 -config
  openssl.cnf
  ```

This example generates the client private key in the file client-key.pem and the signing request or unsigned client certificate client-req.pem.

To create the client certificate

- Create the client certificate by running the following command:

  ```
  ca -policy policy_anything -out $ClientCertFilename -config openssl.cnf -infiles
  $ClientRequestFilename
  ```

  For example:

```
OpenSSL>ca -policy policy_anything -out client-cert.pem -config openssl.cnf -
infiles client-req.pem
```

This example generates the signed client certificate client-cert.pem.

The client certificate and private key is used by the BI Server Client components such as Oracle BI ODBC client.

Creating Passphrase Files or Passphrase-Producing Programs for Server and Client Keys

To create the passphrase files or passphrase-producing programs for server and client keys, use the following procedure.

To create passphrase files of passphrase-producing programs

1. Server passphrase file:
   - Under $DIR, create a passphrase file called serverpwd.txt.
   - In this file, input the passphrase that was used to encrypt the server private key in the topic "Generating Server Certificate and Server Private Key" on page 87.
2. Client passphrase file:
   - Under $DIR, create a passphrase file called clientpwd.txt.
   - In this file, input the passphrase that was used to encrypt the client private key in the topic "Creating the Client Certificate and Client Private Key" on page 89.
3. Optionally, create an executable program that outputs the server private key pass phrase in a standard output.

   Place this program under a directory that the system can find, for example, OracleBI_HOME\server\Bin.

Creating the Java Keystore

For BI components that are Java-based, a Java certificate store must be created that contains certificates and key files.

This procedure creates a Java Keystore that can be used for Oracle BI Job Manager, Oracle BI Presentation Services, Oracle BI Javahost and Oracle BI Presentation Services Plug-in (Java Servlet). This keystore stores the certificate and private key used by these components.

The keystore is generated and managed using the keytool command-line executable that ships with JDK. For more information on Java keystores and the keytool utility, refer to the JDK documentation.

Generating the Private Key

To generate the private key, use the following procedure.

To generate the private key

- Use the genkey subcommand with inputs as shown:

  ```
  keytool -genkey -v -alias jobmanagerkey -keyalg rsa -keysize 1024 -validity 365
  -keystore jobmanager.keystore -storepass analytics
  ```

  In this example, the keystore called jobmanager.keystore stores the private key with an alias of jobmanagerkey and with a password of "analytics".

  The alias and password values are referenced when setting SSL-related parameters for Oracle BI components.

Generating the Certificate

To generate the certificate, use the following procedure.

To generate the client certificate

- Use the certreq subcommand with the inputs as shown:

  ```
  keytool -certreq -v -alias jobmanagerkey -file certreq.txt -keystore
  jobmanager.keystore -storepass analytics
  ```

The certificate request must be signed by a CA, as shown in the following procedure.

Signing the Client Certificate

For testing purposes, the same Certificate Authority cacert.pem that was created in topic "Creating the Certificate Authority (CA) Certificate" on page 85 can be used.

To sign the certificate request using the openssl utility and the cacert.pem Certificate Authority

1. In OpenSSL, run the following command:

   ```
   ca -policy policy_anything -out jobmanager-cert.cer -config ssl.cnf -infiles
   certreq.txt.
   ```

   Opening this file creates a certificate called jobmanager-cert.cer.

2. Change the certificate file jobmanager-cert.cer into an X509 file:

   Remove all lines that appear before the text ----BEGIN CERTIFICATE---- and after the text ----END CERTIFICATE----.

   The certificate file should be similar to the following example:

   ```
   -----BEGIN CERTIFICATE-----
   MIICcjCCAdugAwIBAgIBEjANBgkqhkiG9w0BAQQFADBkMQswCQYDVQQGEwJVUzEL
   MAkGA1UECBMCQ0ExEjAQBgNVBAcTCVNhbiBNYXRlbzEPMA0GA1UEChMGU2llYmVs
   MRIwEAYDVQQLEwlBbmFseXRpY3MxDzANBgNVBAMTBkNBQ2VydDAeFw0wNjAzMDYx
   OTU2NTFaFw0wNzAzMDYxOTU2NTFaMGcxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJD
   QTESMBAGA1UEBxMJU2FuIE1hdGVvMQ8wDQYDVQQKEwZTaWViZWwxEjAQBgNVBAsT
   CUFuYWx5dGljczESMBAGA1UEAxMJSm9iTWFuZ2VyMIGfMA0GCSqGSIb3DQEBAQUA
   A4GNADCBiQKBgQDBIR0ATYE6UtEL4W/bQ1xPIHsT7S5EcmfpezZQCeumBSg00/5U
   nIYFfPHBjcjgJKChVG+DSZRxPJOAifLeKTb6pk3lHoJJ9Gr/HuryYOc46Efd/q0+
   cXMQ+fPC+0M/0caohryfSAjcW+O0IwycHjbmj4VXc4L4OTgRHIQv0oVRkQIDAQAB
   ozEwLzAtBglghkgBhvhCAQ0EIBYeR2VuZXJhdGVkIHdpdGggUlNBIEJTQUZFIFNT
   TC1DMA0GCSqGSIb3DQEBBAUAA4GBAC81Hi77GChZYyiYXTNIN0oEVS4CiKpAMUg9
   55QMaQU/RsdWYe8ne34EpXDWY6LVdBi8nxL41l/VLckM2Gbn72LZx5KelzPuzgwy
   qC8Z4HGOvjAzYHA8AQhRHaFSYWZoUkbay/6/8bYofJJarkjD68rdz1i0m9L3/sWM
   MmEQmHNo
   -----END CERTIFICATE-----
   ```

Importing the Certificate Authority File to a Java Keystore

The Certificate Authority (CA) certificate that was used to sign the certificate request as described in the topic "Generating the Certificate" on page 91 must be imported to a Java keystore. Use the keytool utility as shown in the following procedure.

To import the certification authority file to the Java keystore

- In keytool, import the CA certificate to the keystore database.

  NOTE: For enhanced security, create a new keystore database named, for example, trust.keystore and import the CA certificate into this keystore.

- In the following example, the CA certificate cacert.pem is imported to the same keystore that contains the certificate and key, jobmanager.keystore.

  ```
  keytool -import -keystore jobmanager.keystore -storepass analytics -alias
  cacertificates -file cacert.pem
  ```

Importing the Certificate to the Java Keystore

The certificate jobmanager-cert.cer, created in the topic "Generating the Certificate" on page 91, must be imported to the Java keystore.

To import the certificate to the Java keystore

- Using the keytool utility, run the command:

```
keytool -import -keystore jobmanager.keystore -storepass analytics -alias
jobmanagerkey -file jobmanager-cert.cer
```

Configuring Oracle Business Intelligence to Communicate Over SSL

The components of Oracle BI are configured to communicate over SSL by setting SSL-related parameters.

Table 10 on page 93 provides a description of the parameters and example values used when configuring the BI components for SSL.

Table 10. SSL Parameters Used by Oracle BI Components

Parameter	Description
Certificate File	The certificate file. For components acting as SSL servers such as BI Server and BI Scheduler, this is the Server Certificate filename. For example, server-cert.pem. For client components, such as BI ODBC Client Data Source, this is the Client Certificate filename. For example, client-cert.pem.
Private Key File	The private key file. For server components, this is Server Private Key filename. For example, server-key.pem. For client components, this is the Client Private Key filename. For example, client-key.pem.
Passphrase File or Passphrase Program	Used to obtain the passphrase needed to decrypt the private key. Specify either a file containing the passphrase or a program that outputs the passphrase.
CA Certificate File or CA Certificate Directory	These two parameters reference the CA certificate file. The CA is used to verify the server or client certificate when Verify Peer is set to true. Set either the CA Certificate File or CA Certificate Directory parameter. The CA Certificate File parameter specifies the name and path of the trusted CA Certificate. The CA Certificate Directory contains hash versions of trusted CAs.

Table 10. SSL Parameters Used by Oracle BI Components

Parameter	Description
Verify Peer	When set to true, the BI component verifies that the other component to the connection has a valid certificate (that is, mutual authentication). A value of false permits a connection to any peer.
Certificate Verification Depth	The depth of certificate chain. A depth of one means a certificate has to be signed by one of the trusted CAs. A depth of two means the certificate was signed by a CA that was further verified by one of the CAs.
Trusted Peer Distinguished Names	Used to specify individual named clients (by Distinguished Name) that are allowed to connect. DN identifies the entity that holds the private key that matches the public key of the certificate.
Cipher Status	A list of cipher suites that should be permitted. See OpenSSL documentation. For example, SSL_CIPHER_LIST="EXP-DES-56-SHA";

Minimum Security and Near-Maximum Security Scenarios

Two configuration scenarios are defined:

- **Minimum security scenario.**

  For server components such as Oracle BI Server or Oracle BI Cluster Controller, the minimum security scenario satisfies the following conditions:

  - Enable SSL is set to true.
  - The parameters for Certificate, Private Key file and either passphrase file or passphrase-producing program are also set. The Certificate, private Key file, and passphrase file (or program) are located on the machine.

  For Client components such as BI ODBC Client, minimum security scenario is when the parameter to enable SSL is set to true.

- **Near-Maximum security scenario.**

  For Server components, near-maximum security scenario satisfies the following conditions, in addition to the settings in minimum security scenario:

  - Certificate Authority File parameter or the parameter specifying the directory containing the hashed version of the CA is set.
  - Peer Verification is set to true, and Trusted Peer Distinguished Names are provided.
  - A Certification Verification Depth of 1 is specified. The CA is also located on the machine.

  For Client components, near-maximum security scenario conditions are the following:

  - SSL parameter is set to true.
  - The parameters for Client Certificate, Client Key file, and passphrase file are set.

- Either CA File parameter or the parameter specifying the directory containing the hashed version of CA is set.
- Peer Verification is set to true.
- Trusted Peer Distinguished Names are provided.
- A Certification Verification Depth of 1 is specified.
- The Certificate, private Key file, passphrase file and CA are located on the machine.

NOTE: It is highly recommended that you first configure your Oracle BI deployment for functionality and ensure that all Oracle BI components are operational and functional, including BI Publisher if you are using the Oracle BI Reporting and Publishing feature, before you enable communication of BI components to occur over SSL. Determine whether you wish to implement the minimum or maximum security scenario.

The configuration tasks are for configuring a single instance of each BI component. If you have multiple instances of a BI component in your deployment, perform the configuration for all instances of each component. Alternately, you may configure one instance of a BI component and copy the configuration files and certificates, keys, and stores as appropriate to other instances, and perform machine-specific changes to the configuration file if needed.

NOTE: Before performing the configuration, stop all BI services and processes. Restart services and processes after configuration is complete for the changes to take effect.

Configuring Oracle BI Cluster Controller

Skip this section if you have not deployed the BI Cluster Controller.

The process of configuring Oracle BI Cluster Controller to communicate over SSL consists of modifying parameters in the NQClusterConfig.INI file. This file is located in one of the following directories:

- Windows: OracleBI_HOME\server\config
- Linux or UNIX: OracleBI_HOME/server/config

Perform this configuration on all machines where BI Cluster Controller has been deployed, or copy the modified NQClusterConfig.INI to the secondary cluster controller machine.

This topic consists of the following sub-topics:

- "Configuring Oracle BI Cluster Controller in Minimum Security Scenario." Use this procedure if you are deploying Oracle BI with minimum security.
- "Configuring Oracle BI Cluster Controller in Near-Maximum Security Scenario." Use this procedure if you are deploying Oracle BI with near maximum security.

Configuring Oracle BI Cluster Controller in Minimum Security Scenario

On the machine where the Oracle BI Cluster Controller has been installed, modify the NQClusterConfig.INI file as described in the following procedure.

To configure the cluster controller for minimum security in NQClusterConfig.INI

1 Open the NQClusterConfig.INI file for editing. Locate the following lines:

```
#SSL=NO;
#SSL_CERTIFICATE_FILE="servercert.pem";
#SSL_PRIVATE_KEY_FILE="serverkey.pem";
#SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
```

2 Uncomment these lines and change the settings as follows:

```
SSL=YES;
SSL_CERTIFICATE_FILE="<Server Certificate Filename>";
SSL_PRIVATE_KEY_FILE="<Server Private Key Filename>";
SSL_PK_PASSPHRASE_FILE="<passphrase file>";
```

After modification, the NQClusterConfig.INI file should be similar to the following text:

```
SSL=YES;
SSL_CERTIFICATE_FILE="server-cert.pem";
SSL_PRIVATE_KEY_FILE="server-key.pem";
SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
```

3 Copy the server certificate, private key, and passphrase file.

For example, copy the server-cert.pem, server-key.pem and serverpwd.txt files to OracleBI_HOME\server\Config or to OracleBI_HOME/server/Config.

Configuring Oracle BI Cluster Controller in Near-Maximum Security Scenario

On the machine where the Oracle BI Cluster Controller has been installed, modify the NQClusterConfig.INI file as described in the following procedure.

To configure the cluster controller for near-maximum security in NQClusterConfig.INI

1 Open the NQClusterConfig.INI file for editing. Locate the following lines:

```
#SSL=NO;
#SSL_CERTIFICATE_FILE="servercert.pem";
#SSL_PRIVATE_KEY_FILE="serverkey.pem";
#SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
#SSL_PK_PASSPHRASE_PROGRAM="sitepwd.exe";
#SSL_VERIFY_PEER=NO;
```

```
#SSL_CA_CERTIFICATE_DIR="CACertDIR";
#SSL_CA_CERTIFICATE_FILE="CACertFile";
#SSL_TRUSTED_PEER_DNS="";
#SSL_CERT_VERIFICATION_DEPTH=9;
#SSL_CIPHER_LIST="";
```

2 Uncomment these lines and change the settings as follows:

```
SSL=YES;
SSL_CERTIFICATE_FILE="<Server Certificate Filename>";
SSL_PRIVATE_KEY_FILE="<Server Private Key Filename>";
```

3 Set one of the following lines, depending on if you are using a passphrase file or a passphrase program:

- If you are using a passphrase file, uncomment and set the line:

  ```
  SSL_PK_PASSPHRASE_FILE="<passphrase file>";
  ```

- If you are using a passphrase program, uncomment and set the line:

  ```
  SSL_PK_PASSPHRASE_PROGRAM="<passphrase-producing program>";
  ```

4 Uncomment and set the following line:

```
SSL_VERIFY_PEER=YES;
```

When this parameter is set to YES, Oracle BI Server clients must provide valid certificates that will be verified by a trusted CA.

5 Set one of the following lines, depending on if you are using the Certification Authority (CA) certificate file or the hashed version of the CA certificate:

- If you are using the CA certificate file, uncomment and set the line:

  ```
  #SSL_CA_CERTIFICATE_FILE="<Certificate Authority Certificate filename>";
  ```

- If you are using the hashed version of the CA certificate, uncomment and set the line:

  ```
  SSL_CA_CERTIFICATE_DIR="OracleBI_HOME\ssl";
  ```

The directory specified must contain the hash version of the CA certificate.

6 Uncomment and set the following line:

```
SSL_TRUSTED_PEER_DNS="";
```

The DNS identifies the clients allowed to connect. The DNS entry can be empty, multiple, or part of one Distinguished Name.

7 Uncomment and set the following line:

```
SSL_CERT_VERIFICATION_DEPTH=<value>;
```

8 Uncomment and set the following line:

```
SSL_CIPHER_LIST="";
```

After modification, the SSL portion of the NQClusterConfig.INI file should be similar to:

```
SSL=YES;
SSL_CERTIFICATE_FILE="server-cert.pem";
SSL_PRIVATE_KEY_FILE="server-key.pem";
#SSL_PK_PASSPHRASE_FILE="serverpwd.”

# Line above commented out since passphrase program is used
SSL_PK_PASSPHRASE_PROGRAM="passphrase.exe”;
SSL_VERIFY_PEER=YES;
#SSL_CA_CERTIFICATE_DIR="CACertDIR";

# Line above commented out since certificate file is used
SSL_CA_CERTIFICATE_FILE="cacert.pem";
SSL_TRUSTED_PEER_DNS="C=US/ST=CA/L=Redwood Shores/O=Oracle/OU=BI/
CN=clientcertificate”; SSL_CERT_VERIFICATION_DEPTH=1;
SSL_CIPHER_LIST=" EXP-DES-56-SHA";
```

9 Copy the server certificate, private key, and passphrase file or program.

For example, copy the server-cert.pem, server-key.pem and serverpwd.txt files to OracleBI_HOME\server\Config or to OracleBI_HOME/server/Config.

10 If you have specified the CA Certificate File parameter, also copy the CA certificate file to this same location, and copy the hash version of the CA certificate to the directory specified.

Configuring Oracle BI Server for Communication Over SSL

The process of configuring Oracle BI Server to communicate over SSL consists of modifying parameters in the NQSConfig.INI file. This file is located in OracleBI_HOME\server\Config. On Linux or UNIX, this file is located in the OracleBI_HOME/server/Config directory. Perform this configuration on all machines where Oracle BI Server has been deployed, or copy the modified NQSConfig.INI to other Oracle BI servers, editing any machine-specific information such as file paths.

This topic consists of the following sub-topics:

- “Configuring Oracle BI Server in Minimum Security Scenario.” Use this procedure if you are deploying Oracle BI with minimum security.
- “Configuring Oracle BI Server in Near-Maximum Security Scenario.”Use this procedure if you are deploying Oracle BI with near maximum security.

NOTE: If your BI Server uses ODBC to connect to the database, then data including passwords may be sent in plain text over TCP/IP. To secure this communication, refer to manufacturer documentation for the ODBC driver.

Configuring Oracle BI Server in Minimum Security Scenario

Use the following procedure to configure Oracle BI Server in a minimum security scenario.

To configure Oracle BI Server in minimum security scenario

1 On the machine where the Oracle BI Server has been installed, modify the NQSConfig.INI file as described below. This file is located in OracleBI_HOME\server\Config. On UNIX, this file is located in the OracleBI_HOME/server/Config directory.

2 Open the NQSConfig.INI file for editing.

3 Locate the following lines:

```
#SSL=NO;
#SSL_CERTIFICATE_FILE="servercert.pem";
#SSL_PRIVATE_KEY_FILE="serverkey.pem";
#SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
```

4 Uncomment the above lines and set the parameters as follows:

```
SSL=YES;
SSL_CERTIFICATE_FILE="<Server Certificate Filename>";
SSL_PRIVATE_KEY_FILE="<Server Private Key Filename>";
SSL_PK_PASSPHRASE_FILE="<passphrase file>";
```

After modification, the NQSConfig.INI file should be similar to the following example:

```
SSL=YES;
SSL_CERTIFICATE_FILE="server-cert.pem";
SSL_PRIVATE_KEY_FILE="server-key.pem";
SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
```

5 Copy the server certificate, server private key, and passphrase file. For example, server-cert.pem, server-key.pem and serverpwd.txt files to OracleBI_HOME\server\Config on Windows and to OracleBI_HOME/server/Config Linux or UNIX.

Configuring Oracle BI Server in Near-Maximum Security Scenario

On the machine where Oracle BI Server has been installed, modify the NQSConfig.INI file as described in the following procedure. The NQSConfig.INI file is located in the directory OracleBI_HOME\server\Config or in the directory OracleBI_HOME/server/Config.

To configure Oracle BI Server in near-maximum security scenario

1 Open the NQSConfig.INI file for editing. Locate the following lines:

```
#SS=NO;
#SSL_CERTIFICATE_FILE="servercert.pem";
#SSL_PRIVATE_KEY_FILE="serverkey.pem";
#SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
#SSL_PK_PASSPHRASE_PROGRAM="sitepwd.exe";
#SSL_VERIFY_PEER=NO;
#SSL_CA_CERTIFICATE_DIR="CACertDIR";
#SSL_CA_CERTIFICATE_FILE="CACertFile";
#SSL_TRUSTED_PEER_DNS="";
#SSL_CERT_VERIFICATION_DEPTH=9;
#SSL_CIPHER_LIST="";
```

2 Uncomment the lines shown below and set the parameter values:

```
SSL=YES;
SSL_CERTIFICATE_FILE="<Server Certificate Filename>";
SSL_PRIVATE_KEY_FILE="<Server Private Key Filename>";
Set one of the following lines depending on whether you are using a passphrase
file or a passphrase program:
#SSL_PK_PASSPHRASE_FILE="serverpwd.txt";
#SSL_PK_PASSPHRASE_PROGRAM="sitepwd.exe";
If you are using a passphrase file, uncomment and set the line:
SSL_PK_PASSPHRASE_FILE="<passphrase file>";
If you are using a passphrase program, uncomment and set the line:
SSL_PK_PASSPHRASE_PROGRAM="<passphrase-producing program>";
```

3 Uncomment and set the following line:

```
SSL_VERIFY_PEER=YES;
```

When this parameter is set to YES, Oracle BI Server clients must provide valid certificates.

4 Set one of the following lines depending on whether you are using the Certificate Authority (CA) certificate file or the hashed version of the CA certificate:

```
#SSL_CA_CERTIFICATE_DIR="CACertDIR";
#SSL_CA_CERTIFICATE_FILE="CACertFile";
```

If you are using the CA certificate file, uncomment and set the line:

```
#SSL_CA_CERTIFICATE_FILE="<Certificate Authority Certificate filename>";
```

If you are using the hashed version of the CA certificate, uncomment and set the line:

```
SSL_CA_CERTIFICATE_DIR="OracleBI_HOME>\ssl";
```

The directory specified must contain the CA certificate named by the hash value.

5 Uncomment and set the following line:

```
SSL_TRUSTED_PEER_DNS="";
```

The DNS may be empty, multiple or part of one DN. It specifies the clients allowed to connect.

For example:

```
SSL_TRUSTED_PEER_DNS="C=US/ST=CA/L=Redwood Shores/O=Oracle/OU=BI/
CN=clientcertificate";
```

6. Uncomment and set the following line:

   ```
   SSL_CERT_VERIFICATION_DEPTH=<value>;
   ```

7. Uncomment and set the following line:

   ```
   SSL_CIPHER_LIST="";
   ```

   For example: SSL_CIPHER_LIST="EXP-DES-56-SHA";

   After modification, the SSL portion of the NQSConfig.INI file should be similar to:

   ```
   SSL=YES;
   SSL_CERTIFICATE_FILE="server-cert.pem";
   SSL_PRIVATE_KEY_FILE="server-key.pem";
   #SSL_PK_PASSPHRASE_FILE="serverpwd.” \
   # Line above commented out since passphrase program is used
   SSL_PK_PASSPHRASE_PROGRAM="passphrase.exe”;
   SSL_VERIFY_PEER=YES;
   #SSL_CA_CERTIFICATE_DIR="CACertDIR";
   # Line above commented out since certificate file is used
   SSL_CA_CERTIFICATE_FILE="cacert.pem";
   SSL_TRUSTED_PEER_DNS="C=US/ST=CA/L=Redwood Shores/O=Oracle/OU=BI/
   CN=clientcertificate”; SSL_CERT_VERIFICATION_DEPTH=1;
   SSL_CIPHER_LIST=" EXP-DES-56-SHA";
   ```

8. Copy the server certificate, private key and passphrase file or program to the directory OracleBI_HOME\server\Config or OracleBI_HOME/server/Config.

9. To the location specified in Step 8, also copy the CA certificate file if you have specified the CA Certificate File parameter.

10. If you have specified the CA Certificate Directory parameter, copy the hash version of the CA certificate to the directory specified.

Configuring Oracle BI Server Client

The following section contains information about configuring the Oracle BI Server client for minimum or near-maximum security deployment.

Configuring Oracle BI Server Client on Windows in Minimum Scenario

Use this procedure to configure Oracle BI Server Client (BI ODBC Data Source) to communicate over SSL in a minimum security deployment. It is assumed that neither the Oracle BI Cluster Controller nor the Oracle BI servers have been set to require peer verification.

To configure Oracle BI Server client on Windows in minimum scenarios

1. On the Windows machine where the Oracle BI Server Client has been installed, open the ODBC Data Source Administrator.

2. Navigate to the System DSN tab and the select Oracle Analytics Server DSN (by default called AnalyticsWeb). Click the Configure button to open the Oracle Analytics Server Configuration window.
3. Check the Use SSL check box that appears on the configuration window.

Configuring Oracle BI Server Client on Windows in Near-Maximum Security Scenario

Use this procedure to configure Oracle BI Server Client (BI ODBC Data Source) to communicate over SSL in a maximum security deployment.

To configure Oracle BI Server client on Windows in near-maximum security scenarios

1. On the Windows machine where the Oracle BI Server Client has been installed, open the ODBC Data Source Administrator.
2. Navigate to the System DSN tab and the select Oracle Analytics Server DSN (by default called AnalyticsWeb). Click the Configure button to open the Oracle Analytics Server Configuration window.
3. Check the Use SSL check box that appears on the configuration window.
4. Click the Configure SSL button to open the Secure Socket Layer Configuration dialog box.
5. In the Secure Socket Layer Configuration dialog box, enter the following:

   - In the Certificate File text box, enter the path and file name of the Client Certificate file. For example:

     ```
     Certificate File = OracleBI_HOME\ssl\client-cert.pem
     ```

   - In the Certificate Private Key File text box, enter the path and file name of the Client Private Key file. For example:

     ```
     Certificate Private Key File = OracleBI_HOME\ssl\client-key.pem
     ```

   - In the File Containing Passphrase text box, enter the path and file name of the passphrase file for the Client Key. For example:

     ```
     File Containing Passphrase = OracleBI\ssl\clientpwd.txt
     ```

   The above three entries are required when either Oracle BI Cluster Controller or Oracle BI Severs have been configured to require peer verification.

   - Check the Verify Peer check box.
   - If you are using the hashed version of the CA certificate, provide the directory where the hashed file is located in the CA Certificate Directory text box. For example:

     ```
     CA Certificate Directory = OracleBI_HOME\ssl
     ```

   - If you are using the CA certificate, provide the path and file name of the CA Certificate file in the CA Certificate File text box.

     ```
     CA Certificate File = <OracleBI>\ssl\cacert.pem
     ```

- In the Cipher List text box, enter the list of ciphers to be used. For example:

  ```
  Cipher List = EXP-DES-56-SHA
  ```

- Specify a value of 1 for Certificate Verification Depth. For example:

  ```
  Certification Verification Depth = 1
  ```

- In the Trusted Peer Distinguished Names text box, enter DNs of servers that will allowed to connect. For example:

  ```
  Trusted Peer Distinguished Names = C=US/ST=CA/L=Redwood Shores/O=Oracle/
  OU=BI/CN=servercertificate
  ```

6. Copy the client certificate, client private key and passphrase file, for example client-cert.pem, client-key.pem and clientpwd.txt to the directory specified in the parameters. In the examples specified, the directory is OracleBI_HOME\ssl. If you have set the CA Certificate File parameter, copy the CA certificate file, for example cacert.pem, to the directory specified. If you have set the CA Certificate Directory parameter, copy the hash version of the CA certificate to the directory specified.

Configuring Oracle BI Server Client on UNIX in Minimum Security Scenario

Perform this task to configure Oracle BI Server Client to communicate over SSL.

To configure Oracle BI Server Client on UNIX in minimum security scenarios

- Modify the odbc.ini file located in the OracleBI_HOME/setup directory by adding the following line to the [AnalyticsWeb] section of the file:

  ```
  SSL=YES
  ```

  In a minimum security deployment, no additional parameters for SSL need to be set. It is assumed that neither the Oracle BI Cluster Controller nor the Oracle BI servers have been configured to verify peers or have the trusted peers DNs set.

Configuring Oracle BI Server Client on UNIX in near-Maximum Security Scenario

Perform the following task to configure Oracle BI Server Client to communicate over SSL.

To configure Oracle BI Server client on UNIX in near-maximum security scenario

1. Modify the odbc.ini file located in the OracleBI_HOME/setup directory by adding the following lines to the [AnalyticsWeb] section of the file:

   ```
   SSL=YES
   SSLertificateFile=<Directory and filename of client certificate>
   SSLPrivateKeyFile==<Directory and filename of client private key file>
   SSLPassphraseFile=<Directory and filename of passphrase file for client key>
   SSLipherList=<cipher list>
   ```

```
SSLVerifyPeer=Yes
SSLTrustedPeerDNs=<Distinguished Names of trusted peers>
SSLertVerificationDepth=<Depth of chain>
```

2. If you are using the hashed version of the CA Certificate file, add the line:

   ```
   SSLACertificateDir=<Directory containing the hashed CA certificate>
   ```

3. If you are using the CA Certificate file, add the line:

   ```
   SSLACertificateFile=<Directory and filename of CA Certificate file>
   ```

   After modification, the [AnalyticsWeb] section of the odbc.ini file should have additional entries similar to the following example:

   ```
   [AnalyticsWeb]
   .
   .
   .
   SSL=YES
   SSLertificateFile=OracleBI_HOME/ssl/client-cert.pem
   SSLPrivateKeyFile=OracleBI_HOME/ssl/client-key.pem
   SSLPassphraseFile=OracleBI_HOME/ssl/clientpwd.txt
   SSLipherList= EXP-DES-56-SHA
   SSLVerifyPeer=Yes
   SSLACertificateDir=OracleBI_HOME/ssl
   SSLACertificateFile=OracleBI_HOME/ssl/cacert.pem
   SSLTrustedPeerDNs= C=US/ST=CA/L=Redwood Shores/O=Oracle/OU=BI/
   CN=servercertificate
   SSLertVerificationDepth=1
   ```

4. Copy the client certificate, client private key and passphrase file to the directory specified in the parameters.

   In the examples specified, the directory is OracleBI_HOME/ssl.

5. Copy the CA certificate file if you have set the CA Certificate File parameter to the directory specified.

6. If you have set the CA Certificate Directory parameter, copy the hash version of the CA certificate to the directory specified.

Configuring Oracle BI Scheduler

The following section contains information about configuring the Oracle BI Scheduler for minimum or near-maximum security deployment.

Configuring Oracle BI Scheduler on Windows in a Minimum Security Scenario

This topic describes the process to configure Oracle BI Scheduler installed on Windows in a minimum security deployment to communicate over SSL.

On Windows, Oracle BI Scheduler may be configured to communicate over SSL using either Oracle BI Job Manager or the schconfig command line utility.

Use this procedure to configure Oracle BI Scheduler using Oracle BI Job_Manager.

To configure Oracle BI Scheduler on Windows in a minimum security deployment

1. Launch Oracle BI Job Manager.
2. Navigate to File > Configuration Options to open the Scheduler Configuration dialog box.
3. Select Scheduler tab > Advanced tab
4. Check the Use Secure Socket Layer check box.

You may use the schconfig utility to configure Oracle BI Scheduler. The schconfig.exe executable is located in the OracleBI_HOME\server\Bin directory. Refer to the topic "Configuring Oracle BI Scheduler on UNIX in a near-Maximum Security Scenario" on page 107 for instructions on how to use this utility to configure Oracle BI Scheduler for communication over SSL.

Configuring Oracle BI Scheduler on Windows in a near-Maximum Security Scenario

This topic describes the process to configure Oracle BI Scheduler installed on Windows in a maximum security deployment for communication over SSL.

On Windows, Oracle BI Scheduler may be configured to communicate over SSL using either Oracle BI JobManager or the schconfig command line utility.

Use this procedure to configure Oracle BI Scheduler using Oracle BI Job_Manager.

To configure Oracle BI Scheduler on Windows in a near-maximum security scenario

1. Launch Oracle BI Job Manager.
2. Navigate to File > Configuration Options to open the Scheduler Configuration dialog box.
3. Select Scheduler tab > Advanced tab
4. Check the Use Secure Socket Layer check box.
5. In the SSL section of the dialog box, make the following changes:
   - In the SSL Certificate File Path text box, enter the path and file name of the Server Certificate file.

     For example:

     ```
     SSL Certificate File Path = OracleBI_HOME\ssl\server-cert.pem
     ```

   - In the SSL Certificate Private Key File text box, enter the path and file name of the Server Private Key file.

     For example,

     ```
     SSL Certificate Private Key File = OracleBI_HOME\ssl\server-key.pem
     ```

- If you are using a passphrase file, select the SSL File Containing Passphrase radio button and enter the path and file name of the passphrase file for the Server Key.

  For example,

  ```
  SSL File Containing Passphrase = OracleBI_HOME\ssl\serverpwd.txt
  ```

- If you are using a passphrase program, select the SSL Program Producing Passphrase radio button and enter the name of the passphrase producing program.

  For example,

  ```
  SSL Program Producing Passphrase = passphrase.exe
  ```

The entries made above are required when either Oracle BI Cluster Controller or Oracle BI servers have been configured to require peer verification.

- Check the SSL Require Client check box.
- Specify a value for SSL Certificate Verification Depth.
- If you are using the hashed version of the CA certificate, select the CA Certificate Directory radio button and enter the directory where the hashed file is located in the corresponding text box.

  For example,

  ```
  CA Certificate Directory = OracleBI_HOME\ssl
  ```

- If you are using the CA certificate, select the CA Certificate File radio button and enter the path and file name of the CA Certificate file in the text box.

  For example,

  ```
  CA Certificate File = <OracleBI>\ssl\cacert.pem
  ```

- In the SSL Trusted Peer Distinguished Names text box, enter the DNs of clients that will be allowed to connect.

  For example,

  ```
  Trusted Peer Distinguished Names = C=US/ST=CA/L=Redwood Shores/O=Oracle/
  OU=BI/CN=clientcertificate
  ```

- In the SSL Cipher List text box, enter the list of ciphers to be used.

  For example,

  ```
  SSL Cipher List = EXP-DES-56-SHA
  ```

6. Copy the server certificate, server private key and passphrase file or program to the directory specified in the parameters.

   In the examples, the directory is OracleBI_HOME\ssl.

7. If you have set the CA Certificate File parameter, copy the CA certificate file to the directory specified.

8. If you have set the CA Certificate Directory parameter, copy the hash version of the CA certificate to the directory specified.

You may use the schconfig utility to configure Oracle BI Scheduler. The schconfig.exe executable is located in the OracleBI_HOME\server\Bin directory. Refer to topic "Configuring Oracle BI Scheduler on UNIX in a Minimum Security Scenario" on page 107 for instructions on how to use this utility to configure Oracle BI Scheduler for communication over SSL.

Configuring Oracle BI Scheduler on UNIX in a Minimum Security Scenario

This topic describes the process to configure Oracle BI Scheduler installed on UNIX in a minimum security deployment for communication over SSL. Use this procedure to configure Oracle BI Scheduler.

Configuring Oracle BI Scheduler on UNIX in a Minimum Security Scenarios

1. Execute schconfig located in OracleBI_HOME/OracleBI/server/bin.
2. Choose the following option:

   ```
   1 – Configure Scheduler
   ```

3. Choose the following option:

   ```
   3 – Advanced
   ```

4. Choose option 5 and set Use SSL to "y".

Configuring Oracle BI Scheduler on UNIX in a near-Maximum Security Scenario

This topic describes the process to configure Oracle BI Scheduler installed on UNIX in a maximum security deployment for communication over SSL.

Use this procedure to configure Oracle BI Scheduler.

To configure Oracle BI Scheduler on UNIX in near-maximum security scenario

1. Execute schconfig located in OracleBI_HOME/setup.
2. Choose the following option:

   ```
       1 – Configure Scheduler
   ```

3. Choose the following option:

   ```
       3 – Advanced
   ```

4 Options 5 to 13 are SSL-related. Set them as shown in the following table.

Scheduler Advanced Configuration Option	Value
5 – Use SSL	True
6 – SSL Certificate File Path	<Directory and file name of Server Certificate file> For example, OracleBI_HOME/ssl/server-cert.pem
7 - SSL Certificate Private Key File	<Directory and file name of Server Private Key file> For example, OracleBI_HOME/ssl/server-key.pem
8 – SSL File Containing Passphrase	<Directory and filename of passphrase file for Server key> For example, OracleBI_HOME/ssl/serverpwd.txt
9 – SSL Require Client Certificate	True
10 – SSL Certificate Verification Depth	<Depth of chain>
11- CA Certificate Directory	<Directory containing the hashed CA Certificate> For example, OracleBI_HOME/ssl
12 – SSL Trusted Peers DNs	<Distinguished Names of trusted peers> For example, C=US/ST=CA/L=Redwood Shores/O=Oracle/OU=BI/CN=servercertificate
13 – SSL Cipher List	<Cipher List, if any> For example, EXP-DES-56-SHA

5 Copy the server certificate, server private key and passphrase file to the directory specified in the parameters.

In the examples specified, the directory is OracleBI_HOME/ssl.

6 If you have set the CA Certificate File parameter, copy the CA certificate file to the directory specified.

7 If you have set the CA Certificate Directory parameter, copy the hash version of the CA certificate to the directory specified.

Securing Communication Between Oracle BI Scheduler and SMTP Server

The communication between BI Scheduler and the SMTP server can be secured. The server certificate from the SMTP server must be obtained. This file can either be copied to a directory on the BI Scheduler machine, or the hash version of this file, named <hashvalue>.0 copied to a directory of trusted CAs on the BI Scheduler machine.

Use this procedure to enable the communication between BI Scheduler and the SMTP server.

To secure communication between Oracle BI Scheduler and SMTP Server

1. Launch Job Manager and connect to the Scheduler instance. Navigate to Mail > Advanced tab.
2. Check the "Use Secure Socket Layer" check box.
3. Select either the CA Certificate Directory radio button and specify the path and file name of the SMTP server certificate or the CA Certificate File radio button and specify the directory containing the hash version of the SMTP certificate.
4. Set the SSL Certificate Verification Depth.
5. Specify an SSL Cipher List, if required.

Using SASchInvoke and SchShutdown When BI Scheduler is SSL-Enabled

To use SASchInvoke command line utility when BI Scheduler is enabled for communication to occur over SSL, you must specify SSL-related options as shown below:

```
SASchInvoke -u <Admin Name>/<Admin Password>  (-j <job id> | -i <iBot path>)  [-m
<machine name>[:<port>]]  [(-r <replace parameter filename> | -a <append parameter
filename>)] [-l [ -c SSL certificate filename> -k <SSL certificate private key
filename> [ -w <SSL passphrase>  | -q <passphrase file>  | -y ]] [-h <SSL cipher
list>] [-v [-e <SSL verification depth>] [-d <CA certificate directory>] [-f <CA
certificate file>] [-t <SSL trusted peer DNs>] ] ]
```

To use the SchShutdown command line option when BI Scheduler is enabled for communication to occur over SSl, you must specify the SSL-related options as shown below:

```
SchShutdown -s <machine:port> -u <username> -p <password> [ -l [-c <ssl certificate
file path>-k <ssl private key file path> [-q <ssl private key passphrase file path>
| -w <ssl private key passphrase> | -y ] [-h <ssl cipher list> ]-v [ -e <ssl
verification depth> ] -d <CA Certificate Directory path> | [-f <CA Certificate File
path>][-t <SSL Trusted Peer DNs ]]
```

Configuring Oracle BI Job Manager

To successfully connect to BI Scheduler that has been enabled for SSL, BI Job Manager must also be configured to communicate over SSL.

BI Job Manager is a Java based component and the keys and certificates that it will use must be stored in a java keystore database. The jobmanager.keystore created in topic "Creating the Java Keystore" on page 91 will be used.

Use this procedure to configure BI Job Manager to communicate with the BI Scheduler server over SSL.

To configure Oracle BI Job Manager

1. Open Job Manager and go to File > Open Scheduler Connection.
2. In the Secure Socket Layer section of the dialog box, check the SSL check box. If you are deploying in a minimum security scenario, you do not need to provide any additional values in this dialog box. Click OK to exit.
3. If BI Scheduler has been set to "Require Client Certificate", then Key Store and Key Store Password must be set:

   ```
   Key Store = <Path and file name of the keystore containing Client Certificate and
   Private Key files>. For example, jobmanager.keystore.
   ```

   ```
   Key Store Password = <password of keystore>. For example, analytics.
   ```

4. Check the Verify Server Certificate check box. When this is checked, the trust store file must be specified. This trust store contains the CA that will be used to verify the Scheduler server certificate.
5. In the Trust Store text box, enter the path and file name of the keystore that contains the Certificate Authority file. In the example provided in this chapter, the CA certificate was stored in the same keystore that contains the certificate and private key, jobmanager.keystore.
6. In the Trust Store Password text box, enter the password of the keystore entered in step 5. For example, analytics.
7. Copy the keystore and trust store files to the locations specified in the parameters above.

Configuring Oracle BI Presentation Services for Communication Over SSL

The process of configuring Oracle BI Presentation Services to communicate over SSL consists of modifying parameters in the instanceconfig.xml configuration file. BI Presentation Services accesses certificates and key files from its credential store. The paths to certificates and keys that BI Presentation Services uses must be stored in its credential store.

Specifying Certificate and Key Paths in BI Presentation Services Credential Store

Add locations of all certificates and keys that BI Presentation Services will access into its credential store. The procedure described below adds certificates and keys to the default credential store XML file called credentialstore.xml for BI Presentation Services. You may choose to define the BI Presentation Services Credential Store as a Java keystore or a custom store. For more information on the BI Presentation Services Credential Store and the supported storage systems, refer to Chapter 5, “Oracle BI Presentation Services Credential Store.”

Use this procedure to specify server certificate, private key and CA certificate paths in the credential store called credentialstore.xml. The default location of the credentialstore.xml file is

- For Windows, OracleBIData_HOME\web\config
- For Linux or UNIX, OracleBIData_HOME/web/config

To specify certificate and key paths in BI Presentation Services Credential Store

1 Open the credentialstore.xml file for editing.

2 Add lines similar to the following to specify the paths to the server certificate and private key files:

```
<sawcs:credential type="x509" alias="sawclient">
   <sawcs:key
      encoding="pem"
         passphraseFile="OracleBI_HOME\ssl\serverpwd.txt"
         path="OracleBI_HOME\ssl\server-key.pem"/>
    <sawcs:certificate encoding="pem" path="OracleBI_HOME\ssl\server-cert.pem"/>
</sawcs:credential>
```

NOTE: In the above example, the certificate and key paths are stored under the alias "sawclient". You may specify any alias value.

3 Specify the CA certificate file or the directory of trusted CAs.

4 If using the CA certificate file, add lines similar to the following example:

```
<sawcs:trustedCertificate alias="cacert" encoding="pem"
path="OracleBI_HOME\ssl\cacert.pem"/>
```

5 Where the path to the trusted CA certificate file is stored under the alias "cacert"

6 If using the hash version of the CA certificate, specify the path to the trusted CA directory by adding lines similar to:

```
<sawcs:trustedCertificateDir path="OracleBI_HOME\ssl\CA"/>
```

where the trusted CAs are in a directory called CA under OracleBI_HOME\ssl.

7. Copy the server certificate, private key, passphrase file and CA certificate or hash version of the file to the locations that you have specified in the xml file.

Configuring BI Presentation Services for SSL Communication

The instanceconfig.xml file is located in the OracleBIData_HOME\web\config directory. On Linux or UNIX, it is located in the OracleBIData_HOME/web/config directory.

To configure Oracle BI Presentation Services for communication over SSL

1. Open the instanceconfig.xml file for editing.

2. Modify the existing <ScheduleServer> node:

```
<Alerts>
<ScheduleServer ssl="true" credentialAlias="sawclient"
certificateVerificationDepth="1" verifyPeers="true"><BI Scheduler Host></
ScheduleServer>
</Alerts>
```

3. Add the following elements between the <ServerInstance></ServerInstance> node

```
<Listener ssl="true" credentialAlias="sawclient"
certificateVerificationDepth="1" verifyPeers="true">
</Listener>
<JavaHostProxy>
   <Hosts>
      <Host address="<BI Javahost Host>" port="9810" ssl="true"
credentialAlias="sawclient" certificateVerificationDepth="1" verifyPeers="true"/
>
   </Hosts>
</JavaHostProxy>
```

4. Specify the credential store that stores the paths to the server certificate, private key, and CA.

```
<CredentialStore>
   <CredentialStorage type="file"
path="<OracleBIData_HOME\web\config\credentialstore.xml"/>
</CredentialStore>
```

In the preceding example configuration, BI Presentation Services is directed to obtain the certificate and key using the alias "sawclient". You must specify the alias under which the certificates and keys were stored in the credential store. In the example, the keystore that contains the certificate, private key, and CA is the XML file store called credentialstore.xml.

Online Catalog Manager

The online Catalog Manager may fail to connect to BI Presentation Services when the HTTP web server for Oracle BI is enabled for SSL. You must import the SSL server certificate or CA certificate from the web server into the Java Keystore of the JVM that is specified by the system JAVA_HOME variable.

To import the exported web server certificate to Java's default truststore:

1. Navigate to Java's default trust store located at JAVA_HOME/ jre/lib/security. The default trust store is called cacerts.
2. Copy the certificate exported from the web server to the same location as Java's default truststore.
3. Execute the command to import the certificate to the default truststore:

   ```
   keytool -import -trustcacerts -alias bicert -file $WebServerCertFilename -
   keystore cacerts -storetype JKS
   ```

   NOTE: The default password for the Java trust store is "changeit".

   where the web server certificate file $WebserverCertFilename is imported into Java's default trust store named cacerts under an alias of bicert.

4. Restart the Java process.

Configuring Oracle BI Presentation Services Plug-In for Communication over SSL

The BI Presentation Services Plug-in is of two types.

- If your Oracle Business Intelligence deployment uses a J2EE Application Server, for example Oracle Application Server, to service web requests, the BI Presentation ServicesWeb Plug-in is a Java Servlet deployed on the J2EE container. For information about configuring the Java Servlet for communication over SSL, see "Configuring BI Presentation Services Plug-in (ISAPI) for Communication Over SSL" on page 117.
- If your Oracle Business Intelligence deployment uses Internet Information Services (IIS) to service web requests, the BI Presentation Services Plug-in is an ISAPI plug-in. For information about configuring the ISAPI Plug-in to communicate over SS, see "Configuring BI Presentation Services Plug-in (ISAPI) for Communication Over SSL" on page 117.

Configuring BI Presentation Services (Java Servlet) for Communication over SSL

The process of configuring the BI Presentation Services Plug-in (Java Servlet) deployed on a J2EE container consists of adding SSL-related entries in the web.xml file. The default version of this file is located in the directory OracleBI_HOME/web/app/WEB-INF.

The BI Presentation Services Plug-In (Java Servlet) uses a Java keystore to store certificates and keys. Use the keystore that was created in topic "Creating the Java Keystore" on page 91. Copy this keystore (named jobmanager.keystore in the example) to all machines where the BI Presentation Services Plug-in is deployed.

To configure the BI Presentation Services (Java Servlet) plug-in for SSL communication

1. Open the web.xml for the analytics application deployed on your J2EE server. The file is located in the WEB-INF directory for the analytics Web application.
2. Insert the following elements and values inside the <servlet> tag:

```
<init-param>

<param-name>oracle.bi.Secure</param-name>

<param-value>Y</param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.CertAlias</param-name>

<param-value><Alias of stored Certificate and key></param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.CertStoreFile</param-name>

<param-value><Path and file name of keystore containing certificates></param-
value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.CertStorePwd</param-name>

<param-value><password for keystore></param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.TrustStoreFile</param-name>
```

```
<param-value><Path and filename of keystore containing Certificate Authority certificates></param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.TrustStorePwd</param-name>

<param-value><password for keystore></param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.Protocol</param-name>

<param-value>TLS</param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.TrustAnyPeer</param-name>

<param-value>Y</param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.TrustedPeerDNs</param-name>

<param-value> </param-value>

</init-param>
```

After modification, the web.xml file should appear similar to the following example:

```
<init-param>

<param-name>oracle.bi.Secure</param-name>

<param-value>Y</param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.CertAlias</param-name>

<param-value>jobmanagertkey</param-value>

</init-param>

<init-param>

<param-name>oracle.bi.ssl.CertStoreFile</param-name>
```

```
<param-value>OracleBI_HOME>/ssl/jobmanager.keystore</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.ssl.CertStorePwd</param-name>
<param-value>analytics</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.ssl.TrustStoreFile</param-name>
<param-value>OracleBI_HOME>/ssl/jobmanager.keystore</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.ssl.TrustStorePwd</param-name>
<param-value>analytics</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.ssl.Protocol</param-name>
<param-value>TLS</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.ssl.TrustAnyPeer</param-name>
<param-value>Y</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.ssl.TrustedPeerDNs</param-name>
<param-value> </param-value>
</init-param>
```

3 Place a copy of the modified web.xml file in OracleBI_HOME/web/app/WEB-INF.

4 Restart the J2EE Containers.

Configuring BI Presentation Services Plug-in (ISAPI) for Communication Over SSL

The process of configuring the BI Presentation Services ISAPI to communicate over SSL consists of adding SSL-related elements to the isapiconfig.xml file. This file is located in the OracleBIData_HOME\web\config directory.

The BI Presentation Services Plug-in accesses certificates and keys from the credential store defined for BI Presentation Services. Copy the BI Presentation Services credential store containing the certificate and private key that was created in topic "Specifying Certificate and Key Paths in BI Presentation Services Credential Store" on page 111, to all machines that host the BI Presentation Services Plug-in, for example, to OracleBIData_HOME\web\config.

NOTE: Perform this configuration only if your web server is IIS.

To configure the BI Presentation Services Plug-in (ISAPI) for communication over SSL

1. Open the isapiconfig.xml file for editing. This file is located in OracleBIData_HOME\web\config directory.
2. Add the following SSL-related elements as shown in the following example:

   ```
   <ServerConnectInfo address="<BI Presentation Services Host" port="9710" ssl
   ="true" credentialAlias="sawclient" certificateVerificationDepth="1"
   sslVersion="SSLv23"/>
   <CredentialStore>

   <CredentialStorage type="file"
   path=OracleBIData_HOME\web\config\credentialstore.xml"/>
   </CredentialStore>
   ```

   In the preceding example, the credential store is the default xml file store called credentialstore.xml. This store contains the paths to the certificate and private key file stored on disk.
3. Copy the certificate, private key file and CA certificate or hash version of the file to the locations for these files as specified in the credential store.
4. Restart the World Wide Web Publishing Service.

Configuring Oracle BI Javahost for Communication Over SSL

The BI Javahost component is Java based and uses the Java Keystore to store certificates and keys that it uses. The keystore that was created in topic "Creating the Java Keystore" on page 91 is used.

BI Javahost is configured by setting the config.xml file. The SSL-related settings are under the Listener node. The Secure element when set to true enables SSL. The SSL sub-element under the Listener nodes specifies additional SSL settings.

To configure Oracle BI Javahost for communication over SSL

1 Open the config.xml file for editing. This file is located in the OracleBI_HOME/web/config directory. On UNIX, this file is located in the OracleBI_HOME/web/config directory.

2 Add the following SSL-related elements and values under the Listener node as shown in the following example:

```
<Listener>

.
<Secure>Yes</Secure>
   <SSL>
      <CertAlias><Alias for certificate and key></CertAlias>
      <CertStoreFile><Path and filename for keystore containing certificate and
key></CertStoreFile>
      <CertStorePwd><Keystore password></CertStorePwd>
      <KeyPwd><Password for CertAlias; same as CertStorePwd></KeyPwd>
      <CertStoreType>JKS</CertStoreType>
       <TrustStoreFile><Path and filename for trust store containing CAs></
TrustStoreFile>
      <TrustStorePwd><Password for Trust Store></TrustStorePwd>
      <TrustStoreType>JKS</TrustStoreType>
      <TrustAnyPeer>Y</TrustAnyPeer>

   <!--  <EnabledCipherSuites/> -->
   </SSL>
.
.
</Listener>
```

NOTE: The config.xml file has the above-mentioned elements commented out. You may choose to uncomment the elements and add the corresponding values. Or, you may leave the elements commented out and create new ones as described above.

After modification, the config.xml file should be similar to the following example:

```
<Listener>
  .
   <Secure>Yes</Secure>
   <SSL>
      <CertAlias>jobmanagerkey</CertAlias>
      <CertStoreFile>D:\OracleBI\ssl\jobmanager.keystore</CertStoreFile>
      <CertStorePwd>analytics</CertStorePwd>
      <KeyPwd>analytics</KeyPwd>
      <CertStoreType>JKS</CertStoreType>
      <TrustStoreFile>D:\OracleBI\ssl\jobmanager.keystore</TrustStoreFile>
      <TrustStorePwd>analytics</TrustStorePwd>  <TrustStoreType>JKS</
TrustStoreType>
      <TrustAnyPeer>Y</TrustAnyPeer>
   </SSL>|
            .
            .
        </Listener>
```

NOTE: Copy the Java keystore and trust store to the locations specified in the configuration file. In the example, the jobmanger.keystore also contains the CAs.

Configuring BI Presentation Services and BI Publisher When SSL is Enabled

If you are using the Oracle BI Reporting and Publishing feature and have configured BI Publisher to integrate with Oracle BI, then you must perform the following additional steps to ensure that communication between BI Presentation Services and BI Publisher occurs successfully when the SSL Everywhere feature has been enabled for Oracle BI and the BI Publisher application is accessed via the HTTPS protocol.

NOTE: It is assumed that you have installed and configured BI Publisher to integrate with Oracle BI. For details of this install and configuration, see the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

- "Exporting the Apache Certificate from the Wallet" on page 119
- "Modifying the AdvancedReporting tag in instanceconfig.xml" on page 120
- "Modifying BI Publisher Settings" on page 121

When the HTTP web server for the BI Publisher application is enabled for HTTPS, the certificate or CA of the web server must be exported from the web server and imported to the default Java trust store that will be accessed by BI Publisher J2EE container. The certificate must also be imported to the BI Presentation Services credential store.

NOTE: To enable SSL for the web server for the BI Publisher application, refer to the web server vendor documentation. For more information about enabling SSL for Oracle Application Server and Oracle HTTP Server, refer to the *Oracle Application Server Administrator's Guide 10g Release 3 (10.1.3.1.0)* and the *Oracle HTTP Server Administrator's Guide 10g (10.1.3.1.0)*Oracle HTTP Server Administrator's Guide 10g (10.1.3.1.0).

Exporting the Apache Certificate from the Wallet

The following procedures describes how to export the Apache certificate from the wallet on Oracle HTTP Server.

NOTE: To export a certificate or CA of the HTTP server for BI Publisher, see your web server vendor documentation.

To export the certificate from the wallet

1. On the machine where Oracle HTTP Server is installed, navigate to ORACLE_HOME/bin.
2. Export the Apache certificate from the wallet by executing the command:

```
orapki wallet export -wallet wallet_location -dn
certificate_dn -cert certificate_filename
```

This command exports a certificate with the subject's distinguished name (-dn) from a wallet to a file that is specified by -cert, where:

- The "-dn" should be the certificate Distinguished Name for the OHS web server.
- The wallet is located at ORACLE_HOME/Apache/Apache/conf/ssl.wlt/default/ewallet.p12.

NOTE: For more information, see the chapter about managing wallets and certificates in the *Oracle Business Intelligence Server Administration Guide*.

To import the exported certificate to Java's default trust store named cacerts

1. Navigate to Java's default trust store located at JAVA_HOME/ jre/lib/security/cacerts.
2. Copy the web server certificate to the same location as the Java's default truststore.
3. Execute the command to import the certificate to the default trust store:

   ```
   keytool -import -trustcacerts -alias bicert -file $WebServerCertFilename
   -keystore cacerts -storetype JKS
   ```

   where $WebServerCertFilename is the name of the certificate exported from the web server. The certificate is stored under the alias "bicert" in the cacerts trust store.

   NOTE: The default password for the Java trust store is "changeit".

4. Restart the Java process and Application Server.
5. Import the exported web server certificate to the BI Presentation Services Credential Store. The credential store of each instance of BI Presentation Services in your deployment must contain this certificate.

Modifying the AdvancedReporting tag in instanceconfig.xml

When the BI Publisher application is accessed using the HTTPS protocol, you must modify the AdvancedReporting tag in the instanceconfig.xml file for BI Presentation Services to identify the BI Publisher URL.

To modify the AdvancedReporting tag in instanceconfig.xml

1. On the BI Presentation Services machine, open the instanceconfig.xml file for editing. This file is located in OracleBIData_HOME\web\config on Windows and in OracleBIData_HOME/web/config on Linux or UNIX.
2. Locate and set the <AdvancedReporting> element to identify the BI Publisher URL as follows:

   ```
   <AdvancedReporting>.
   .
   <ServerURL>https://bi-publisher.mycompany.com:443/xmlpserver/services/
   XMLPService</ServerURL>
   <WebURL>https://bi-publisher.mycompany.com:443/xmlpserver</WebURL>
   <AdminURL>https://bi-publisher.mycompany.com:443/xmlpserver/servlet/admin</
   ```

```
AdminURL>
.
.
</AdvancedReporting>
```

3 Perform the above modifications on each instance of BI Presentation Services in your deployment.

Modifying BI Publisher Settings

When Oracle BI components are enabled for communication to occur over SSL, the settings in BI Publisher for integration with Oracle BI must be modified.

To modify the BI Publisher settings

1 Access the BI Publisher application using its URL, for example:

```
https://bi-publisher.mycompany.com/xmlpserver
```

2 Log in to the BI Publisher application.

3 Navigate to the Admin tab, Under Data Sources, click the JDBC Connection link.

4 On the JDBC tab of the Data Sources page, select the Oracle BI EE data source.

5 Modify the Connect String field as follows:

Append the following string to the Connect String:

```
ssl=true;sslKeystorefilename=<path and filename of
keystore>;sslKeystorepassword=<password of keystore and
key>;trustanyserver=true;
```

where the SSLKeystorefilename identifies the Java keystore that contains the certificate exported from the web server, and sslKeystorepassword is the password for the keystore.

The connection string should be similar to the following example:

```
jdbc:oraclebi://<BI Server>:9703;ssl=true;sslKeystorefilename=<path and filename
of keystore>;sslKeystorepassword=<password of keystore and
key>;trustanyserver=true;
```

6 If you have Oracle BI clustering enabled, the connection string should be similar to the following:

```
jdbc:oraclebi://<Primary Cluster Controller>:9706/PrimaryCCS=<Primary Cluster
Controller>;PrimaryCCSPort=9706;SecondaryCCS=<Secondary Cluster
Controller>;SecondaryCCSPort=9706;ssl=true;sslKeystorefilename=<path and
filename of keystore>;sslKeystorepassword=<password of keystore and
key>;trustanyserver=true;
```

7 Test the connection by clicking the Test Connection button.

8 Set the Username and Password fields to the Oracle BI Administrator credentials.

9 Verify that the Database Driver Class is set to the following:

```
oracle.bi.jdbc.AnaJdbcDriver
```

Click Apply.

10 Navigate to Admin tab > Integration - Oracle BI Presentation Services, and set the following fields:

- Server Protocol = https
- Server Version = v4
- Server = bi.mycompany.com

  Where bi.mycompany.com is the Web Server or Application Server host where Oracle BI is deployed.
- Port = 443
- Administrator Username = Administrator
- Administrator Password = <Password for Oracle BI Administrator user>
- URL Suffix = analytics/saw.dll

NOTE: You must specify the Administrator user in the Administrator Username field. This is the Administrator user defined in the Oracle BI repository (rpd).

11 If you have defined BI Server Security as the security model in BI Publisher, you must modify the JDBC connection string as done in this procedure.

In the BI Publisher application > Admin tab > Security Configuration page, append following SSL string the Connection String field:

```
ssl=true;sslKeystorefilename=<path and filename of
keystore>;sslKeystorepassword=<password of keystore and
key>;trustanyserver=true;
```

where the SSLKeystorefilename identifies the Java keystore that contains the certificate exported from the web server, and sslKeystorepassword is the password for the keystore.

7 Oracle Business Intelligence Authentication Mechanisms

This chapter summarizes the authentication methods supported by Oracle Business Intelligence. The Oracle Business Intelligence server supports the methods of authentication shown in Table 11 on page 123.

Table 11. Oracle BI Server Authentication Methods

Method	Description
Database authentication	The Oracle Business Intelligence repository is preconfigured for database authentication. This may be changed using the Server Administration Tool. See the *Oracle Business Intelligence Server Administration Tool Online Help*.
LDAP (Lightweight Directory Access Protocol) server authentication	Oracle Business Intelligence Server supports LDAP in both Secure Socket Layer (SSL) and regular (non-SSL) modes. An LDAP server treats Oracle Business Intelligence Server as a regular LDAP client. Oracle Business Intelligence Server supports authentication against multiple LDAP servers.
ADSI (Active Directory Service Interfaces) authentication	Oracle Business Intelligence Server supports ADSI in both Secure Socket Layer (SSL) and regular (non-SSL) modes. An Active Directory Server treats Oracle Business Intelligence Server as a regular LDAP client. Oracle Business Intelligence Server supports authentication against multiple Active Directory servers. **NOTE:** Oracle Business Intelligence Server is still a LDAP client when it runs against ADSI.

Authentication on LDAP and ADSI servers uses Oracle Business Intelligence Server session variables. Some session variables, such as PASSWORD, are populated automatically. They receive their values when a user begins a session by logging on. Instead of storing user names and passwords in an Oracle Business Intelligence Server repository, the Oracle Business Intelligence Server passes the user's user name and password to an LDAP server for authentication.

Some session variables, such as GROUP, need to be manually created in the Oracle BI repository. Initialization blocks specify the attributes to be retrieved in session variables. Certain session variables, called *system* session variables, have special uses. For more information about session variables, the USER system variable, and the Variable Manager, see the appropriate topics in *Oracle Business Intelligence Server Administration Guide* or *Oracle Business Intelligence Server Administration Tool Online Help*.

The following key restrictions apply to LDAP and ADSI authentication:

- Importing of user information into the repository is supported on regular LDAP servers, but not supported on ADSI servers.
- Groups are defined in the repository. However, if lists of users are stored on LDAP servers, the group membership information must be obtained from a database table.
- When a User exists in both the repository and in an external source (such as LDAP servers), the local repository User definition takes precedence. This restriction allows the Oracle Business Intelligence Server Administrator to override users that exist in an external security system.

8 Implementing Single Sign-On Products With Oracle Business Intelligence

Oracle Business Intelligence provides an open interface to enable web integration with Single Sign-On (SSO) products. Any SSO product that complies with industry standard techniques for passing authentication credentials can achieve SSO integration with Oracle BI.

This chapter describes how the integration of Oracle Business Intelligence with Single Sign-On (SSO) products may be achieved. Details of the configuration that is needed for Oracle BI in order to enable single sign-on are provided, along with an explanation of how Oracle BI operates when SSO has been enabled. To help you implement Oracle BI with the SSO system of choice, sample configuration files are provided that cover different scenarios.

NOTE: For details on integrating Oracle SSO with Oracle BI, see chapter Enabling Oracle Single Sign-On for Oracle Business Intelligence in this guide.

Prerequisites for SSO Systems to Integrate With Oracle Business Intelligence

To enable SSO with Oracle Business Intelligence, the SSO system of choice must be able to provide Oracle BI (specifically, the Oracle BI Presentation Services component) with the username of the authenticated user that is issuing a request to Oracle BI. Oracle BI Presentation Services must receive the username of the end user by one of the following mechanisms:

- Through an HTTP header or HTTP cookie containing the username of the end user. The header can be any valid HTTP header or cookie name.
- Or, by using one of the following server-side options:
  - When using a J2EE Application Server and the BI Presentation Services Plug-In (Java Servlet), from the getRemoteUser method of the javax.servlet.http.HttpServletRequest.getRemoteUser API.

    In this case, the SSO system must be able to integrate with the J2EE environment of choice and set up the framework such that the getRemoteUser method returns the username of the end user.
  - When using Internet Information Server (IIS) and the BI Presentation Services Plug-In (ISAPI Plug-in), from the REMOTE_USER server variable that is populated with the username of the end user.

    REMOTE_USER is a server variable queried through the use of the ISAPI Extension API GetServerVariable.

Understanding How Oracle BI Presentation Services Operates in an SSO Environment

In an environment where SSO has been implemented, when Oracle BI Presentation Services receives an incoming web request, it assumes that the user who issued the request has already been authenticated by the SSO system. Oracle BI Presentation Services uses its own credentials to establish a connection with the Oracle BI Server on behalf of the end user. User personalization and access controls such as data-level security are maintained in this environment. Oracle BI Presentation Services then uses the Oracle BI Server Impersonation feature to create a connection to the Oracle BI Server on behalf of the authenticated end user.

To establish the connection toOracle BI Server, Oracle BI Presentation Services issues a connection string. This connection string has one required parameter called Impersonate. In addition, any parameters supplied by the SSO system, for example locale, default dashboard or other personalization parameters, can be passed to the Oracle BI Server through the connection string.

Oracle BI Presentation Services must be instructed on how to build the connection string. This is done by setting param name attributes for each of the parameters that need to be passed in the connection string in the instanceconfig.xml configuration file.

For every parameter that is passed through in the connection string, Oracle BI Presentation Services has to be instructed on where to query the value of the parameter from. The possible sources, as described in topic Prerequisites for SSO Systems to Integrate With Oracle Business Intelligence, are:

- HTTP header
- HTTP cookie
- Server variable

For example, the following section of the instanceconfig.xml file has been configured to flag that SSO is enabled to direct Oracle BI Presentation Services to include the Impersonate parameter in the connection string and obtain its value from server variable using the server-side option:

```
 <!-- other settings ... -->
   <Auth>
      <SSO enabled="true">
         <ParamList>
            <!--IMPERSONATE param is used to get the authenticated user's username
and is required -->
            <Param name="IMPERSONATE"
            source="serverVariable"
            nameInSource="REMOTE_USER"/>
         </ParamList>
      </SSO>
<!-- other settings ... -->
```

NOTE: Any URL parameters, for example, nQUser, nQPassword, Impersonate, take precedence over SSO authentication. If Oracle BI Presentation Services is passed authentication information in the URL, it will ignore any values read from configured SSO sources. For example, Symbolic URLs for integration with Oracle's Siebel CRM are configured to support nQUser/nQPassword authentication. To enable SSO authentication, you must remove nQUser and nQPassword from the URLs.

Enabling SSO Authentication for Oracle Business Intelligence

The process of enabling SSO for Oracle Business Intelligence consists of configuring the Oracle BI Presentation Services component to operate in an SSO environment.

Oracle BI Presentation Services must first be configured to use the impersonator user so that it can establish a connection to the Oracle BI Server on behalf of the authenticated end user that issued a request to Oracle BI.

Configuring Oracle BI Presentation Services to Use Impersonator User

This configuration consists of the following tasks:

- "Creating the Oracle BI Server Impersonator User" on page 127
- "Adding Impersonator User Credentials to Oracle BI Presentation Services Credential Store" on page 128
- "Configuring Oracle BI Presentation Services to Identify the Credential Store and Decryption Passphrase" on page 130

Creating the Oracle BI Server Impersonator User

Oracle BI Presentation Services uses the Oracle BI Impersonation feature to establish a connection to the Oracle BI Server on behalf of the authenticated end user. For this purpose, a special user that Oracle BI Presentation Services will utilize for impersonating the authenticated end user needs to be created. This document refers to this special user as the impersonator user.

NOTE: The Oracle BI Server supports a notion of privileged users being able to impersonate other users. This functionality is used by OBI PS to implement SSO support in various scenarios.

The impersonator user is created in the Oracle BI Server repository. If an impersonator user has already been created, you do not need to create a new one. Use this procedure to create the impersonator user in the Oracle BI Server repository. For more information on creating users and granting Group membership, refer to the Oracle Business Intelligence Server Administration Guide.

To create the Oracle BI Server impersonator user

1. Open the Oracle BI Server repository file (.rpd) using Oracle BI Administration Tool.
2. Select Manage > Security to display the Security Manager.
3. Select Action > New > User to open the User dialog box.
4. Enter a name and password for this user.

   For example, Name = Impersonator and Password = secret.
5. Click OK to create the user.

6 To make this user a member of the group Administrators, double-click on the icon for the user that was created. In the

7 In the Group Membership portion of the dialog box, check the Administrators group to grant the user created above membership to this group.

Adding Impersonator User Credentials to Oracle BI Presentation Services Credential Store

For Oracle BI Presentation Services to be able to utilize the user created above for impersonation of the authenticated end user, it must be able to identify the impersonator user and obtain the impersonator user credentials. The impersonator user credentials must be added to the Oracle BI Presentation Services Credential Store. To obtain the impersonator user credentials, Oracle BI Presentation Services will search the credential store for a username-password credential with an alias of impersonation.

For more information about the BI Presentation Services Credential Store and the supported storage systems, see Chapter 5, “Oracle BI Presentation Services Credential Store”.

Use the procedure below to add the impersonator user credentials to the credential store called credentialstore.xml with an alias of impersonation. The default location of the credentialstore.xml file is OracleBIData_HOME\web\config on Windows and OracleBIData_HOME/web/config on Linux or UNIX.

To add impersonator user credentials to Oracle BI Presentation Services Credential Store

The procedure below assumes that the credentials store is the BI Presentation Services proprietary XML file store. You may choose to store credentials in a Java keystore or a custom store.

1 Open a command prompt window or command shell on the machine where Oracle BI Presentation Services has been installed.

2 Navigate to the directory OracleBI_HOME\web\bin on Windows or OracleBI_HOME/web/bin on Linux or UNIX. This is the location for the CryptoTools utility.

3 Execute the CryptoTools utility to add the impersonator user credentials to the Oracle BI Presentation Services Credential Store:

```
cryptotools credstore -add -infile <OracleBIData>/web/config/credentialstore.xml
```

For more information on the CryptoTool utility, its syntax and supported sub-commands, refer to Appendix B, “Using the CryptoTools Utility”.

4 Supply values for the prompted parameters, as shown in the following table.

Parameter or Prompt	Value or Input	Description
Credential Alias	impersonation	Specify the value impersonation to identify the user as the impersonator user.
Username	<name of the user>	Name of the user created in the topic "Creating the Oracle BI Server Impersonator User" on page 127. For example, Impersonator.
Password	<password of the user>	Password of the user created in the topic "Creating the Oracle BI Server Impersonator User" on page 127. For example, secret.
Do you want to encrypt the password?	y	
Passphrase for encryption	<passphrase>	Provide a passphrase. For example, another_secret.
Do you want to write the passphrase to the xml?	n	

For example:

```
cryptotools credstore -add -infile <OracleBIData>/web/config/credentialstore.xml
>Credential Alias: impersonation
>Username: Impersonator
>Password: secret
>Do you want to encrypt the password? y/n (y):
>Passphrase for encryption: another_secret
>Do you want to write the passphrase to the xml? y/n (n):
>File "<OracleBIData>/web/config/credentialstore.xml" exists. Do you want to
overwrite it? y/n (y):
```

The CryptoTools utility updates the credentialstore.xml file. After executing the CryptoTools utility with inputs as specified above, the credentialstore.xml file contains entries similar to the following example:

```
<sawcs:credential type="usernamePassword" alias="impersonation">
<sawcs:username>Impersonator</sawcs:username>
<sawcs:password>
   <xenc:EncryptedData>
   <xenc:EncryptionMethod Algorithm="http://www.rsasecurity.com/rsalabs/pkcs/
schemas/pkcs-5#pbes2">
      <pkcs-5:PBES2-params Algorithm="http://www.rsasecurity.com/rsalabs/pkcs/
schemas/pkcs-5#pbkdf2">
         <pkcs-5:KeyDerivationFunc>
         <pkcs-5:Parameters>
         <pkcs-5:IterationCount>1024</pkcs-5:IterationCount>
      </pkcs-5:Parameters>
      </pkcs-5:KeyDerivationFunc>
      <pkcs-5:EncryptionScheme Algorithm="http://www.w3.org/2001/04/
xmlenc#tripledes-cbc"/>
      </pkcs-5:PBES2-params>
   </xenc:EncryptionMethod>
   <xenc:CipherData>
   <xenc:CipherValue>jeThdk8ZklnTlyKIat8Dkw</xenc:CipherValue>
   </xenc:CipherData>
   </xenc:EncryptedData>
</sawcs:password>
</sawcs:credential>
```

NOTE: If you have multiple instances of BI Presentation Services in your deployment, you must add the Impersonator credentials to the credential store for every BI Presentation Services instance. Or, you may copy the credential store with updated credentials to each BI Presentation Services machine. The instanceconfig.xml file for each BI Presentation Services must specify the location of the credential store.

Configuring Oracle BI Presentation Services to Identify the Credential Store and Decryption Passphrase

Oracle BI Presentation Services must be directed to the credential store that contains the impersonator user credentials. This is done by setting parameters in the Oracle BI Presentation Servicesconfiguration file, instanceconfig.xml. In addition, the passphrase that Oracle BI Presentation Services will use to decrypt the impersonator password credential must be specified.

To configure Oracle BI Presentation Services to identify the Credential Store and decryption passphrase

1. Open the instanceconfig.xml file for editing.

2. Locate the <CredentialStore> node within this file.

   Specify attribute values as shown below. If the <CredentialStore> node does not exist, create this element with sub-elements and attributes with attribute values as shown in the following example.

   ```
   <WebConfig>
      <ServerInstance>
         <!-- other settings ... -->
         <CredentialStore>
            <CredentialStorage type="file" path="<path to credentialstore.xml>"
   passphrase="<passphrase>"/>
            <!-- other settings ... -->
         </CredentialStore>
         <!-- other settings ... -->
      </ServerInstance>
   </WebConfig>
   ```

Table 12 on page 131 summarizes the attributes and attribute values for the CredentialStorage element. For more information on the CredentialStore and CredentialStorage elements of the instanceconfig.xml file, and for their settings when credential stores other than the XML file store are used, see Chapter 5, "Oracle BI Presentation Services Credential Store."

Table 12. CredentialStorage Element Attributes

Attribute	Attribute Value	Description
type	file	This describes the type of credential store. Set to file for the proprietary XML file credential store.
path	<path to XML file credential store (credentialstore.xml)>	Location and filename for the XML file credential store. For example, OracleBIData_HOME/web/config/credentialstore.xml
passphrase	<passphrase>	Determines the passphrase used to decrypt encrypted files. Provide the value entered in step 4 under topic Adding Impersonator User Credentials to Oracle BI Presentation Services Credential Store. In the example provided, this value is another_secret.

After the modification described in the preceding procedure, the instanceconfig.xml contains entries should appear as in the following example:

```
<?xml version="1.0"?>

<WebConfig>
   <ServerInstance>
   <!-- other settings ... -->
      <CredentialStore>
         <CredentialStorage type="file" path="<OracleBIData>/web/config/
credentialstore.xml" passphrase="another_secret"/>
   <!-- other settings ... -->
      </CredentialStore>
   <!-- other settings ... -->
   </ServerInstance>
</WebConfig>
```

NOTE: Both the files, credentialstore.xml and instanceconfig.xml should be protected using OS filesystem protection capabilities as their combination could reveal a privileged user's password. Note that neither file on its own has enough information to expose the password.

Configuring Oracle BI Presentation Services to Operate in an SSO Environment

Perform the following configuration on all instances of Oracle BI Presentation Services in your deployment. Shut down the Oracle BI Presentation Services before making any changes.

The instanceconfig.xml file is located in the OracleBIData>\web\config directory. On Linux or UNIX, the file is located in OracleBI_HOME/OracleBIData/web/config.

To configure Oracle BI Presentation Services to Operate in an SSO environment

- Open instanceconfig.xml for editing. Locate the <Auth> element. If this does not exist, create this element, sub-elements and parameters as shown in the following example:

```
<!-- other settings ... -->
   <Auth>
      <SSO enabled="true">
         <ParamList>
            <!--IMPERSONATE param is used to get the authenticated user's
username and is required -->
            <Param name="IMPERSONATE"
               source="serverVariable"
               nameInSource="REMOTE_USER"/>
         </ParamList>
      </SSO>
```

In the preceding example, Oracle BI Presentation Services is directed to build a connection string containing the Impersonate parameter and to retrieve the value for this parameter using the server-side option. This is connection string will be used to establish a connection with Oracle BI Server on behalf of the authenticated user.

Additional parameters that are supplied by the SSO system (for example, locale) can be included in the connection string by setting param name attributes in the intstanceconfig.xml file for each of the parameters and specifying where Oracle BI Presentation Services must query the value of the parameter from. The possible attribute values for the source attribute are cookie, header and serverVariable.

For detailed examples, review the sample configuration files provided in the topic "Sample Configuration Files" on page 134.

Important Considerations For Implementing SSO for Oracle Business Intelligence

When implementing SSO for Oracle Business Intelligence, consider the following:

- When the authentication source is a Microsoft Windows domain, and there is a need to strip out the domain portion from the username, then a special attribute may be specified to do this task.
- When accepting such trusted information from the HTTP server or servlet container, it is essential to secure the machines that are permitted to communicate with Oracle BI Presentation Services directly. This can be done by setting the Listener\Firewall node in instanceconfig.xml with the list of HTTP Server or servlet container IP addresses. In addition, the Firewall node must include the IP addresses of all BI Scheduler instances, BI Presentation Services Plug-in instances (ISAPI Plug-in or Java Servlet) and BI Javahost instances. If any of these components are co-located with Oracle BI Presentation Services, then address 127.0.0.1 must be added in this list as well. Note that this setting does not control end-user browser IP addresses.
- When using mutually-authenticated SSL, you must specify the Distinguished Names (DNs) of all trusted hosts in the Listener\TrustedPeers node.
- For information, refer to Chapter 6, "Enabling Secure Communication in Oracle Business Intelligence."
- Configure optional Logoff/Logon URLs.

  In environments where Single Sign-On (SSO) is enabled, you can configure log out and log on links to appear on Oracle BI Presentation Services screens. To do so, you add the elements shown in the following table as children of the SSO element in the instanceconfig.xml file.

Element	Description
LogoffUrl	Turns on the log off link on Oracle BI Presentation Services screens and specifies the URL to navigate to when a user clicks the link.
LogonUrl	Turns on the log on link on the screen that appears when a user is not logged in to Oracle BI Presentation Services and specifies the URL to navigate to when a user clicks the link.

For example:

```
<SSO>
    <LogoffUrl>http://hostname:port/the_url_to_logoff_sso</LogoffUrl>
    <LogonUrl>http://hostname:port/the_url_to_logon_sso</LogonUrl>
</SSO>
```

The logoff and logon URLs can also contain expressions. For example, @{user.id} can be inserted to the logoff URL. Oracle BI Presentation Services will replace it with the ID of the user. For more information on web variables and the expressions that may be used, refer to the *Oracle Business Intelligence Server Administration Guide*.

Sample Configuration Files

This section provides samples of the instanceconfig.xml configured in different scenarios. For each scenario, the resulting connection string for a user named "testuser" has been provided.

Using J2EE integration

```
<?xml version="1.0"?>

<WebConfig>
<ServerInstance>
<!-- other settings ... -->
<Listener>
<Firewall>
<Allow address="127.0.0.1"/>
<Allow address="192.168.1.100/>
<Allow address="192.168.1.101/>
</Firewall>
<!-- other settings ... -->
</Listener>
<!-- other settings ... -->
<CredentialStore>
<CredentialStorage type="file" path="<OracleBIData>/web/config/
credentialstore.xml" passphrase="another_secret"/>
<!-- other settings ... -->
</CredentialStore>
<!-- other settings ... -->
<Auth>
<SSO enabled="true">
<ParamList>
<!--IMPERSONATE param is used to get the authenticated user's username and is
required -->
<Param name="IMPERSONATE"
source="serverVariable"
nameInSource="REMOTE_USER"/>
<!--Optional, NQ_SESSION.LOCALE sets up the user's locale as determined by SSO
system -->
<Param name="NQ_SESSION.LOCALE"
source="cookie"
nameInSource="SSO_LOCALE"/>
</ParamList>
<!--Optional. Replace the URLs with actual logoff/logon URL-->
<LogoffUrl>http://hostname:port/sso/logoff</LogoffUrl>
<LogonUrl>http://hostname:port/logon</LogonUrl>
</SSO>
</Auth>
<!-- other settings ... -->
</ServerInstance>
</WebConfig>
```

This results in a connection string that takes on the following form:

```
UID=Impersonator;PWD=secret;IMPERSONATE=testuser;NQ_SESSION.LOCALE=en-US;
```

Using Microsoft IIS to perform user authentication

```
<?xml version="1.0"?>

<WebConfig>
   <ServerInstance>
   <!-- other settings ... -->
      <Listener>
         <Firewall>
            <Allow address="127.0.0.1"/>
            <Allow address="192.168.1.100/>
            <Allow address="192.168.1.101/>
         </Firewall>
   <!-- other settings ... -->
      </Listener>
      <!-- other settings ... -->
      <CredentialStore>
         <CredentialStorage type="file" path="<OracleBIData>/web/config/
credentialstore.xml" passphrase="another_secret"/>
   <!-- other settings ... -->
      </CredentialStore>
   <!-- other settings ... -->
      <Auth>
         <SSO enabled="true">
            <ParamList>
            <!--IMPERSONATE param is used to get the authenticated user's
username and is required -->
            <Param name="IMPERSONATE"
               source="serverVariable"
               nameInSource="REMOTE_USER"
               stripWindowsDomain="true"/>
            </ParamList>
         </SSO>
      </Auth>
   <!-- other settings ... -->
   </ServerInstance>
</WebConfig>
```

This results in a connection string that takes on the following form:

```
UID=Impersonator;PWD=secret;IMPERSONATE=testuser;
```

Using an HTTP header with additional parameters

```
<?xml version="1.0"?>

<WebConfig>
   <ServerInstance>
   <!-- other settings ... -->
```

```
      <Listener>
         <Firewall>
            <Allow address="127.0.0.1"/>
            <Allow address="192.168.1.100/>
            <Allow address="192.168.1.101/>
         </Firewall>
   <!-- other settings ... -->
      </Listener>
   <!-- other settings ... -->
   <CredentialStore>
      <CredentialStorage type="file" path="<OracleBIData>/web/config/
credentialstore.xml" passphrase="another_secret"/>
   <!-- other settings ... -->
   </CredentialStore>
   <!-- other settings ... -->
   <Auth>
      <SSO enabled="true">
         <ParamList>
            <!--IMPERSONATE param is used to get the authenticated user's
username and is required -->
            <Param name="IMPERSONATE"
               source="httpHeader"
               nameInSource="x-Foo-SSO-GUID"/>
            <!--Optional, NQ_SESSION.LOCALE sets up the user's locale as
determined by SSO system -->
            <Param name="NQ_SESSION.LOCALE"
source="cookie"
               nameInSource="FOO_SSO_LocaleID"/>
           <!--Optional, NQ_SESSION.P1 sets up some other parameter from SSO
system -->
            <Param name="NQ_SESSION.P1"
               source="header"
               nameInSource="FOO_SSO_P1"/>
          </ParamList>
      </SSO>
   </Auth>
   <!-- other settings ... -->
   </ServerInstance>
</WebConfig>
```

This results in a connection string that takes on the following form:

```
UID=Impersonator;PWD=secret;IMPERSONATE=testuser;NQ_SESSION.LOCALE=en-
GB;NQ_SESSION.P1=param1;
```

9 Other Deployment-Related Topics

This chapter provides information on the following topics:

- "Administrator Accounts and Password Synchronization" on page 137

  Refer to this topic if you are using Oracle Business Intelligence Delivers or if you are using the BI Reporting and Publishing feature with BI Publisher integrated with our Oracle BI environment.

- "Oracle BI Communications Across Security Firewalls" on page 140

  Refer to this topic for information on deploying Oracle Business Intelligence when the BI web components reside in a De-Militarized Zone.

- "Improving Oracle BI Web Client Performance" on page 144

  Refer to this topic for strategies to improve performance for Oracle Web Clients.

Administrator Accounts and Password Synchronization

Review this topic if you are using the Oracle Business Intelligence Reporting and Publishing feature or the Oracle Business Intelligence Delivers feature in your Oracle Business Intelligence deployment. To enable functionality for these features requires that the BI Presentation Services component be made aware of the BI Publisher and BI Scheduler Administrator credentials. In addition, BI Publisher must be made aware of the Oracle BI Administrator credentials. The user names and passwords that BI Presentation Services uses to log in and authenticate with other components are stored in the BI Presentation Services Credential Store. BI Publisher is made aware of the Oracle BI credentials by setting integration-related parameters for username and password in the BI Publisher application.

Depending on the passwords policy at your company, you may be required to change administrator passwords on a periodic basis. This topic describes how the Administrator account credentials for the different components are used to enable functionality. Also described is how passwords need to be synchronized across the different BI components when a password change is made.

The following are the Oracle Business Intelligence Administrator accounts:

- Oracle BI Administrator account.

  The default administrator for Oracle BI has username "Administrator". This user is defined in every BI repository (rpd), and cannot be deleted. The username cannot be changed. The password may be changed using BI Administration Tool. Refer to the Oracle Business Intelligence Server Administration Guide for more information on the Oracle BI Administrator account.

- Oracle BI Publisher Administrator account.

  The default username and password for the account on installation is administrator/Administrator. The username cannot be changed. The password may be changed from the BI Publisher application > Admin tab > Users.

- Oracle BI Scheduler Administrator account.

  There is no default administrator account defined upon installation of BI Scheduler. The BI Scheduler Administrator account is specified during initial configuration of the BI Scheduler component after installation. See the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for information on the configuration of BI Scheduler after installation. The Scheduler administrator that you specify must be a user in the BI repository with Administrator group membership assigned.

BI Presentation Services and BI Scheduler

Communication occurs between BI Presentation Services and BI Scheduler for BI Delivers. When a web user submits an iBot request, BI Presentation Services establishes communication with BI Scheduler. BI Presentation Services uses the username and password stored in its credential store under the alias of "admin" to connect to BI Scheduler. This username and password must match the BI Scheduler Administrator credential for successful authentication to occur.

The BI Scheduler Administrator account must be an account defined in the BI repository. This user must be assigned to the Administrator Group in the repository. This administrator account is required for two purposes:

- BI Scheduler uses its administrator credentials to run jobs on behalf of users that have submitted job requests. It uses the Oracle BI Server impersonation feature to impersonate users and submit the jobs on their behalf. To be able to use the impersonation feature, BI Scheduler must connect to BI Server as a repository administrator user.
- BI Presentation Services uses the credential to execute queries against the System subject area. This requires connecting as an repository administrator user. BI Presentation Services uses the credential stored under the alias "admin" in its credential store to make the connection.

The BI Scheduler Administrator account may be set to match the Oracle BI Administrator account. See the topic "Synchronizing Oracle BI Administrator Password Changes" on page 139.

Synchronizing BI Scheduler Administrator Account Changes

If you change the BI Scheduler Administrator username or password or both, you must synchronize this change with other components:

- Update the credential stored in the BI Presentation Services Credential Store under the alias admin. If you have multiple instances of BI Presentation Services in your Oracle BI deployment, then you must update the credential store for each BI Presentation Services instance to reflect this credential change.

- Ensure that a user with the new Scheduler Administrator username and password also exists in the repository and that this user has the Administrator group membership assigned to it. If you change the password for the BI Scheduler Administrator, you must change the password for that user in the repository to match the password reset.

NOTE: For information on setting or changing the BI Scheduler Administrator account, see the chapter on configuring the BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*. This chapter also provides information on storing the BI Scheduler Administrator credentials in the BI Presentation Services Credential Store. For information on creating users in the BI repository and assigning Administrator group membership to users, refer to the *Oracle Business Intelligence Server Administration Guide*.

BI Presentation Services and BI Publisher

The Oracle BI Reporting and Publishing feature provides the capability to integrate BI Publisher with Oracle BI. Users can access the BI Publisher application from the More Products > BI Publisher link in BI Presentation Services.

BI Presentation Services establishes a connection with BI Publisher. It uses the BI Publisher Administrator username and password stored with the alias bipublisheradmin in the BI Presentation Services Credential Store.

NOTE: The BI Publisher Administrator credentials may be stored under a different alias in the credential store. You must direct BI Presentation Services to access the appropriate alias by specifying the credential alias in the BI Presentation Services instanceconfig.xml file. Refer to the chapter on configuring BI Publisher in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide* for more information.

Synchronizing BI Publisher Administrator Account Changes

If you change the BI Publisher Administrator credentials in the BI Publisher application, you must update the Presentation Services Credential store to reflect this change. If you have multiple Presentation Services instances in your deployment you must reflect the credential change in the credential store for each Presentation Services instance.

NOTE: For information on user administration for BI Publisher, refer to the *Oracle Business Intelligence Publisher User's Guide*. For information on storing the BI Publisher Administrator credentials in the BI Presentation Services Credential Store, refer to the chapter on configuring BI Publisher in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

Synchronizing Oracle BI Administrator Password Changes

The Oracle BI Administrator with username "Administrator" is a special user defined in the BI repository (rpd). If you have set the Scheduler Administrator account to match the credentials of the Oracle BI Administrator account, then each time the password for user Administrator is changed in the rpd, you must synchronize this change:

- Reset the password for the Scheduler Administrator using either Job Manager or the schconfig utility to match the password change for the Oracle BI Administrator account. For more information on setting the BI Scheduler password using Job Manager or the schconfig utility, refer to the chapter on configuring BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.
- Update the credential stored in the BI Presentation Services Credential Store under the alias admin. If you have multiple instances of BI Presentation Services in your Oracle BI deployment, then you must update the credential store for each BI Presentation Services instance to reflect this credential change. For more information on adding the BI Scheduler Administrator credentials to the BI Presentation Services Credential Store refer to the chapter on configuring BI Scheduler in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

If you change the Oracle BI Administrator password, then the BI Publisher application integration parameters of for BI Administrator username and password must be changed:

- Log in to BI Publisher as administrator. In the Admin tab of the BI Publisher application, select Integration - Oracle BI Presentation Services. In the Admin Password fields, enter the new password that was set for Oracle BI Administrator.
- If you are using the BI Server as a data source for BI Publisher, then in the Admin tab of the BI Publisher application, select JDBC Data Sources. In the Username field, enter the new password that was set for Oracle BI Administrator.
- If you are using BI Server as the Security Model in BI Publisher, then in the Admin tab of the BI Publisher application, select Security Configuration. In the Username field, enter the new password that was set for Oracle BI Administrator.

Oracle BI Communications Across Security Firewalls

In enterprise deployments, the Web Server typically resides in a De-Militarized Zone (DMZ) in a web tier separated from the public Internet and the corporate Intranet by firewalls. Oracle Business Intelligence supports this partitioning by providing a BI Presentation Services Plug-in component that can reside in the web tier allowing the deployment of BI Presentation Services within the intranet.

The BI Presentation Services Plug-in is of two types:

- For deployments using J2EE based application servers, it consists of a Java Servlet deployed in a web container on the J2EE server.
- For deployments using Internet Information Services (IIS) as the web server, the BI Presentation Services Plug-in is an ISAPI plug-in.

Figure 6 on page 141 depicts the deployment of Oracle BI web components (BI Presentation Services and BI Presentation Services Plug-in) when the Web Server servicing Oracle BI resides in a DMZ. Two scenarios, one using IIS and the other with a J2EE based Application Server are shown.

v

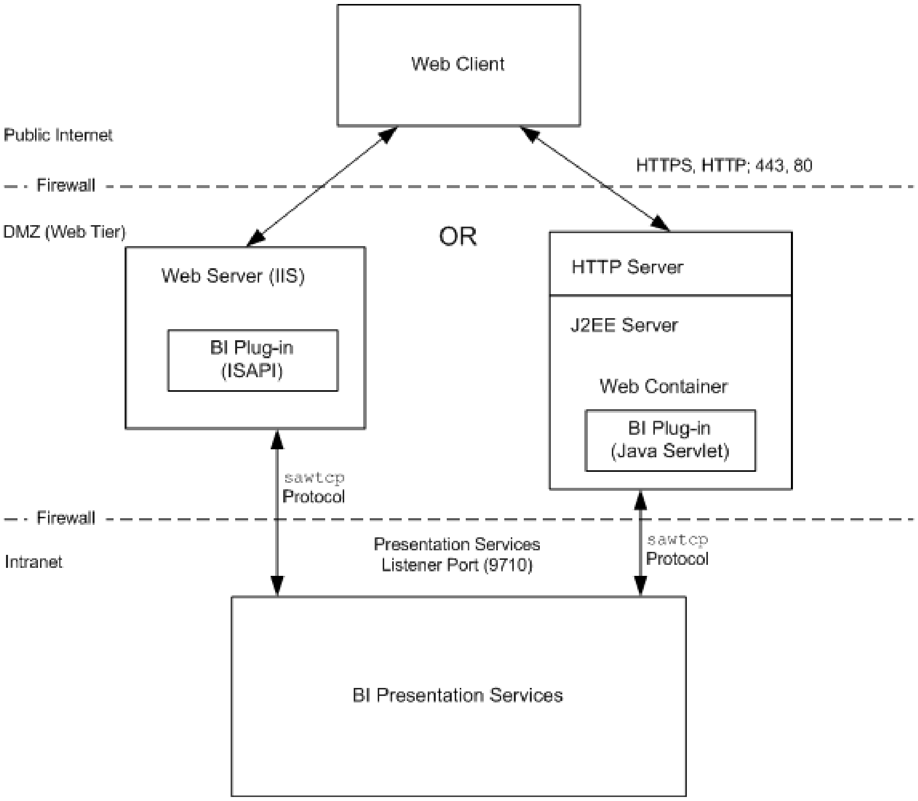

Figure 6. Deployment of Oracle BI Web Components and Communication Across Firewalls

The BI Presentation Services Plug-in, labeled in Figure 6 as BI Plug-in, directs web requests from Oracle BI clients to BI Presentation Services. The communication occurs using a proprietary protocol called sawtcp that is TCP/IP based. BI Presentation Services listens to remote procedure calls from one or more BI Presentation Services Plug-ins on the Listener port (default 9710).

The firewall separating the web tier from the corporate intranet must allow communication over the sawtcp TCP/IP protocol with the BI Presentation Services Listener port open.

Changing the BI Presentation Services Listener Port

The port that BI Presentation Services listens on can be configured in the BI Presentation Services configuration file instanceconfig.xml by creating and setting the RPC/Listener element, as shown in the following procedure.

The BI Presentation Services is configured by setting parameters in the configuration file instanceconfig.xml. The instanceconfig.xml file is located in the following directories:

- Under Windows: OracleBIData_HOME\web\config
- Under Linux or UNIX: OracleBIData_HOME/web/config

To change the Presentation Services listener port

1. Open the instanceconfig.xml file for editing.
2. Locate the <WebConfig> element.
3. Within the <WebConfig> tags, create the element <RPC> and assign it the RPC Listener Port value for the BI Presentation Services instance, as shown in the following example.

```
<WebConfig>
   <ServerInstance>
   ...
      <RPC>
         <Listener port="9715" />
      </RPC>
   ...
   </ServerInstance>
</WebConfig>
```

   NOTE: In the preceding example, the RPC Listener port for the BI Presentation Services instance has been changed to port 9715 from the default of 9710.

4. Save changes to the file.

Configuring BI Presentation Services Plug-in when BI Presentation Services Listener Port Has Been Changed

This topic contains information about configuring the BI Presentation Services Plug-in when the BI Presentation Services listener port has been changed.

The changes are made to two files:

- "Using the web.xml File" on page 142
- "Using the isapiconfig.xml File" on page 143

Using the web.xml File

The web.xml file is located in the Oracle BI applications WEB-INF directory.

To modify BI Presentation Services Plug-in (web.xml)

1. Open the web.xml file for editing.
2. Locate the oracle.bi.presentation.sawserver.Port parameter and set the port value as shown:

```
<init-param>
   <param-name>oracle.bi.presentation.sawserver.Host</param-name>
   <param-value><BI Presentation Services Host></param-value>
</init-param>
```

```
<init-param>
   <param-name>oracle.bi.presentation.sawserver.Port</param-name>
   <param-value>9715</param-value>
</init-param>
```

NOTE: In the preceding example , BI Presentation Services Plug-in communicates with BI Presentation Services on port 9715.

3 If you have multiple BI Presentation Services instances in your deployment, locate the oracle.bi.presentation.Sawservers parameter and set the ports as shown:

```
<init-param>
   <param-name>oracle.bi.presentation.Sawservers</param-name>
   <param-value>Server1:9715;Server2>:9715</param-value>
</init-param>
```

NOTE: In the above example, BI Presentation Services Plug-in communicates with the each of the two BI Presentation Services instances, Server1 and Server2, on port 9715, where both BI Presentation Services instances are configured to listen on port 9715.

Refer to Chapter 3, “Clustering, Load Balancing, and Failover in Oracle Business Intelligence,” for more information on configuration settings for a multi-server deployment of Oracle BI.

4 Save changes to the file.

5 Copy the web.xml file to OracleBI_HOME\web\app\WEB-INF on Windows and to OracleBI_HOME/web/app/WEB-INF on Linux.

6 Restart your Java Servlet container.

NOTE: If you have multiple instances of BI Presentation Services Plug-in, you must perform the configuration as described for all instances.

Using the isapiconfig.xml File

Use the following procedure to configure BI Presentation Services Plug-in to communicate with BI Presentation Services on a port other than the default port. The isapiconfig.xml file is located in the directory OracleBIData_HOME\web\config.

To modify BI Presentation Services Plug-in (isapiconfig.xml)

1 Open the isapiconfig.xml file for editing.

2 If you have a single instance of BI Presentation Services in your deployment, locate the entry:

```
<ServerConnectInfo address="<BI Presentation Services Hostname>" port="9710"/>
```

3 Modify the port value to match the Listener port set for BI Presentation Services:

```
<ServerConnectInfo address="<BI Presentation Services Hostname>" port="9715"/>
```

4 If you have multiple instances of BI Presentation Services in your deployment, the entries should be similar to the following example:

```
<ServerConnectInfo>
   <Hosts>
      <Host address="Server1" port="9715"/>
      <Host address="Server2" port="9715"/>
   </Hosts>
</ServerConnectInfo>
```

where:

- The Host sub-element defines the BI Presentation Services host and port pair.
- Set the ports to match the Listener port for the BI Presentation Services instances.

Refer to Chapter 3, "Clustering, Load Balancing, and Failover in Oracle Business Intelligence," for more information on configuration settings for a multi-server deployment of Oracle BI.

5. Save changes to the file.
6. Restart IIS.

   NOTE: If you have multiple instances of BI Presentation Services Plug-in, you must perform the configuration as described for all instances.

Improving Oracle BI Web Client Performance

This topic describes some recommendations to improve performance for Oracle Web Clients.

Static File Caching

Performance of the Oracle BI web client can be improved by caching small frequently used static files such as .javascript, .gif and .css files. By enabling caching and content expiration on the web server, web browsers can be directed to how often they should reload the static files from the server.

NOTE: Static file caching is enabled by default for Oracle HTTP Server when Oracle Business Intelligence is deployed using Oracle Application Server. The Oracle HTTP Server must be installed along with the J2EE Server for this automatic configuration to occur. The content expiry time is set to seven days by default.

The procedures shown are for the following web servers:

- Microsoft IIS Server
- Apache HTTP Server
- Oracle Containers for J2EE (OC4J)

As an example, static file content is set to expire after seven days.

Microsoft IIS Server

To set up static file caching for Microsoft IIS Server

1. On the Web Server machine, navigate to Start > Settings > Control Panel > Administrative Tools.

2. Run Internet Service Manager.

3. In Internet Service Manager, right-click on Default Web Site.

   NOTE: You may specify content expiration at the individual web site folder level or virtual directory level or for a file.

4. In Default Web Site Properties, click the HTTP Headers tab.

5. Check the Enable Content Expiration check box.

6. Select Expire After, and specify the value of seven.

   NOTE: This sets expiration for static files after seven days. Specify a value appropriate for your deployment.

7. Restart IIS

Apache HTTP Server

Use the procedure below to specify static file caching and content expiration when your HTTP Server is Apache based.

To set up static file caching for Apache HTTP Server

1. On the Web server machine, open the file httpd.conf for editing.

   This file is located in the Web server installation directory.

2. Verify that the following directive is included and not commented out:

   ```
   LoadModule expires_module modules/mod_expires.so
   ```

   NOTE: For Apache versions prior to 1.3.15 on Windows, the directive is LoadModule expires_module modules/ApacheModuleExpires.dll

3. Add the following lines to the file below the directive specified in step 2:

   ```
   ExpiresActive On

   <IfModule mod_expires.c>
   ExpiresByType image/gif "access plus 7 days"
   ExpiresByType image/jpeg "access plus 7 days"
   ExpiresByType application/x-javascript "access plus 7 days"
   ExpiresByType text/css "access plus 7 days"
   </IfModule>
   ```

   NOTE: As an example, content is set to expire in seven days. Set a value that is appropriate for your deployment.

4 Save the file.

5 Restart the HTTP Server.

Oracle Containers for J2EE (OC4J)

Oracle Containers for J2EE (OC4J) may be used as a standalone web server. OC4J can also be configured for static file caching. Use the following procedure to configure static file caching for OC4J.

Use the orion-web.xml file, located in one of the following directories:

- $ORACLE_HOME\ j2ee\home\application-deployments\analytics\analytics
- $ORACLE_HOME/ j2ee/home/application-deployments/analytics/analytics

To set up the static file Caching for OC4J

1 Open the BI application orion-web.xml file for editing.

2 Add the following lines into <orion-web-app> session:

```
<expiration-setting expires="604800" url-pattern="*.css"/>
<expiration-setting expires="604800" url-pattern="*.js"/>
<expiration-setting expires="604800" url-pattern="*.gif"/>
<expiration-setting expires="604800" url-pattern="*.jpg"/>
<expiration-setting expires="604800" url-pattern="*.png"/>
```

NOTE: As an example, the content expiration is set to seven days where the time unit is seconds. Set this to a value that is appropriate for your deployment.

3 Restart OC4J.

Static File Bypass

In deployments using J2EE based application servers, performance can be improved by configuring the HTTP Server to serve the static files. By default, the static file requests for Oracle BI are served by the J2EE server.

Because the Oracle BI static files reside outside the HTTP Server's document root, you must configure the HTTP Server to access the files from a file system that is not the document root for the HTTP Server. Consult your vendor documentation for more information on configuring the HTTP Server to access files not residing in the document root.

On Apache HTTP Servers, the Alias directive may be used to map file systems that lie outside documentroot to the web space.

Use the following procedure to configure the bypass of Oracle BI static files from the J2EE server and directing Oracle HTTP Server to serve the static file requests.

To bypass static files when using Oracle Application Server

1 Open the httpd.conf file for editing.

This file is located in $ORACLE_HOME/Apache/Apache/conf.

2 Verify that the following directive is not commented out in the httpd.conf file:

```
LoadModule expires_module modules/mod_expires.so
```

NOTE: For Apache versions prior to 1.3.15 on Windows, the directive is LoadModule expires_module modules/ApacheModuleExpires.dll

3 Add the following configuration into the httpd.conf file:

```
<Directory $ORACLE_HOME\j2ee\home\applications\analytics\analytics\res>
Order allow,deny
Allow from all
</Directory>

<Directory $ORACLE_HOME\j2ee\home\applications\analytics\analytics\olh>
Order allow,deny
Allow from all
</Directory>

Alias /OBIContent_res
$ORACLE_HOME\j2ee\home\applications\analytics\analytics\res

Alias /OBIContent_olh
$ORACLE_HOME\j2ee\home\applications\analytics\analytics\olh


<VirtualHost *:*>
#ServerName bi.mycompany.com
RewriteEngine on
RewriteRule ^/analytics/res/(.*)$ /OBIContent_res/$1 [PT]
RewriteRule ^/analytics/olh/(.*)$ /OBIContent_olh/$1 [PT]
</VirtualHost>
```

where:

The aliases OBIContent_res and OBIContent_olh are used to map the static files located in the following directories on the J2EE server:

- $ORACLE_HOME\j2ee\home\applications\analytics\analytics\res
- $ORACLE_HOME\j2ee\home\applications\analytics\analytics\olh

NOTE: Replace bi.mycompany.com with the virtual host name for your deployment.

4 If SSL has been enabled on the HTTP Server, add the following lines to the ssl.conf file located in the same directory as httpd.conf:

```
RewriteEngine on
RewriteRule ^/analytics/res/(.*)$ /OBIContent_res/$1 [PT]
RewriteRule ^/analytics/olh/(.*)$ /OBIContent_olh/$1 [PT]
```

```
As shown below:
<VirtualHost _default_:443>
# General setup for the virtual host

DocumentRoot "C:\OAS10.1.3\OracleAS_1\Apache\Apache\htdocs"
ServerName bi.mycompany.com


RewriteEngine on
RewriteRule ^/analytics/res/(.*)$ /OBIContent_res/$1 [PT]
RewriteRule ^/analytics/olh/(.*)$ /OBIContent_olh/$1 [PT]

ServerAdmin you@your.address
```

5. Restart Oracle HTTP Server.

10 Integrating Oracle Internet Directory With Oracle Business Intelligence

This chapter describes the tasks necessary to configure Oracle Business Intelligence to perform user authentication using Oracle Internet Directory. It also describes provisioning for Oracle BI using Directory Integration Platform (DIP).

NOTE: This topic area assumes that Oracle Internet Directory has been installed and configured. Refer to the *Oracle Application Server Installation Guide 10g (10.1.4.0.1)* for the desired platform for details on installing Oracle Internet Directory. Refer to the *Oracle Internet Directory Administrator's Guide 10g (10.1.4.0.1)* for information on configuring and using Oracle Internet Directory.

Using Oracle Internet Directory for User Authentication in Oracle BI

The following tasks are required to configure Oracle Business Intelligence to perform user authentication using Oracle Internet Directory:

- "Creating an LDAP Server Entry in the Repository for OID" on page 149
- "Configuring the Initialization Block Used for User Authentication" on page 150

Creating an LDAP Server Entry in the Repository for OID

This task is part of "Using Oracle Internet Directory for User Authentication in Oracle BI."

Create a new LDAP Server entry in the repository (rpd) for Oracle Internet Directory using the following procedure.

To modify the repository for user authentication in OID

1. Open the rpd in the BI Administration Tool and select Manage > Security from the application menu.
2. From the Security Manager menu, choose Action > New > LDAP Server.
3. In the General tab, enter values for fields as shown in the following example:

```
Hostname = <Oracle Internet Directory hostname>
Port number = <Oracle Internet Directory port>
LDAP version = LDAP 3
```

```
Base DN = <Base distinguished name (DN)>
Bind DN = <Distinguished name required to bind to OID>
Bind password = <Password of bind DN>
```

where the Base DN field identifies the starting point of the authentication search.

If the Bind DN and Bind password entries are blank, anonymous binding is assumed. By default, OID supports anonymous binding.

For example:

```
Hostname: oid.mycompany.com
Port number: 389
LDAP version: LDAP 3
Base DN: ou=Business Intelligence,c=us,o=mycompany
Bind DN: cn=orcladmin
Bind password: <orcladmin password >
```

where orcladmin is the Oracle Internet Directory superuser.

4 In the Advanced tab, set the attribute type that will be used to uniquely identify users:

Uncheck the “Automatically generated” check box and enter the attribute name in the text box.

For example, enter “mail”.

The mail attribute identifies a user's primary e-mail address.

For example: mail: user.name@mycompany.com

Refer to the *Identity Management User Reference* 10g (10.1.4.0.1) for a list of standard LDAP attributes used by Oracle Internet Directory.

5 Return to the General tab and click on the Test Connection button to ensure the connection to OID server is successful.

Configuring the Initialization Block Used for User Authentication

This task is part of “Using Oracle Internet Directory for User Authentication in Oracle BI.”

If an initialization block that is used for user authentication is already created in the repository, then you must modify it to use LDAP authentication.

NOTE: This initialization block is typically named Authentication. If no initialization block for user authentication exists in the repository, you must create one and configure it to use LDAP authentication.

To configure an initialization block for user authentication in OID

1 In Administration Tool, open the Variable Manager by selecting Manage > Variable from the menu bar.

2. Under Session > Initialization Blocks, select the existing init block for user authentication (typically named Authentication).

   If no such init block exists, create a new init block.

   - Right-click and select New Initialization Block.
   - In the Session Variable Initialization window, enter a name for the init block.

     For example, Authentication.

   - Check the "Required for authentication" check box.

3. In the Session Variable Initialization Block window, click on the "Edit Data Source..." button.

4. In the Session Variable Initialization Block Data Source window, select LDAP as the Data Source Type from the drop-down.

5. Click on the Browse button and select the LDAP Server that was created in the task"Creating an LDAP Server Entry in the Repository for OID" on page 149.

   Click OK. Click OK again to close the Data Source window.

6. In the Variable Target box of the Session Variable Initialization Block window, select the Edit Data Target button.

7. In the Session Variable Initialization Block Variable Target window, the USER session variable should be listed in the Variable column. If this variable exists, proceed to the next step.

   If this variable does not exist, click on the New button. In the System Session Variable window, enter "USER" in the Name field. Click OK. Click OK when asked to confirm if you want to use this name.

8. In the Session Variable Initialization Block Variable Target window, set the LDAP variable for the USER variable to the LDAP attribute that should be mapped to the USER variable. Note that the LDAP variable should exactly match the case of the LDAP attribute in Oracle Internet Directory.

   For example, LDAP variable = mail

9. Click OK to close the Session Variable Initialization Block Variable Target window.

10. Test the authentication by clicking on the Test button in the Session Variable Initialization Block window.

    Enter the credentials for a test user.

    In this example, mail was set as the attribute to uniquely identify users in Step 4 of "Creating an LDAP Server Entry in the Repository for OID" on page 149.

    User ID = user.name@mycompany.com

    Password = <password for user>

    The Results window should show the value of the LDAP variable that was set in Step 8. In this example, the mail variable value for the user will be returned.

In addition to basic user authentication, Oracle Internet Directory can also provide the Oracle BI Server with other attribute information, such as the user display. This information is contained in the LDAP variables that get passed to session variables during the process of user authentication.

Provisioning for Oracle BI Using Directory Integration Platform (DIP)

Provisioning enables an application to be notified of directory changes. Changes to user or group information, for example, can affect whether the application allows a user access to its processes and determines which resources can be used.

Oracle Directory Integration Platform (DIP) is a component of the Identity Management infrastructure and provides for provisioning tasks.

This topic area describes how to set up provisioning using DIP for Oracle Business Intelligence. Refer to chapter Deploying Provisioning-Integrated Applications in the *Oracle Identity Management Integration Guide 10g (10.1.4.0.1)* for detailed information.

The following tasks are required to configure DIP to provision for Oracle BI:

- "Creating an Identity for Oracle BI Application in OID" on page 152
- "Adding Oracle BI Application to Privileged Groups" on page 153
- "Implementing the Directory Integration Platform (DIP) Standard Package" on page 153
- "Defining Provisioning Policies for Oracle BI" on page 154

Creating an Identity for Oracle BI Application in OID

This task is part of "Provisioning for Oracle BI Using Directory Integration Platform (DIP)."

A unique identity for the Oracle BI application must be created in OID Directory Information Tree (DIT). This is required since Oracle BI does not get added to any realm in OID during installation.

The ldap commands are used to register or create an identity for the Oracle BI application. Refer to chapter Oracle Internet Directory Data Management Tools in the *Oracle Identity Management User Reference 10g (10.1.4.0.1)* for syntax and usage of the ldap commands, including ldapmodify.

To create an Identity for Oracle BI Application in OID

This procedure creates an identity for Oracle BI in DIT as a dummy application.

- Run the following command on the machine where OID and DIP have been installed:

  ```
  ldapmodify -c -a -h $host -p $port -D cn=orcladmin -w $passwd -v -f uspapp.dat
  ```

  Where:

  - $host = Hostname of the machine where OID and DIP are installed
  - $port = 389 for non ssl connection
  - $passwd = Password for superuser orcladmin

  The example creates the application identity in the cn=Products, cn=OracleContext container, and assumes the application name and type are DummyappOnline and DUMMYAPP:

```
dn:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleConte
xt
```

The contents to the uspapp.dat file are provided in Appendix C, "Supporting Files For Provisioning Using Directory Integration Platform (DIP)."

Adding Oracle BI Application to Privileged Groups

This task is part of "Provisioning for Oracle BI Using Directory Integration Platform (DIP)."

Add Oracle BI to privileged groups in Oracle Internet Directory using the procedure outlined below. Also grant necessary privileges to the application so that it can be manipulated using command line utilities.

Refer to chapter Deploying Provisioning-Integrated Applications in the *Oracle Identity Management Integration Guide 10g (10.1.4.0.1)* for a list of the common privileged groups in OID.

To add Oracle BI Application to Privileged Groups

- Run the following command:

  ```
  ldapmodify -a -h $host -p $port -D cn=orcladmin -w $passwd -v -f appriv.dat
  ```

  The following lines in the apppriv.dat file grants "create user" privileges in all realms to the DummyappOnline application:

  ```
  dn: cn=OracleDASCreateUser,cn=Groups,cn=OracleContext
  changetype: modify
  add: uniquemember
  uniquemember:
  orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleConte
  xt
  ```

The entire contents of the appriv.dat file are provided in Appendix C, "Supporting Files For Provisioning Using Directory Integration Platform (DIP)."

Implementing the Directory Integration Platform (DIP) Standard Package

This task is part of "Provisioning for Oracle BI Using Directory Integration Platform (DIP)."

The PL/SQL interface provided by the Directory Integration Platform (DIP) has standard package (sql files) which access and manipulate data in the backend tables in the database where provisioning is to happen. Database objects must first be created before the standard package is able to access the tables.

Examine the contents of the following files and modify as necessary for your database. Adapt and run the following SQL in the order listed below.

1. ldap_ntfy_30.sql
2. ldap_ntfy_30.pks
3. ldap_ntfy_30.pkb

The PL/SQL package consists of the following:

- ldap_ntfy_30.sql: DDLs for table and type definitions
- ldap_ntfy_30.pks: Interface for the package LDAP_NTFY
- ldap_ntfy_30.pkb: Implementation of the package LDAP_NTFY

The name of the PL/SQL package can be changed. However, the procedures within the package should have the exact same signature as in the source file.

The contents of the files listed above are provided in Appendix C, "Supporting Files For Provisioning Using Directory Integration Platform (DIP)." These are sample files only. You must modify them before using.

Defining Provisioning Policies for Oracle BI

You use the Provisioning Subscription tool (oidprovtool) to create provisioning profile entries in the directory. Refer to chapter Oracle Directory Integration Platform Tools in *Oracle Identity Management User Reference 10g (10.1.4.0.1)* for syntax and usage of oidprovtool.

To define Provisioning Policies for Oracle BI

1. Run the following command to define the provisioning policies for the dummy application:

   ```
   oidprovtool
   ```

   The following example shows the usage of oidprovtool. Alter the parameters according to your application's provisioning requirements.

   ```
   oidprovtool operation=create profile_mode=OUTBOUND profile_status=enabled
   ldap_host=$host ldap_port=$port ldap_user_dn='cn=orcladmin'
   ldap_user_password=$passwd
   application_dn='orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Product
   s,cn=OracleContext' organization_dn='dc=us,dc=oracle,dc=com' interface_name=<pl/
   sql package name eg.LDAP_NTFY
   interface_version=3.0 interface_type=PLSQL
   interface_connect_info=$db_host:1521:$connect:$db_userid:$db_userpassword
   user_data_location='cn=User
   Properties,orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=
   OracleContext' default_provisioning_policy=PROVISIONING_REQUIRED
   application_type=DummyApp
   application_display_name='DummyApp Online'
   event_subscription='USER:dc=us,dc=oracle,dc=com:ADD(*)'
   event_subscription='USER:dc=us,dc=oracle,dc=com:MODIFY(*)'
   event_subscription='USER:dc=us,dc=oracle,dc=com:DELETE'
   ```

```
event_subscription='GROUP:dc=us,dc=oracle,dc=com:DELETE'
event_subscription='GROUP:dc=us,dc=oracle,dc=com:ADD(*)' schedule=30
max_retries=5
```

Where:

- $db_host = host name of the Oracle database server where the provisioning is to be done
- $connect = Oracle SID (directory database connect string, or the net service name in tnsnames.ora)
- $db_userid = db user
- $db_userpassword = password for db user
- ADD(*) = attributes you want provisioned through DIP when a new entity is added in OID. Use asterisk (*) to provision all attributes, or provide a list of required attributes.

If provisioning for an application or database is to occur when a user is added in OID, then the procedure PutOIDEvents gets invoked. Add custom code in this procedure per your application provisioning requirements.

11 Enabling Oracle Single Sign-On for Oracle Business Intelligence

This chapter describes the integration of Oracle Business Intelligence with Oracle Single Sign-On.

When Oracle Business Intelligence is registered with the Oracle Single Sign-On server as a partner application, user authentication is delegated to the Oracle Single Sign-On server. An Oracle HTTP Server module called mod-osso transmits header values such as username of the authenticated user to Oracle BI, specifically, to BI Presentation Services.

In order for mod_osso to redirect the user to the Single Sign-On server, the Oracle BI URL must be protected. You can secure URLs in one of two ways: statically or dynamically. Static directives simply protect the application, ceding control over user interaction to mod_osso.

When a web user tries to access Oracle BI with Oracle SSO enabled, the user is redirected to the Single Sign-On server and is challenged for credentials via a JSP login page. After verifying the credentials in Oracle Internet Directory, the server sets an SSO session cookie and passes an authentication token to Oracle BI.

If the user is already logged in to the Single Sign-On server and then tries to access Oracle BI, the user is redirected to the Single Sign-On server but is not challenged for credentials. The SSO session cookie is used to validate the user identity. The server passes an authentication token to Oracle BI.

BI Presentation Services then utilizes the BI Server Impersonation feature to create a connection to the BI Server on behalf of the authenticated end user. Additional authorizations for the user takes place in the BI repository that determines, for example, the security groups associated to the user. This in turn determines subject area access, presentation catalog access and data visibility that must be applied for the user.

The user request is processed, and BI Presentation Services then serves the content.

The following steps enable Oracle Single Sign-On with Oracle BI:

- "Registering Oracle BI as a Partner Application to the Oracle Single Sign-On Server" on page 158
- "Configuring Oracle BI for SSO" on page 159

  NOTE: This step includes statically protecting the Oracle BI URL.

- "Configuring BI Presentation Services to Use the Impersonator User" on page 160
- "Configuring BI Presentation Services to Operate in the SSO Environment" on page 166

NOTE: It is assumed that user authentication with Oracle Internet Directory has already been configured for Oracle BI. This is a necessary configuration. Refer to the chapter Chapter 10, "Integrating Oracle Internet Directory With Oracle Business Intelligence," for more information on setting up and using Oracle Internet Directory for BI user authentication.

Registering Oracle BI as a Partner Application to the Oracle Single Sign-On Server

In the following procedure, the command script you use depends on your operating system:

Under UNIX: use ORACLE_HOME/sso/bin/ssoreg.sh

Under Windows: use ORACLE_HOME\sso\bin\ssoreg.bat

To register Oracle BI as a partner application to Oracle SSO

1. Ensure that Oracle Identity Management is running.
2. On the machine that hosts the Oracle Single Sign-On server, set the ORACLE_HOME environment variable to point to the directory where Oracle Single Sign-On server is installed.
3. On the machine that hosts the Oracle Single Sign-On server, run the following command:

```
ssoreg.sh or ssoreg.bat
-oracle_home_path orcl_home_path
-site_name <site_name>
-config_mod_osso TRUE
-mod_osso_url <mod_osso_url>
[-virtualhost]
[-update_mode CREATE | DELETE | MODIFY]
[-remote_midtier]
[-config_file config_file_path]
[-admin_info admin_info]
[-admin_id adminid]
```

where:

- site_name is the name of the site. It is typically the effective host name and port of the partner application being registered. For example, bi.mycompany.com:7777, or bi.mycompany.com:443 (if SSL is enabled).
- config_mod_osso is set to TRUE to indicate the application being registered is mod_osso. You must include config_mod_osso for osso.conf to be generated.
- mod_osso_url is the effective URL of the partner application. This is the URL that is used to access the partner application. For example, http://bi.mycompany.com:7777, or https://bi.mycompany.com:443 (if SSL is enabled).

Refer to the chapter on configuring and administering partner applications in the *Oracle Application Server Single Sign-On Administrator's Guide 10g (10.1.4.0.1)* for details of the syntax and parameters for ssoreg.

For example:

```
Oracle_HOME/sso/bin/ssoreg.sh -oracle_home_path Oracle_HOME -config_mod_osso
TRUE -site_name bi.mycompany.com:7777 -remote_midtier -config_file Oracle_HOME/
Apache/Apache/conf/osso/biosso.conf -mod_osso_url http://bi.mycompany.com:7777
```

where:

- remote_midtier specifies that the mod_osso partner application (in this case, the Oracle BI application) to be registered is at a remote midtier.
- biosso.conf is the name of the resulting obfuscated osso configuration file created. The details of the registration can be checked at Oracle_HOME/sso/log/ssoreg.log.

Configuring Oracle BI for SSO

Apply this procedure to the obfuscated osso configuration file (biosso.conf) that was created in the topic "Registering Oracle BI as a Partner Application to the Oracle Single Sign-On Server" on page 158.

To configure Oracle BI for SSO

1. Copy the file biosso.conf to the directory Oracle_HOME/Apache/Apache/conf/osso on the machines where the BI Presentation Services Plug-in Java Servlet is running.
2. Open the mod_osso.conf file for editing.

   This file is located in Oracle_HOME/Apache/Apache/conf/.
3. Add the following directive:

   ```
   OssoConfigFile Oracle_HOME/Apache/Apache/conf/osso/biosso.conf
   ```
4. Statically protect the Oracle BI URL (identified by the keyword "analytics") by adding the following lines to mod_osso.conf:

   ```
   <IfModule mod_osso.c>
   ..<Location /analytics>
   ....AuthType Basic
   ....require valid-user
   ..</Location>
   </IfModule>
   ```
5. Modify the above section as follows:

   ```
   <Location /analytics>
   ....Header unset Pragma
   ....OssoSendCacheHeaders off
   ....AuthType Basic
   ....require valid-user
   </Location>
   ```

   After modifications described above, the mod_osso.conf file should be similar to the following example:

   ```
   OssoIpCheck off
   OssoIdleTimeout off
   OssoConfigFile Oracle_HOME/Apache/Apache/conf/osso/biosso.conf
   .......................
   <IfModule mod_osso.c>
   ..<Location /analytics>
   ```

```
....Header unset Pragma
....OssoSendCacheHeaders off
....AuthType Basic
....require valid-user
..</Location>
</IfModule>
```

6 Modify the file httpd.conf (in Oracle_HOME/Apache/Apache/conf/) by uncommenting the following line:

```
include “Oracle_HOME/Apache/Apache/conf/mod_osso.conf”
```

7 Restart the Oracle HTTP server using the following command:

```
ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
```

If Oracle BI has been configured for communication over SSL, then perform these additional steps:

8 From the machine that hosts Oracle SSO server, copy the sso_apache.conf file located in Oracle_HOME/sso/config to the directory Oracle_HOME/Apache/Apache/conf on the machines that host the BI Presentation Services Plug-in (Java Servlet).

9 On the BI Presentation Services Plug-in (Java Servlet) host machines, modify the file httpd.conf (in Oracle_HOME/Apache/Apache/conf/) to add the following directive:

```
include "Oracle_HOME /Apache/Apache/conf/sso_apache.conf"
```

10 Modify the sso_apache.conf file to enable the SSL section and comment out the rewrite section (only the section shown in the example is enabled):

```
<IfDefine SSL>
...Oc4jExtractSSL on
...<Location /sso>
.......SSLOptions +ExportCertData +StdEnvVars

...</Location>
</IfDefine>
```

Configuring BI Presentation Services to Use the Impersonator User

If an impersonator user has already been created in the BI Server repository, and if the impersonator user credentials have been added to BI Presentation Services credential store with an entry in the instanceconfig.xml file pointing to this credential store, then you do not need to perform the steps outlined below. Proceed to the section “Configuring BI Presentation Services to Operate in the SSO Environment” on page 166.

The steps to configure BI Presentation Services to use the Impersonator user are:

- “Creating the Oracle BI Server Impersonator User” on page 161
- “Adding Impersonator User Credentials to Oracle BI Presentation Services Credential Store” on page 161

- “Configuring Oracle BI Presentation Services to Identify the Credential Store and Decryption Passphrase” on page 164

Creating the Oracle BI Server Impersonator User

BI Presentation Services uses the Oracle BI Impersonation feature to establish a connection to the BI Server on behalf of the authenticated end user. For this purpose, a special user that BI Presentation Services will utilize for impersonating the authenticated end user needs to be created. This document refers to this special user as the impersonator user.

To create the Oracle BI Server Impersonator User

Use this procedure to create the impersonator user in the BI Server repository.

1. Open the BI Server repository file (.rpd) using BI Administration Tool.
2. Select Manage > Security to display the Security Manager.
3. Select Action > New > User to open the User dialog box.
4. Enter a name and password for this user.

   For example, Name = Impersonator and Password = secret.

5. Click OK to create the user.

   Make this user a member of the group **Administrators**.

6. Double-click on the icon for the user that was created.
7. In the Group Membership portion of the dialog box, check the Administrators group to grant the user created above membership to this group.

For more information on creating users and granting Group membership, refer to the *Oracle Business Intelligence Server Administration Guide*.

Adding Impersonator User Credentials to Oracle BI Presentation Services Credential Store

For BI Presentation Services to be able to utilize the user created above for impersonation of the authenticated end user, it must be able to identify the impersonator user and obtain the impersonator user credentials. The impersonator user credentials must be added to the BI Presentation Services Credential Store. To obtain the impersonator user credentials, BI Presentation Services will search the credential store for a username-password credential with an alias of `impersonation`.

In the instructions provided below, it is assumed that the credential store being used by BI Presentation Services is the credential store XML file called credentialstore.xml. Other supported storage facilities for the credential store may be used to store the credentials. For more information on the BI Presentation Services Credential Store and the supported storage types, see Chapter 5, “Oracle BI Presentation Services Credential Store”.

Use this procedure to add the impersonator user credentials to the BI Presentation Services Credential Store with an alias of impersonation. The impersonator user credentials are added to the BI Presentation Services proprietary credential store named credentialstore.xml. This file is in the following locations:

- Windows: OracleBIData_HOME\web\config
- Linux or UNIX: OracleBIData_HOME/web/config

NOTE: You must perform this procedure for all instances of BI Presentation Services in your deployment. As an alternate, you can copy the credential store to which the impersonator credentials have been added to all BI Presentation Services machines.

To add impersonator user credentials to Oracle BI Presentation Services credential store

1. Open a command prompt window or command shell on the machine where BI Presentation Services has been installed.
2. Navigate to the directory OracleBI_HOME\web\bin or OracleBI_HOME/web/bin.
3. Execute the CryptoTools utility to add the impersonator user credentials to the BI Presentation Services Credential Store:

   ```
   cryptotools credstore -add -infile OracleBIData_HOME/web/config/
   credentialstore.xml
   ```

   For more information on the CryptoTool utility, its syntax and supported sub-commands, refer to "Using the CryptoTools Utility" on page 171.
4. Supply values for the prompted parameters, as shown in Table 13 on page 162.

Table 13. Adding Impersonator user credentials to credentialstore.xml using CryptoTools

Parameter or Prompt	Value or Input	Description
Credential Alias	impersonation	You must specify the value **impersonation** to identify the user as the impersonator user
Username	<name of the user>	Name of the user created in topic *Creating the Oracle BI Server Impersonator User*. For example, **Impersonator**.
Password	<password of the user>	Password of the user created in topic *Creating the Oracle BI Server Impersonator User*. For example, **secret**.
Do you want to encrypt the password?	y	Provide a passphrase. For example, **another_secret**.

Table 13. Adding Impersonator user credentials to credentialstore.xml using CryptoTools

Parameter or Prompt	Value or Input	Description
Passphrase for encryption	*<passphrase>*	
Do you want to write the passphrase to the xml?	n	For enhanced security, specify “n”. The passphrase will not be written to the credential store and must be provided in the instanceconfig.XML file.

For example:

```
cryptotools credstore -add -infile <OracleBIData>/web/config/credentialstore.xml
```

```
>Credential Alias: impersonation
>Username: Impersonator
>Password: secret
>Do you want to encrypt the password? y/n (y):
>Passphrase for encryption: another_secret
>Do you want to write the passphrase to the xml? y/n (n):
>File "OracleBIData_HOME/web/config/credentialstore.xml" exists. Do you want to
overwrite it? y/n (y):
```

The CryptoTools utility updates the credentialstore.xml file. This file is located in the OracleBIData\web\config directory on Windows and OracleBIData/web/config on Linux and UNIX.

After executing the CryptoTools utility with inputs as specified above, the credentialstore.xml file contains entries similar to the following:

```
<sawcs:credential type="usernamePassword" alias="impersonation">
   <sawcs:username>Impersonator</sawcs:username>
   <sawcs:password>
      <xenc:EncryptedData>
      <xenc:EncryptionMethod Algorithm="http://www.rsasecurity.com/rsalabs/pkcs/
schemas/pkcs-5#pbes2">
         <pkcs-5:PBES2-params Algorithm="http://www.rsasecurity.com/rsalabs/pkcs/
schemas/pkcs-5#pbkdf2">
         <pkcs-5:KeyDerivationFunc>
            <pkcs-5:Parameters>
            <pkcs-5:IterationCount>1024</pkcs-5:IterationCount>
            </pkcs-5:Parameters>
         </pkcs-5:KeyDerivationFunc>
         <pkcs-5:EncryptionScheme Algorithm="http://www.w3.org/2001/04/
xmlenc#tripledes-cbc"/>
         </pkcs-5:PBES2-params>
      </xenc:EncryptionMethod>
      <xenc:CipherData>
         <xenc:CipherValue>jeThdk8ZklnTlyKIat8Dkw</xenc:CipherValue>
      </xenc:CipherData>
      </xenc:EncryptedData>
   </sawcs:password>
</sawcs:credential>
```

Configuring Oracle BI Presentation Services to Identify the Credential Store and Decryption Passphrase

BI Presentation Services must be directed to the credential store that contains the impersonator user credentials. This is done by setting parameters in the BI Presentation Services configuration file, instanceconfig.xml. In addition, the passphrase that BI Presentation Services will use to decrypt the impersonator password credential must be specified.

The default location of this file is instanceconfig.xml file for editing. This file is located in the OracleBIData\web\config directory on Windows and in the OracleBIData/web/config directory on Linux or UNIX. This directory structure is the same on Linux platform.

NOTE: You must perform this configuration for all instances of BI Presentation Services in your deployment.

To configure Oracle BI Presentation Services to Identify the Credential Store and Decryption Passphrase

- Locate the <CredentialStore> node within the config.xml file.
- Specify attribute values as shown in the following example.

  If the <CredentialStore> node does not exist, create this element with sub-elements and attributes with attribute values given in the following example.

```
<WebConfig>
   <ServerInstance>
      <!-- other settings ... -->
      <CredentialStore>
      <CredentialStorage type="file" path="<path to credentialstore.xml>"
passphrase="<passphrase>"/>
      <!-- other settings ... -->
      </CredentialStore>
      <!-- other settings ... -->
   </ServerInstance>
</WebConfig>
```

Table 14 on page 165 summarizes the attributes and attribute values for the CredentialStorage element. For detailed information on the CredentialStore and CredentialStorage elements of the instanceconfig.xml file and for their settings when credential stores other than the XML file store are used, see Chapter 5, “Oracle BI Presentation Services Credential Store.”

Table 14. Attributes and Attribute Values for the CredentialStorage element

Attribute	Attribute Value	Description
type	file	This describes the type of credential store. Set to file for the proprietary XML file credential store.
path	*<path to XML file credential store (credentialstore.xml)>*	Location and filename for the XML file credential store. For example, OracleBI`Data_HOME/web/config/credentialstore.xml>`
passphrase	*<passphrase>*	Determines the passphrase used to decrypt encrypted files. Provide the value entered in step 4 under topic *Adding Impersonator User Credentials to Oracle BI Presentation Services Credential Store*. In the example provided, this value is **another_secret**.

After modification as described above, instanceconfig.xml contains entries as shown in the example:

```
<?xml version="1.0"?>
<WebConfig>
   <ServerInstance>
      <!-- other settings ... -->
      <CredentialStore>
         <CredentialStorage type="file" path="OracleBIData_HOME/web/config/
credentialstore.xml" passphrase="another_secret"/>
         <!-- other settings ... -->
      </CredentialStore>
      <!-- other settings ... -->
   </ServerInstance>
</WebConfig>
```

NOTE: Both the files, `credentialstore.xml` and `instanceconfig.xml` should be protected using OS filesystem protection capabilities as their combination could reveal a privileged user's password. Note that neither file on its own has enough information to expose the password.

Configuring BI Presentation Services to Operate in the SSO Environment

Perform the following configuration on all instances of BI Presentation Services in your deployment. Shut down the BI Presentation Services before making any changes.

The instanceconfig.xml file is located in the OracleBIData_HOME\web\config directory on Windows and OracleBIData_HOME/Data/web/config on Linux or UNIX.

To configure BI Presentation Services to Operate in the SSO Environment

1 Open instanceconfig.xml for editing.

2 Locate the <Auth> element.

If this does not exist, create this element, sub-elements and parameters under the ServerInstance tag as shown in the following example:

```
<ServerInstance>
<!-- other settings ... -->
   <Auth>
         <SSO enabled="true">
            <ParamList>

<!--IMPERSONATE param is used to get the authenticated user's username and is 
required -->
               <Param name="IMPERSONATE"
                  source="serverVariable"
                  nameInSource="REMOTE_USER"/>
            </ParamList>

<LogoffUrl> http://<OSSO_HOST:HTTP_PORT>/pls/orasso/
orasso.wwsso_app_admin.ls_logout?p_done_url=https%3A%2F%2F<BI_HOST:PORT>%2Fanal
ytics%2F</LogoffUrl>

<LogonUrl> http://<OSSO_HOST:HTTP_PORT>/pls/orasso/
orasso.wwsso_app_admin.ls_login</LogonUrl>
         </SSO>
   </Auth>
<!-- other settings ... -->
</ServerInstance>
```

3 Secure the machines that are permitted to communicate with BI Presentation Services directly.

This can be done by setting the Listener\Firewall node in instanceconfig.xml with the list of HTTP Server or servlet container IP addresses.

- In addition, the Firewall node must include the IP addresses of all BI Scheduler instances, BI Presentation Services Plug-In (Java Servlet) and BI Javahost instances.
- If any of these components are co-located with the BI Presentation Services, then address 127.0.0.1 must be added in this list as well.

NOTE: This setting does not control end-user browser IP addresses.

4 When using mutually-authenticated SSL, you must specify the Distinguished Names (DNs) of all trusted hosts in the Listener\TrustedPeers node.

For more information, see Chapter 6, “Enabling Secure Communication in Oracle Business Intelligence”.

```
For example:

<ServerInstance>
   <!-- other settings ... -->
      <Listener>
         <Firewall>
            <Allow address="127.0.0.1"/>
            <Allow address="192.168.1.100”/>
            <Allow address="192.168.1.101”/>
         </Firewall>
         <!-- other settings ... -->
      </Listener>
   <!-- other settings ... -->
</ServerInstance>
```

5 Verify that an entry pointing to the BI Presentation Services credential store exists in the instanceconfig.xml file. This credential store contains the credentials for the Impersonator user.

```
      CredentialStorage type="file" path="<OracleBIData>/web/config/
credentialstore.xml" passphrase="another_secret"/>
      <!-- other settings ... -->
      </CredentialStore>
```

6 Restart BI Presentation Services.

Additional Configuration When SSO is Enabled for Oracle BI and BI Publisher

If you are using the Oracle BI Reporting and Publishing feature and have deployed BI Publisher, then you must enable SSO for BI Publisher. Otherwise, users who access BI Publisher directly or by using the More Products > BI Publisher link in BI Presentation Services need to log on. To enable Oracle SSO with BI Publisher, refer to the topic on setting up Oracle Single Sign On in the *Oracle Business Intelligence Publisher User's Guide* for Release 10.1.3.2.

The following additional configuration must be done for BI Presentation Services and Publisher to communicate when SSO is enabled for both Oracle BI and BI Publisher.

To enable SSO for Oracle BI and BI Publisher

1 On the same machine where BI Presentation Services Plug-in has been deployed, deploy another Presentation Services Plug-in using the file analytics.ear.

Locate analytics.ear in the directory OracleBI_HOME/web/.

2 Name the new Plug-in analyticsSOAP.

Make the same modifications to the web.XML file for this analyticsSOAP servlet that were made to the web.XML file for the default “analytics” servlet.

3 Make the following modification to the file mod_osso.conf to statically protect analyticsSOAP.

mod_osso.conf is located in the directory Oracle_HOME/Apache/Apache/conf.

```
<Location /analyticsSOAP>
   require valid-user
   AuthType Basic
   Allow from All
   Satisfy any
</Location>
```

4 In the Admin tab of the BI Publisher application, select Integration - Oracle BI Presentation Services.

In the URL Suffix field, enter the following suffix:

```
analyticsSOAP/saw.dll
```

Ensure that all other fields on this page have been appropriately set as documented in the chapter on configuring BI Publisher in the *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

NOTE: In the Admin Username and Admin Password fields, the username and password for the Oracle BI administrator must be specified.

A Granting the Oracle BI Log On as Service Right

Operating System: Windows only.

As part of Chapter 4, “Deploying Oracle Business Intelligence for High Availability,” you are required to identify a Domain account under which all clustered Oracle Business Intelligence Servers and cluster controllers run. This Domain account must also have the *Log on as a service* right. Members of the Administrators group do not have this right by default. Therefore, if the cluster controllers are running under a Windows operating system, grant the Log on as a service right explicitly to this account on each computer using one of the following methods.

NOTE: The Domain account for the clustered Oracle Business Intelligence Server and cluster controllers must have the Log on as a service right explicitly granted on each computer running the servers and cluster controllers. However, the Oracle BI Presentation Services can be run under a different account or domain, as long as the Web Server login has the appropriate privileges.

Use one of the following procedures to perform the task of granting the Log on as a service right, based on your specific Windows platform.

Granting the Oracle BI Log On as a Service Right (Windows 2000)

To grant the Log on as a service right under a Windows 2000 platform, use the following procedure.

To grant the Log on as a service right under Windows 2000

1. Choose Administrative Tools from the Control Panel and click Local Security Policy.

   The Local Security Settings window appears.

2. Expand the Local Policies tree in the left pane and double-click User Rights Assignment.
3. Locate the Log on as a service right, double-click it to open the Local Security Policy Setting window, and click Add.

   The Select Users or Groups window appears.

4. From the Look In drop-down list, select the domain that the account is in.
5. Locate the account in the Name list, highlight it, and click Add.

   Click OK.

6. Click OK to return to the Local Security Settings window.

   The Log on as a service right has been added to the account. Close this window.

Granting the Oracle BI Log On as a Service Right (Windows XP)

To grant the Log on as a service right under a Windows XP platform, use the following procedure.

To grant the Log on as a service right under Windows XP

1. Choose Administrative Tools from the Control Panel and click Local Security Policy.
2. In the Local Security Settings window, expand the Local Policies tree in the left pane and double-click User Rights Assignment.
3. Locate the Log on as a service right, double-click it to open the Log on as a service Properties window, and click Add User or Group.
4. In the Select Users or Groups window, in the field From this location, make sure the correct domain for the account is displayed.

   If it is not, click Locations and select the domain that the account is in. Locate the account in the Name list, highlight it, and click OK.
5. Type the name of the account in the field labeled Enter the object names to selec*t.*

   Click the Check Names button to verify that it is correct. Click OK.
6. Click OK again to return to the Local Security Settings window.

   The Log On as a service right has been added to the account. Close this window.

B Using the CryptoTools Utility

Oracle BI Presentation Services Credential Store supports storage of credentials in an XML file of proprietary format. The CryptoTools utility that is provided with Oracle Business Intelligence is a general purpose utility for the manipulation of the credential store XML file. This utility is located in the OracleBI_HOME\web\bin directory on Windows and OracleBI_HOME/web/bin directory on Linux or UNIX.

CryptoTools is used to directly edit the credential store XML file, or to generate encrypted XML fragments that can be embedded manually into the credential store file. This appendix describes the syntax for this tool, and its supported sub-commands.

For more information on the BI Presentation Services Credential Store and the credential store XML file, see Chapter 5, “Oracle BI Presentation Services Credential Store”.

Syntax

The CryptoTools’ syntax involves specifying a sub-command with a set of sub-command specific options:

```
cryptotools <sub-command> [sub-command options]
```

Specifying the help option displays help about the sub-commands:

```
cryptotools -help
```

Credstore Sub-Commands

The supported sub-command for CryptoTools is credentialstore (or credstore).

This sub-command is used to manipulate the credential store XML file. It can be used to add, remove and overwrite credentials from the XML file. It can also be used to create a new credential store file.

NOTE: The use of other CryptoTools sub-commands is not supported.

Immediately following the credstore sub-command, an option must be specified. The sub-command options are shown in the following table.

NOTE: The parameters may be passed on the command line. The user is prompted for any parameters not supplied on the command line.

Option	Description	Parameter	Param Description
new	Creates a new credential store file and adds a username/password credential to it.	outFile	The path to the new credential store file.
		alias	The alias of the new credential.
		username	The username for the new credential.
		password	The password for the new credential.
		passphrase	The passphrase used to encrypt the password.
		noenc	If this option is specified, password encryption is disabled. If a passphrase is supplied on the command line, this parameter is ignored and encryption is enabled.
		writePassphrase	Whether or not to embed the encryption passphrase into the final XML. For security reasons, it is recommended that the passphrase not be written in the file, since unauthorized access to the credential store file will reveal the user password. Instead, the encryption passphrase should be supplied in the Oracle BI configuration files. For more information, see chapter Oracle Business Intelligence BI Presentation Services Credential Store in this guide.
add	Adds a new username and encrypted password credential to a credential store file. The parameters for this command are identical to those for the new command with the additions shown. All required parameters are supplied on the command line.	inFile	The path and the file name of an existing credential store file. If no such file exists, the tool starts with an empty credential store, thus making this command equivalent to the new command.
		outFile	The path where the resultant store is written. If this is not supplied, the tool defaults to the path and file supplied with the inFile parameter.

Option	Description	Parameter	Param Description
addx509	Adds a new X.509 credential to a credential store file.	inFile	The path and file name of an existing credential store file. If no such file exists, the tool starts with an empty credential store.
		outFile	The path where the resultant store is written. If this is not supplied, the tool defaults to the path and file supplied with the inFile parameter.
		alias	The alias of the new credential.
		certfile	The path to a PEM or ASN1 encoded certificate file.
		certencoding	The encoding for the certificate file. If not supplied, no explicit encoding is stored in the credential store, and an encoding is guessed at runtime based on the filename.
		keyfile	The path to a PEM or ASN1 encoded (possibly encrypted) private key file.
		keyencoding	The encoding for the key file. If not supplied, no explicit encoding is stored in the credential store, and an encoding is guessed at runtime based on the filename.
		keypass	If the private key file is encrypted and the user wants to embed the passphrase for decryption right into the credential store XML file, then this parameter is required. For security reasons, it is recommended that this not be done since unauthorized access to the credential store file will reveal the private key. Instead, supply the encryption passphrase in the Oracle BI configuration files. For more information, see Chapter 5, "Oracle BI Presentation Services Credential Store".
remove	Removes a credential identified by the alias from an existing file.	inFile	The path and file name of an existing credential store file.

Option	Description	Parameter	Param Description
alias	The alias of the credential to be removed.		
list	Lists all entries in an existing credential store file.	inFile	The path to an existing credential store file.

C Supporting Files For Provisioning Using Directory Integration Platform (DIP)

This appendix provides the contents of the following files, which were used to set up provisioning for Oracle BI using Directory Integration Platform (DIP):

- Usapp.dat
- appriv.dat
- ldap_ntfy_30.sql
- ldap_ntfy_30.pks
- ldap_ntfy_30.pkb

This provisioning setup is described in the topic "Integrating Oracle Internet Directory With Oracle Business Intelligence" on page 149.

NOTE: The contents of these files are samples only. Before using, review these files and modify based on your environment.

Reference:

$host = Hostname of the machine where OID and DIP are installed

$port = 389 for non ssl connection

$passwd = Password for superuser orcladmin

$db_host = host name of the Oracle database server where the provisioning is to be done

$connect = Oracle SID (directory database connect string, or the net service name in tnsnames.ora)

$db_userid = db user

$db_userpassword = password for db user

ADD(*) = attributes you want provisioned through DIP when a new entity is added in OID. Use asterisk (*) to provision all attributes, or provide a list of required attributes.

Contents of Uspapp.dat:

```
dn: cn=DUMMYAPP,cn=Products,cn=OracleContext

changetype: add

objectclass: orclcontainer

dn: orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext

changetype: add

objectclass: orclApplicationEntity
```

```
orclapplicationcommonname: DummyappOnline

orclappfullname: Oracle Dummyapp Online

userpassword: welcome123

description: This is a test Appliction instance

protocolInformation: Dummy Information

orclVersion: 1.0

orclaci: access to entry by group="cn=odisgroup,cn=DIPAdmins,cn=Directory Integration
Platform,cn=Products,cn=Oraclecontext" (browse,pr

oxy) by group="cn=User Provisioning Admins,cn=Groups,cn=OracleContext"
(add,delete,browse)

orclaci: access to attr=(*) by group="cn=User Provisioning
Admins,cn=Groups,cn=OracleContext" (search,read,write,compare)

dn: cn=User
Properties,orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleC
ontext

changetype: add

objectclass: orclcontainer
```

Contents of appriv.dat:

```
dn: cn=OracleDASCreateUser,cn=Groups,cn=OracleContext

changetype: modify

add: uniquemember

uniquemember:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext

dn: cn=OracleDASDeleteUser,cn=Groups,cn=OracleContext

changetype: modify

add: uniquemember

uniquemember:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext

dn: cn=OracleDASEditUser,cn=Groups,cn=OracleContext

changetype: modify

add: uniquemember

uniquemember:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext
```

```
dn: cn=OracleDASCreateGroup,cn=Groups,cn=OracleContext

changetype: modify

add: uniquemember

uniquemember:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext

dn: cn=OracleDASDeleteGroup,cn=Groups,cn=OracleContext

changetype: modify

add: uniquemember

uniquemember:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext

dn: cn=OracleDASEditGroup,cn=Groups,cn=OracleContext

changetype: modify

add: uniquemember

uniquemember:
orclApplicationCommonName=DummyappOnline,cn=DUMMYAPP,cn=Products,cn=OracleContext
```

Contents of ldap_ntfy_30.sql:

```
Rem
Rem $Header: ldap_ntfy_30.sql 28-mar-2005.01:29:44 sasrivas Exp $
Rem

Rem ldap_ntfy_30.sql

Rem
Rem Copyright (c) 2004, 2005, Oracle. All rights reserved.
Rem

Rem NAME
Rem ldap_ntfy_30.sql - Provisioning -- App tables for out/in bound data.

Rem
Rem DESCRIPTION
Rem <short description of component this file declares/defines

Rem
Rem NOTES
Rem <other useful comments, qualifications, etc.

Rem
Rem MODIFIED   (MM/DD/YY)
Rem sasrivas    03/28/05 - Merge from 101201 to Main Linux
Rem sasrivas    02/17/05 - sasrivas_ocs_sync_tx2
Rem vasokkum    10/05/04 - Change ADT Names to have v3.
Rem sasrivas    06/28/04 - sasrivas_ocsprovtest
Rem sasrivas    06/23/04 - Moved the file from MAIN to OCS10.1.2
Rem snamudur    02/11/04 - snamudur_userprov1
Rem snamudur    02/11/04 - Created
Rem

SET ECHO ON

SET FEEDBACK 1

SET NUMWIDTH 10

SET LINESIZE 80

SET TRIMSPOOL ON

SET TAB OFF

SET PAGESIZE 100

DROP TABLE APP_TO_OID_EVENT_MASTER ;

DROP TABLE APP_TO_OID_EVENT_DETAILS ;

DROP TABLE APP_TO_OID_EVENT_STATUS;

DROP TABLE OID_TO_APP_RECEIVED_EVENTS ;

DROP TABLE OID_TO_APP_EVENTS_COUNT ;

DROP TABLE OID_PROV_PARAMS_STATUS ;

REM This is the Table For the Canned Data To be Propagated From APP to OID

REM Used only for INBOUND profiles

CREATE TABLE APP_TO_OID_EVENT_MASTER (

        EVENT_ID      NUMBER,           -- Numeric Event Identifier

        EVENT_TYPE    VARCHAR2(32),     -- Type of Event like SUBSCRIPTION_ADD

        EVENT_SRC     VARCHAR2(1024),   -- Modifier's Name. Used only from OID

        OBJECT_NAME   VARCHAR2(1024),   -- Used only in Events from OID

        OBJECT_GUID   VARCHAR2(32),     -- Unique Global Identifier

        OBJECT_DN     VARCHAR2(1024),   -- Used only in Events from OID

        OBJECT_TYPE   VARCHAR2(32),     -- Type Of Object. App specific

        PROFILE_ID    VARCHAR2(256),    -- Used only in Events from OID

        DESCRIPTION   VARCHAR2(256))    -- Description

/

CREATE UNIQUE INDEX APP_TO_OID_EVENT_MASTER_I ON
```

```
     APP_TO_OID_EVENT_MASTER (EVENT_ID)
/
REM This is the Table For the Canned Data To be Propagated From APP to OID
REM This contains the event details.Used only for INBOUND profiles
CREATE TABLE APP_TO_OID_EVENT_DETAILS (
        EVENT_ID      NUMBER,        -- Foreign Key
        ATTR_NAME     VARCHAR2(64),
        ATTR_TYPE     NUMBER,
        ATTR_VALUE    VARCHAR2(4000),
        ATTR_MOD_OP   NUMBER)
/
REM This is the Table used to put the STATUS events received from DIP
REM Used only for INBOUND profiles
CREATE TABLE APP_TO_OID_EVENT_STATUS (
       EVENT_ID    NUMBER,
       EVENT_STATUS  VARCHAR2(32),
       EVENT_STATUS_MSG VARCHAR2(1024),
       ORCLGUID  VARCHAR(32))
/
REM This is the Table used to put the EVENTS received from DIP.
REM Used only for OUTBOUND profiles
CREATE TABLE OID_TO_APP_RECEIVED_EVENTS(
       EVENT_TYPE VARCHAR2(32),
       EVENT_ID NUMBER,
       EVENT_SRC VARCHAR2(256),
       EVENT_TIME VARCHAR2(32),
       OBJECT_NAME VARCHAR2(256),
       OBJECT_TYPE VARCHAR2(32),
       OBJECT_GUID VARCHAR2(32),
       OBJECT_DN VARCHAR2(256),
       PROFILE_ID VARCHAR2(256),
       ATTR_NAME VARCHAR2(64),
       ATTR_VALUE VARCHAR2(256),
       ATTR_VALUE_LEN VARCHAR2(32),
       ATTR_MOD_OP VARCHAR2(32))
/
CREATE TABLE OID_TO_APP_EVENTS_COUNT(
       ID            NUMBER,
       EVENT_COUNT   NUMBER
)
/
CREATE TABLE OID_PROV_PARAMS_STATUS(
       LAST_EVENT_PROCESSED_BY_OID NUMBER, --Used to keep track of INBOUND sync
       LAST_EVENT_PROCESSED_BY_APP NUMBER, --Used to keep track of OUTBOUND sync
       EVENT_BATCH_SIZE NUMBER, -- These packages uses this to simulate bulk
                                -- events for INBOUND profiles.Should match
                                -- the value in profile
       ENCRYPTION_KEY VARCHAR2(32)) -- Used to encrypt/decrypt data.
                                    -- Should match with what is in the profile.
/
INSERT INTO OID_PROV_PARAMS_STATUS VALUES (0,0,'1','ABCDDCBA')
/
DROP SEQUENCE PROV_TEST_DBG;
CREATE SEQUENCE PROV_TEST_DBG START WITH 1;
DROP TABLE  OID_PROV_DEBUG_LOG
/
CREATE TABLE OID_PROV_DEBUG_LOG(
       LOG_SEQ NUMBER,
       LOG_MSG VARCHAR2(1024))
/
```

```
exit ;
```

Contents of ldap_ntfy_30.pks:

```
-- The PL/SQL Interface Specification for interface version 3.0
-- The OID 9.0.2 used the interface version # 1.1
-- The OID 9.0.4 used the interface version # 2.0
-- The OID 9.0.? (OCSr3) will use the interface version # 3.0

DROP TYPE LDAP_EVENT_LIST_V3;
DROP TYPE LDAP_EVENT_STATUS_LIST_V3;

DROP TYPE LDAP_EVENT_V3;
DROP TYPE LDAP_EVENT_STATUS_V3;

DROP TYPE LDAP_ATTR_LIST_V3;
DROP TYPE LDAP_ATTR_V3;

DROP TYPE LDAP_ATTR_VALUE_LIST_V3;
DROP TYPE LDAP_ATTR_VALUE_V3;

CREATE TYPE LDAP_ATTR_VALUE_V3 AS OBJECT (
     attr_value       VARCHAR2(4000),
     attr_bvalue      RAW(2048),
     attr_value_len   INTEGER
);
/

GRANT EXECUTE ON LDAP_ATTR_VALUE_V3 to public;

CREATE TYPE LDAP_ATTR_VALUE_LIST_V3 AS TABLE OF LDAP_ATTR_VALUE_V3;
```

```
/

GRANT EXECUTE ON LDAP_ATTR_VALUE_LIST_V3 to public;

CREATE TYPE LDAP_ATTR_V3 AS OBJECT (
     attr_name        VARCHAR2(256),
     attr_type        INTEGER,
     attr_mod_op      INTEGER,
     attr_values      LDAP_ATTR_VALUE_LIST_V3
);
/

GRANT EXECUTE ON LDAP_ATTR_V3 to public;

CREATE TYPE LDAP_ATTR_LIST_V3 AS TABLE OF LDAP_ATTR_V3;
/

GRANT EXECUTE ON LDAP_ATTR_LIST_V3 to public;

CREATE TYPE LDAP_EVENT_V3 AS OBJECT (
          event_type  VARCHAR2(32),
          event_id    VARCHAR2(32),
          event_src   VARCHAR2(1024),
          event_time  VARCHAR2(32),
          object_name VARCHAR2(1024),
          object_type VARCHAR2(32),
          object_guid VARCHAR2(32),
          object_dn   VARCHAR2(1024),
          profile_id  VARCHAR2(1024),
          attr_list   LDAP_ATTR_LIST_V3 ) ;
```

```
/

GRANT EXECUTE ON LDAP_EVENT_V3 to public;

CREATE TYPE LDAP_EVENT_LIST_V3 AS TABLE OF LDAP_EVENT_V3;
/
GRANT EXECUTE ON LDAP_EVENT_LIST_V3 to public;

CREATE TYPE LDAP_EVENT_STATUS_V3 AS OBJECT (
          event_id    VARCHAR2(32),
          event_status  VARCHAR2(32),
          event_status_msg  VARCHAR2(2048),
          orclguid  VARCHAR(32)) ;
/

GRANT EXECUTE ON LDAP_EVENT_STATUS_V3 to public;

CREATE TYPE LDAP_EVENT_STATUS_LIST_V3 AS TABLE OF LDAP_EVENT_STATUS_V3;
/
GRANT EXECUTE ON LDAP_EVENT_STATUS_LIST_V3 to public;

-- A Test Package. The name of the package is configurable. The the procedure
-- definitions are interface spec. compliant.

CREATE OR REPLACE PACKAGE PRASA_LDAP_NTFY AS

    -- The Event Types

    ENTRY_ADD               CONSTANT VARCHAR2(32) := 'ENTRY_ADD';
    ENTRY_DELETE            CONSTANT VARCHAR2(32) := 'ENTRY_DELETE';
```

```
ENTRY_MODIFY            CONSTANT VARCHAR2(32) := 'ENTRY_MODIFY';

USER_ADD                CONSTANT VARCHAR2(32) := 'USER_ADD';
USER_DELETE             CONSTANT VARCHAR2(32) := 'USER_DELETE';
USER_MODIFY             CONSTANT VARCHAR2(32) := 'USER_MODIFY';

IDENTITY_ADD            CONSTANT VARCHAR2(32) := 'IDENTITY_ADD';
IDENTITY_DELETE         CONSTANT VARCHAR2(32) := 'IDENTITY_DELETE';
IDENTITY_MODIFY         CONSTANT VARCHAR2(32) := 'IDENTITY_MODIFY';

GROUP_ADD               CONSTANT VARCHAR2(32) := 'GROUP_ADD';
GROUP_DELETE            CONSTANT VARCHAR2(32) := 'GROUP_DELETE';
GROUP_MODIFY            CONSTANT VARCHAR2(32) := 'GROUP_MODIFY';

SUBSCRIPTION_ADD        CONSTANT VARCHAR2(32) := 'SUBSCRIPTION_ADD';
SUBSCRIPTION_DELETE     CONSTANT VARCHAR2(32) := 'SUBSCRIPTION_DELETE';
SUBSCRIPTION_MODIFY     CONSTANT VARCHAR2(32) := 'SUBSCRIPTION_MODIFY';

SUBSCRIBER_ADD          CONSTANT VARCHAR2(32) := 'SUBSCRIBER_ADD';
SUBSCRIBER_DELETE       CONSTANT VARCHAR2(32) := 'SUBSCRIBER_DELETE';
SUBSCRIBER_MODIFY       CONSTANT VARCHAR2(32) := 'SUBSCRIBER_MODIFY';

-- The Attribute Modification Type

ATTR_TYPE_STRING             CONSTANT NUMBER  := 0;
ATTR_TYPE_BINARY             CONSTANT NUMBER  := 1;
ATTR_TYPE_ENCRYPTED_STRING   CONSTANT NUMBER  := 2;
ATTR_TYPE_DATE_STRING        CONSTANT NUMBER  := 3;

-- The Attribute Modification Type
```

```
MOD_ADD                   CONSTANT NUMBER  := 0;
MOD_DELETE                CONSTANT NUMBER  := 1;
MOD_REPLACE               CONSTANT NUMBER  := 2;

-- The Event dispostions constants

EVENT_SUCCESS             CONSTANT VARCHAR2(32)  := 'EVENT_SUCCESS';
EVENT_ERROR_IGNORE        CONSTANT VARCHAR2(32)  := 'EVENT_ERROR_IGNORE';
EVENT_ERROR_RESEND        CONSTANT VARCHAR2(32)  := 'EVENT_ERROR_RESEND';
EVENT_ERROR_FATAL         CONSTANT VARCHAR2(32)  := 'EVENT_ERROR_FATAL';

-- The Actual Procedures

-- PUTOIDEVENTS : DIP Server invokes this API in the remote Database to
-- propagate a bunch of events to the application.
-- DIP server expects an event_status_list object in response as an OUT
-- parameter. If valid event status object is not sent back or it
-- indicates a RESEND, DIP server will keep resending from that event
-- indefnitely.

PROCEDURE PutOIDEvents (event_list         IN  LDAP_EVENT_LIST_V3,
                        event_status_list  OUT LDAP_EVENT_STATUS_LIST_V3);

-- GETAPPEVENT : DIP Server invokes this API in the remote Database.
-- It is up to the appliction to respond with an event list
-- Once DIP gets the event , it processes the event and sends the status
-- back.
-- The return value indicates whether any more event is returned or not.
```

```
    FUNCTION GetAppEvents (event_list OUT LDAP_EVENT_LIST_V3)
    RETURN NUMBER;

    -- Return CONSTANTS

    EVENT_FOUND             CONSTANT NUMBER  := 0;
    EVENT_NOT_FOUND         CONSTANT NUMBER  := 1403;

    PROCEDURE PutAppEventStatus (event_status_list IN LDAP_EVENT_STATUS_LIST_V3);

    FUNCTION Decrypt(inputStr IN VARCHAR2) RETURN VARCHAR2;

    PROCEDURE LogTrace(inputStr IN VARCHAR2) ;

END PRASA_LDAP_NTFY;
/
exit ;
```

Contents of ldap_ntfy_30.pkb:

```
CREATE OR REPLACE PACKAGE BODY PRASA_LDAP_NTFY AS
    sqlStatus NUMBER;
    LastEventProcByOID NUMBER;
    LastEventSentByAPP NUMBER;
--  ==========================================================================
    PROCEDURE LogTrace(inputStr IN VARCHAR2)
    IS
    BEGIN
      INSERT INTO OID_PROV_DEBUG_LOG VALUES (PROV_TEST_DBG.NEXTVAL,inputStr);
      DBMS_OUTPUT.PUT_LINE(inputStr);
    END ;
--  ==========================================================================
    FUNCTION Encrypt(inputStr IN VARCHAR2)
    RETURN VARCHAR2
    IS
      key VARCHAR2(8);
      inputStrCopy VARCHAR2(128);
      inputStrLen NUMBER;
      paddingRqd NUMBER;
      outputStr VARCHAR2(256);
    BEGIN
      SELECT ENCRYPTION_KEY INTO key FROM OID_PROV_PARAMS_STATUS;
      inputStrLen := LENGTH(inputStr);
      paddingRqd := 8 - (MOD(inputStrLen,8));
      IF (paddingRqd = 0 OR paddingRqd = 8)
      THEN
        paddingRqd := 0;
        inputStrCopy := inputStr;
      ELSE
        inputStrCopy := RPAD(inputStr,inputStrLen+paddingRqd);
      END IF;
      LogTrace('Padding :' || paddingRqd);
      outputStr :=
RAWTOHEX(UTL_RAW.CAST_TO_RAW(DBMS_OBFUSCATION_TOOLKIT.DESENCRYPT(INPUT_STRING = inputS
      LogTrace('Encrypted Value :(' || outputStr || ')' );
      RETURN (outputStr);
    END ;
--  ==========================================================================
    FUNCTION Decrypt(inputStr IN VARCHAR2)
    RETURN VARCHAR2
    IS
      key VARCHAR2(8);
      outputStr VARCHAR2(256);
    BEGIN
      SELECT ENCRYPTION_KEY INTO key FROM OID_PROV_PARAMS_STATUS;
      outputStr := DBMS_OBFUSCATION_TOOLKIT.DESDECRYPT(INPUT_STRING =
UTL_RAW.CAST_TO_RAW(HEXTORAW(inputS
      LogTrace('Decrypted Value :(' || outputStr || ')' );
      RETURN (outputStr);
    END ;
--  ==========================================================================
    FUNCTION GetAppEvents (event_list OUT LDAP_EVENT_LIST_V3)
    RETURN NUMBER
    IS
      attr_list LDAP_ATTR_LIST_V3;
      attr_value_list LDAP_ATTR_VALUE_LIST_V3;
      status  NUMBER;
      lstproc VARCHAR2(32);
      evt_idx NUMBER := 0;
      batchSize NUMBER := 0;
      i NUMBER;
      j NUMBER;
      CURSOR c_EVTS(StartNum IN NUMBER) IS
               SELECT
                 EVENT_ID,
                 EVENT_TYPE,
                 EVENT_SRC,
                 OBJECT_NAME,
                 OBJECT_TYPE,
                 OBJECT_GUID,
                 OBJECT_DN,

                 PROFILE_ID,

                 DESCRIPTION

               FROM

                  APP_TO_OID_EVENT_MASTER

               WHERE

                  EVENT_ID  StartNum

               ORDER BY EVENT_ID;

      CURSOR c_EVT_DTLS (EVENTID IN NUMBER) IS

               SELECT

                 ATTR_NAME,

                 ATTR_TYPE,

                 ATTR_VALUE,

                 ATTR_MOD_OP

               FROM

                  APP_TO_OID_EVENT_DETAILS

               WHERE

                 EVENT_ID = EVENTID

               ORDER BY ATTR_NAME,ATTR_MOD_OP;

      v_EVENT_ID      APP_TO_OID_EVENT_MASTER.EVENT_ID%TYPE;

      v_EVENT_TYPE    APP_TO_OID_EVENT_MASTER.EVENT_TYPE%TYPE;

      v_EVENT_SRC     APP_TO_OID_EVENT_MASTER.EVENT_SRC%TYPE;

      v_OBJECT_NAME   APP_TO_OID_EVENT_MASTER.OBJECT_NAME%TYPE;

      v_OBJECT_TYPE   APP_TO_OID_EVENT_MASTER.OBJECT_NAME%TYPE;

      v_OBJECT_GUID   APP_TO_OID_EVENT_MASTER.OBJECT_TYPE%TYPE;

      v_OBJECT_DN     APP_TO_OID_EVENT_MASTER.OBJECT_DN%TYPE;

      v_PROFILE_ID    APP_TO_OID_EVENT_MASTER.PROFILE_ID%TYPE;

      v_DESCRIPTION   APP_TO_OID_EVENT_MASTER.DESCRIPTION%TYPE;

      v_ATTR_NAME     APP_TO_OID_EVENT_DETAILS.ATTR_NAME%TYPE;

      v_ATTR_TYPE     APP_TO_OID_EVENT_DETAILS.ATTR_TYPE%TYPE;
```

```
   v_ATTR_VALUE    APP_TO_OID_EVENT_DETAILS.ATTR_VALUE%TYPE;
   v_ATTR_MOD_OP   APP_TO_OID_EVENT_DETAILS.ATTR_MOD_OP%TYPE;
   curAttrName APP_TO_OID_EVENT_DETAILS.ATTR_NAME%TYPE := '%APPNAME%';
   curAttrType APP_TO_OID_EVENT_DETAILS.ATTR_TYPE%TYPE := 0 ;
   curAttrModOp APP_TO_OID_EVENT_DETAILS.ATTR_MOD_OP%TYPE := 0;
   attr_idx NUMBER := 1;
   attr_value_idx NUMBER := 1;
   saveStartID NUMBER ;
   startEventID NUMBER;
   v_ATTR_BVALUE  VARCHAR2(2048);
BEGIN
   LogTrace('New Call to GetApp Events....');
   attr_list := LDAP_ATTR_LIST_V3 ();
   attr_value_list := LDAP_ATTR_VALUE_LIST_V3 ();
   event_list := LDAP_EVENT_LIST_V3();
   SELECT NVL(LAST_EVENT_PROCESSED_BY_OID,0)
     INTO LastEventProcByOID
   FROM OID_PROV_PARAMS_STATUS ;
   saveStartID := LastEventProcByOID;
   SELECT NVL(EVENT_BATCH_SIZE,1)
     INTO batchSize
   FROM OID_PROV_PARAMS_STATUS ;
   LogTrace('Getting ' || batchSize || ' Events From ID : '
                                           || LastEventProcByOID);
   OPEN c_EVTS(LastEventProcByOID);
   LogTrace('Opened c_EVTS Cursor');
   evt_idx := 0;
   LOOP
     FETCH c_EVTS INTO v_EVENT_ID,v_EVENT_TYPE,v_EVENT_SRC,
                       v_OBJECT_NAME,v_OBJECT_TYPE,v_OBJECT_GUID,
                       v_OBJECT_DN,v_PROFILE_ID,v_DESCRIPTION;
     EXIT WHEN c_EVTS%NOTFOUND;
     LogTrace('Got Event ID: ' || v_EVENT_ID);
     evt_idx := evt_idx + 1 ;
     IF (evt_idx = 1)
     THEN
       startEventID := v_EVENT_ID;
     END IF;
     OPEN c_EVT_DTLS(v_EVENT_ID);
     LogTrace('Opened c_EVT_DTLS Cursor');
     curAttrName := 'FIRST_ATTR_TAG';
     attr_idx := 1;
     attr_value_idx := 1;
     LOOP
       FETCH c_EVT_DTLS INTO v_ATTR_NAME,v_ATTR_TYPE,
                         v_ATTR_VALUE,v_ATTR_MOD_OP ;
       EXIT WHEN c_EVT_DTLS%NOTFOUND;
       LogTrace('Fetched Attribute : ' || v_ATTR_NAME);
       v_ATTR_BVALUE := '';
       IF (v_ATTR_TYPE = PRASA_LDAP_NTFY.ATTR_TYPE_ENCRYPTED_STRING)
       THEN
         LogTrace('Encrypting Attribute: ' || v_ATTR_NAME);
         v_ATTR_BVALUE := Encrypt(v_ATTR_VALUE);
         v_ATTR_VALUE := Encrypt(v_ATTR_VALUE);
       END IF;
       IF (curAttrName != 'FIRST_ATTR_TAG')
       THEN
         -- Not the First row being fetched
         IF (curAttrName != v_ATTR_NAME)
         THEN
```

```
      -- New Attribute Encountered. Need to store the existing values
      -- for the previous attribute and start a new one.
      attr_list.extend ;
      attr_list(attr_idx) :=
               LDAP_ATTR_V3 (curAttrName,
                          curAttrType,
                          curAttrModOp,
                          attr_value_list);
      LogTrace('Added attr :' || curAttrName);
      attr_idx := attr_idx+1;
      attr_value_list.delete;
      attr_value_idx := 1;
    END IF;
  END IF;
  attr_value_list.extend ;
  attr_value_list(attr_value_idx) :=
               LDAP_ATTR_VALUE_V3 (v_ATTR_VALUE,
                                NULL,
                                LENGTH(v_ATTR_VALUE));
  attr_value_idx := attr_value_idx+1;
  LogTrace('Added Value to attr :' || v_ATTR_NAME);
  curAttrName := v_ATTR_NAME;
  curAttrType := v_ATTR_TYPE;
  curAttrModOp := v_ATTR_MOD_OP;
  LogTrace('curAttrName:' || v_ATTR_NAME);
END LOOP;
CLOSE c_EVT_DTLS;
LogTrace('Fetched All Attributes for Event : ' ||  v_EVENT_ID);
attr_list.extend ;
attr_list(attr_idx) :=
               LDAP_ATTR_V3 (curAttrName,
                          curAttrType,
                          curAttrModOp,
                          attr_value_list);
  LogTrace('Added Last attr :' || curAttrName);
  attr_idx := attr_idx+1; -- Not Required
  attr_value_list.delete;
  attr_value_idx := 1; -- Not Required
  event_list.extend;
  event_list(evt_idx) := LDAP_EVENT_V3( v_EVENT_TYPE,
                       v_EVENT_ID ,
                       v_EVENT_SRC,
                       v_EVENT_SRC,
                       NULL,
                       v_OBJECT_NAME,
                       v_OBJECT_TYPE,
                       v_OBJECT_GUID,
                       v_OBJECT_DN,
                       v_PROFILE_ID,
                       attr_list);
  LogTrace('Constructed Event ' || evt_idx);
  attr_list.delete;
  IF (evt_idx = batchSize)
  THEN
    LogTrace('Accumulated ' || evt_idx || ' Events.Exiting.');
    LastEventSentByAPP := v_EVENT_ID;
    EXIT;
  END IF;
END LOOP;
LogTrace('Out Of Outer Loop ' );
```

```
      CLOSE c_EVTS;
      IF (evt_idx = 0)
      THEN
        RETURN EVENT_NOT_FOUND;
      ELSE
        RETURN EVENT_FOUND;
      END IF;
--      EXCEPTION
--        WHEN OTHERS THEN
--          startEventID := saveStartID ;
--          LogTrace(' Error code    : ' || TO_CHAR(SQLCODE));
--          LogTrace(' Error Message : ' || SQLERRM);
--          LogTrace(' Exception encountered .. exiting');
    END ;
--  ===========================================================================
    PROCEDURE PutAppEventStatus (event_status_list IN LDAP_EVENT_STATUS_LIST_V3)
    IS
      idx NUMBER := 1;
    BEGIN
      IF (event_status_list IS NOT NULL)
      THEN
        FOR idx in event_status_list.first .. event_status_list.last LOOP
          INSERT INTO APP_TO_OID_EVENT_STATUS VALUES (
                                   event_status_list(idx).event_id,
                                   event_status_list(idx).event_status,
                                   event_status_list(idx).event_status_msg,
                                   event_status_list(idx).orclguid);
          UPDATE OID_PROV_PARAMS_STATUS SET LAST_EVENT_PROCESSED_BY_OID =
               TO_NUMBER(event_status_list(idx).event_id);
        END LOOP;
        LogTrace('Setting Event Success for :' ||
                                            event_status_list(idx).event_id);
      END IF;
    END;
--  ==========================================================================
    PROCEDURE PutOIDEvents ( event_list  IN  LDAP_EVENT_LIST_V3,
                             event_status_list OUT LDAP_EVENT_STATUS_LIST_V3)
    IS
       i NUMBER;
       j NUMBER;
       idx NUMBER := 1;
       evt_cnt_idx NUMBER := 1 ;
       evt_sts_idx NUMBER := 1;
       event_status LDAP_EVENT_STATUS_V3;
       temp1 NUMBER := 1;
       temp2 NUMBER := 1;
        prasa_uid varchar2(40) := NULL;
        prasa_gid varchar2(40) := NULL;
        prasa_groupname integer;
    BEGIN
      IF (event_list IS NOT NULL)
      THEN
        IF (event_list.count  0)
        THEN
          event_status_list := LDAP_EVENT_STATUS_LIST_V3();
          FOR idx in event_list.first .. event_list.last LOOP
           IF ((event_list(idx).attr_list IS NULL) OR (event_list(idx).attr_list.count
= 0))
            THEN
              INSERT INTO OID_TO_APP_RECEIVED_EVENTS(EVENT_TYPE,
                                                   EVENT_ID,

                                                   EVENT_SRC,

                                                   EVENT_TIME,

                                                   OBJECT_NAME,

                                                   OBJECT_TYPE,

                                                   OBJECT_GUID,

                                                   OBJECT_DN,

                                                   PROFILE_ID)

                                           VALUES (event_list(idx).event_type,

                                                   event_list(idx).event_id,

                                                   event_list(idx).event_src,

                                                   event_list(idx).event_time,

                                                   event_list(idx).object_name,

                                                   event_list(idx).object_type,

                                                   event_list(idx).object_guid,

                                                   event_list(idx).object_dn,

                                                   event_list(idx).profile_id);

            END IF;

            IF ( (event_list(idx).attr_list IS NOT NULL ) AND
(event_list(idx).attr_list.count  0 ) )

            THEN

             FOR i in event_list(idx).attr_list.first .. event_list(idx).attr_list.last

              LOOP

                IF ( (event_list(idx).attr_list(i).attr_values IS NOT NULL ) AND
(event_list(idx).attr_list(i).attr_values.count  0 )

)

                THEN

                 -- prasa_sn varchar2(30);

                 -- prasa_cn varchar2(30);

        --        prasa_uid varchar2(4000) := '';
```

```
                  FOR j  in event_list(idx).attr_list(i).attr_values.first ..
event_list(idx).attr_list(i).attr_values.last

                  LOOP

                   /* if (event_list(idx).attr_list(i).attr_name = 'cn')

                   then

                   prasa_cn := event_list(idx).attr_list(i).attr_values(j).attr_value;

                   elseif (event_list(idx).attr_list(i).attr_name = 'sn')

                   then

                   prasa_sn := event_list(idx).attr_list(i).attr_values(j).attr_value;

                   elseif (event_list(idx).attr_list(i).attr_name = 'uid')

                   then

                  prasa_uid := event_list(idx).attr_list(i).attr_values(j).attr_value;

                   end if; */

                   if (event_list(idx).attr_list(i).attr_name = 'uid')

                     then

                   prasa_uid := event_list(idx).attr_list(i).attr_values(j).attr_value;

                        insert into sa_user(logon) values
(event_list(idx).attr_list(i).attr_values(j).attr_value);

                        elsif (event_list(idx).attr_list(i).attr_name = 'employeetype')

                        then

                   prasa_gid := event_list(idx).attr_list(i).attr_values(j).attr_value;

                    select count(*) into prasa_groupname from sa_group where group_name
= prasa_gid;

                        if (prasa_groupname = 0)

                        then

                        insert into sa_group values
(event_list(idx).attr_list(i).attr_values(j).attr_value);

                        end if;

                   end if;

                if (prasa_uid is not null) and (prasa_gid is not null)

                then

              insert into sa_user_group(logon,group_name) values (prasa_uid,prasa_gid);
```

```
prasa_uid := NULL;
prasa_gid := NULL;
end if;
    INSERT INTO OID_TO_APP_RECEIVED_EVENTS(EVENT_TYPE,
                                     EVENT_ID,
                                     EVENT_SRC,
                                     EVENT_TIME,
                                     OBJECT_NAME,
                                     OBJECT_TYPE,
                                     OBJECT_GUID,
                                     OBJECT_DN,
                                     PROFILE_ID,
                                     ATTR_NAME,
                                     ATTR_VALUE,
                                     ATTR_VALUE_LEN,
                                     ATTR_MOD_OP)
    VALUES (event_list(idx).event_type,
    event_list(idx).event_id,
    event_list(idx).event_src,
    event_list(idx).event_time,
    event_list(idx).object_name,
    event_list(idx).object_type,
    event_list(idx).object_guid,
    event_list(idx).object_dn,
    event_list(idx).profile_id,
    event_list(idx).attr_list(i).attr_name,
    event_list(idx).attr_list(i).attr_values(j).attr_value,
    event_list(idx).attr_list(i).attr_values(j).attr_value_len,
    event_list(idx).attr_list(i).attr_mod_op);
  END LOOP;
```

```
                END IF ;
              END LOOP;
            END IF ;
            event_status_list.extend;
            event_status_list(evt_sts_idx) :=
                               LDAP_EVENT_STATUS_V3( event_list(idx).event_id,
                                                 'EVENT_SUCCESS',
                                                 'Successfully Processed..',
                                                 NULL);
            evt_sts_idx := evt_sts_idx+1;
            UPDATE OID_PROV_PARAMS_STATUS
                 SET LAST_EVENT_PROCESSED_BY_APP =
                                    TO_NUMBER(event_list(idx).event_id);
          END LOOP;
          temp1 := event_list.first;
          temp2 := event_list.count;
          INSERT INTO OID_TO_APP_EVENTS_COUNT(ID,EVENT_COUNT)
                         VALUES (temp1,temp2);
        END IF;
      END IF;
    END ;
--  ==========================================================================
END PRASA_LDAP_NTFY;
```

Index