Oracle® Endeca Information Discovery Studio

Studio Security Guide

Version 3.0.0 • May 2013

Copyright and disclaimer

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Table of Contents

Preface

Oracle® Endeca Information Discovery Studio is an enterprise data discovery platform for advanced, yet intuitive, exploration and analysis of complex and varied data.

Information is loaded from disparate source systems and stored in a faceted data model that dynamically supports changing data. This integrated and enriched data is made available for search, discovery, and analysis via interactive and configurable applications. Oracle Information Discovery Studio includes a Provisioning Service that allows you to upload data directly from spreadsheet files.

Oracle Endeca Information Discovery Studio enables an iterative “model-as-you-go” approach that simultaneously frees IT from the burdens of traditional data modeling and supports the broad exploration and analysis needs of business users.

About this guide

This guide explains how to install, configure, and use Oracle Endeca Information Discovery Studio securely.

Who should use this guide

This guide is intended for users responsible for system security, including system administrators, Studio administrators, and users who create and configure Studio applications.

Conventions used in this document

The following conventions are used in this document.

Typographic conventions

The following table describes the typographic conventions used in this document.

Typeface	Meaning
User Interface Elements	This formatting is used for graphical user interface elements such as pages, dialog boxes, buttons, and fields.
`Code Sample`	This formatting is used for sample code phrases within a paragraph.
Variable	This formatting is used for variable values. For variables within a code sample, the formatting is *`Variable`*.
`File Path`	This formatting is used for file names and paths.

Symbol conventions

The following table describes symbol conventions used in this document.

Symbol	Description	Example	Meaning
>	The right angle bracket, or greater-than sign, indicates menu item selections in a graphic user interface.	File > New > Project	From the File menu, choose New, then from the New submenu, choose Project.

Contacting Oracle Customer Support

Oracle Customer Support provides registered users with important information regarding Oracle software, implementation questions, product and solution help, as well as overall news and updates from Oracle.

You can contact Oracle Customer Support through Oracle's Support portal, My Oracle Support at *https://support.oracle.com*.

Chapter 1

About Security in Studio

Here is a high-level look at the available security features for Studio and the Provisioning Service, and sources for additional information.

About Studio security functions

Sources for additional information

About Studio security functions

Studio can support varying levels of security. For the most part, Studio security features follow basic industry standards.

Studio has some built-in security measures, including:

- Requiring all users to log in
- Encrypting user passwords in the Studio database
- Encrypting cookies

This guide discusses other optional security-related configuration, including:

- Using Secure Socket Layer (SSL) communication
- Using LDAP or SSO to control user access to Studio
- Restricting user access to Studio functions and applications
- Restricting access to the Studio database and file systems

Sources for additional information

In addition to this guide, the following documents contain additional information to help you secure your Studio implementation.

Guide	Description
Oracle Endeca Server Security Guide	Describes how to secure Oracle Endeca Server.
Security Guide for Integrator	Describes how to secure Integrator.

Guide	Description
Studio Installation Guide	Includes information on: • Changing the Studio database, including the recommended database privileges • Installing the Provisioning Service with SSL
Studio Administration and Customization Guide	Includes information on: • Configuring user access to Studio, including LDAP and single sign-on • Using SecurityManager to restrict access to application data
Studio User's Guide	Includes information on: • Using filters to restrict access to Studio application data • Using SSL to connect a Studio data source to an Endeca Server • Using SSL to connect Studio to the Provisioning Service • Configuring privileges for viewing and editing Studio applications

Chapter 2

Using SSL for Secure Communication

SSL can be used to secure communications among Studio, the Provisioning Service, and Endeca Server.

How SSL is used for communication within Oracle Endeca Information Discovery

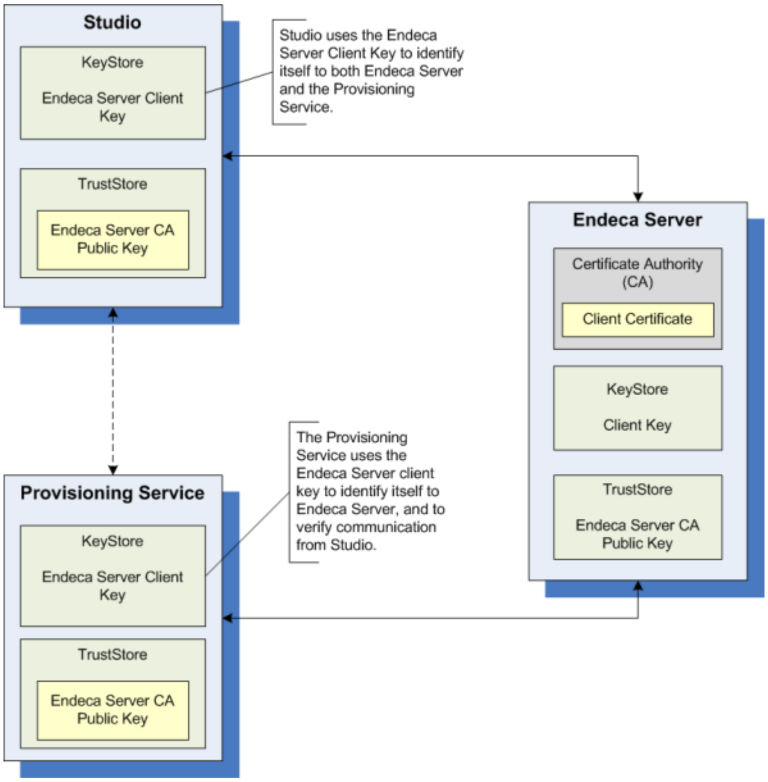
The SSL protocol helps protect the privacy and integrity of data while it is transferred across a network. For Studio, network communication occurs at multiple points. In addition to the connections with the application user's browser and with the LDAP server, there are other connections between Oracle Endeca components.

These communication links are encrypted with mutually-authenticated SSL by default. In each case, because authentication is mutual, both the host and the client must reference both a keystore and a truststore in order to get access to the certificates required to authenticate and be trusted.

The following diagram shows the SSL communication among the components of a secure Oracle Endeca Information Discovery implementation, including:

- Between Studio and the Provisioning Service
- Between Studio and Endeca Server

- Between the Provisioning Service and Endeca Server


Studio
KeyStore
Endeca Server Client Key
TrustStore
Endeca Server CA Public Key
Studio uses the Endeca Server Client Key to identify itself to both Endeca Server and the Provisioning Service.
Endeca Server
Certificate Authority (CA)
Client Certificate
KeyStore
Client Key
TrustStore
Endeca Server CA Public Key
The Provisioning Service uses the Endeca Server client key to identify itself to Endeca Server, and to verify communication from Studio.
Provisioning Service
KeyStore
Endeca Server Client Key
TrustStore
Endeca Server CA Public Key

Configuring SSL on the Studio application server

For increased security, Oracle recommends that you configure the Studio application server to use SSL.

For information on configuring SSL for WebLogic Server, see *Configuring SSL*.

For information on configuring SSL for Tomcat, see the *SSL Configuration How-To*.

Implementing SSL communication from the Provisioning Service

For the Provisioning Service (see the *Studio Installation Guide*), most of the configuration is handled by the domain template.

After using the Provisioning Service domain template (`eidProvisioningTemplate.jar`) to create the domain, to enable SSL communication:

1. Copy the following certificates from the Endeca Server `$DOMAIN_HOME/config/ssl` directory to the following directory of your Provisioning Service installation: `<`*`WebLogicInstallDirectory`*`>/user_projects/domains/oracle.eid-ps/eidProvisioningConfig`:
   - `endecaServerClientCert.ks`

- `endecaServerTrustStore.ks`

For Endeca Server, `$DOMAIN_HOME` is the full path to the Endeca Server WebLogic domain.

2. In `<`*`WebLogicInstallDirectory`*`>/user_projects/domains/oracle.eid-ps/eidProvisioningConfig/plan.xml`, verify that the following values have been set:

   (a) `endeca-server-ws-port` is set to `8202`.

   The Provisioning Service uses port 8202 for SSL-enabled communication, and 8201 for non-SSL-enabled communication.

   (b) `endeca-server-security-enabled` is set to `true`.

   (c) `transport-guarantee` is set to `CONFIDENTIAL`.

   (d) `protected-url-pattern` is set to `*/`.

3. After starting the domain:

   (a) Start a browser.

   (b) Log in to the Administration Console.

   (c) Replace and save the SSL passwords.

   For details about entering the SSL passwords on the Administration Console, see the *Configure Keystores* topic in the *Oracle WebLogic Server Administration Console Online Help*.

Connecting Studio to an SSL-enabled Provisioning Service

When you configure the connection from Studio to the Provisioning Service, you must also configure the SSL communication.

By default, the Provisioning Service has SSL enabled, and the configuration must include the `sslConfig` setting, which contains the following settings:

Setting	Description
`caFile`	The name of the truststore file for the SSL connection to the Provisioning Service. This is the truststore file from the secured Endeca Server configuration. For the default configuration, the file is `endcaServerTrustStore.ks`.
`caPassword`	The password for the truststore file for the SSL connection to the Provisioning Service. This is the password generated during the Endeca Server installation. Note that once you save the Provisioning Service configuration, the value of `caPassword` is masked as `************`. The value also is encrypted in the Studio database. When you edit the Provisioning Service connection, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.

Setting	Description
`certFile`	The name of the keystore file for the SSL connection to the Provisioning Service. This is the keystore file from the secured Endeca Server configuration. For the default configuration, the file is `endecaServerClientCert.ks`.
`certPassword`	The password for the keystore file for the SSL connection to the Provisioning Service. This is the password generated during the Endeca Server installation. Note that once you save the Provisioning Service configuration, the value of `certPassword` is masked as `************`. The value also is encrypted in the Studio database. When you edit the Provisioning Service connection, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.

For example:

```
"sslConfig": {
"caFile":"endecaServerTrustStore.ks",
"caPassword":"************",
"certFile":"endecaServerClientCert.ks",
"certPassword":"************"
}
```

The Studio **Control Panel** includes a **Provisioning Service** page you use to configure the connection.

To configure the Provisioning Service connection:

1. Stop Studio.
2. From the Endeca Server `$DOMAIN_HOME/config/ssl` directory, copy the following files:
   - `endecaServerClientCert.ks`
   - `endecaServerTrustStore.ks`

   For Endeca Server, `$DOMAIN_HOME` is the full path to the Endeca Server WebLogic domain.
3. Place the files into the `endeca-data-sources` directory.

   If Studio was installed using the Tomcat bundle, the directory is `endeca-portal/data/endeca-data-sources`.

   If Studio was installed on a standalone instance of Tomcat, without using the bundle, then you will need to create the `endeca-portal/data/endeca-data-sources` directory.

   If Studio was installed on WebLogic Server, then the directory is the `data/endeca-data-sources` directory in the Liferay Home directory. By default, the Liferay Home directory is `<`*`WebLogicInstallDirectory`*`>/user_projects/domains`.
4. Restart Studio.
5. From the administrator menu, select **Control Panel**.
6. In the **Control Panel** menu, click **Provisioning Service**.

7. On the **Provisioning Service** page, update the placeholder configuration with the connection information for your Provisioning Service.
8. Click **Save**.

Connecting a Studio data source to a secured Endeca Server

When you install Endeca Server, the default option is to use SSL to secure it. To connect to a secured Endeca Server, you copy the Endeca Server certificate files to Studio. In the Studio data source configuration, you then include the certificate file names and passwords.

Note that if you have already copied over the Endeca Server truststore and keystore files as part of configuring the connection to the Provisioning Service, then you can skip to step 5.

To allow Studio data sources to be a secured Endeca Server:

1. Stop Studio.
2. From the Endeca Server `$DOMAIN_HOME/config/ssl` directory, copy the following files:
   - `endecaServerClientCert.ks`
   - `endecaServerTrustStore.ks`

   For Endeca Server, `$DOMAIN_HOME` is the full path to the Endeca Server WebLogic domain.
3. Place the files into the `endeca-data-sources` directory.

   If Studio was installed using the Tomcat bundle, the directory is `endeca-portal/data/endeca-data-sources`.

   If Studio was installed on a standalone instance of Tomcat, without using the bundle, then you will need to create the `endeca-portal/data/endeca-data-sources` directory.

   If Studio was installed on WebLogic Server, then the directory is the `data\endeca-data-sources` directory in the Liferay Home directory. By default, the Liferay Home directory is `<`*`WebLogicInstallDirectory`*`>/user_projects/domains`.
4. Restart Studio.
5. From the **Data Sources** page of the Studio **Control Panel**, add the `sslConfig` setting to the appropriate data source definition. The `sslConfig` setting contains the following settings:

Setting	Description
`caFile`	The name of the truststore file. For the default secured Endeca Server configuration, the file is `endecaServerTrustStore.ks`.

Setting	Description
`caPassword`	The password for the truststore file. You need to obtain the password from whoever installed the Endeca Server and generated the certificates. Note that on the **Data Source Definition** dialog, once you save the data source, the value of `caPassword` is masked as `************`. The value also is encrypted in the Studio database. When you edit the data source, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.
`certFile`	The name of the keystore file. For the default secured Endeca Server configuration, the file is `endecaServerClientCert.ks`.
`certPassword`	The password for the keystore file. You need to obtain the password from whoever installed the Endeca Server and generated the certificates. Note that on the **Data Source Definition** dialog, once you save the data source, the value of `certPassword` is masked as `************`. The value also is encrypted in the Studio database. When you edit the data source, you must re-type the actual password value before saving. Otherwise, Studio uses the masking asterisks as the password value.

For example:

```
"sslConfig": {
"caFile":"endecaServerTrustStore.ks",
"caPassword":"************",
"certFile":"endecaServerClientCert.ks",
"certPassword":"************"
}
```

For details on using the **Data Sources** page to configure Studio data sources, see the *Studio User's Guide*.

Example of a data source connected to secured Endeca Server

The following data source connects to a secured Endeca Server.

```
{
 "server":"server01.lab.acme.com",
 "port":"7002",
 "dataDomainName":"acmeDB",
 "sslConfig": {
   "caFile":"endecaServerTrustStore.ks",
   "caPassword":"************",
   "certFile":"endecaServerClientCert.ks",
   "certPassword":"************"
 }
 "name":"High End Midwest Wines",
 "description":"Transactions for Midwest wines priced over 25 dollars",
 "baseFunctions": [
    {
      "class":"com.endeca.portal.data.functions.DataSourceFilter",
      "filterString":"Regions='Midwest'"
    },
    {
      "class":"com.endeca.portal.data.functions.RangeFilter",
      "property":"P_Price",
      "rangeOperator":"GTEQ",
      "rangeType":"Decimal",
      "value1": "25"
    },
    {
      "class":"com.endeca.portal.data.functions.RefinementFilter",
      "attributeValue":"1999",
      "attributeKey":"Year"
    }
 ]
}
```

Chapter 3

Controlling User Access to Studio

One aspect of securing Studio is controlling who can log in to Studio and the functions they have access to within Studio.

Using LDAP to manage Studio users

In any application that protects secure information, a key requirement is to clearly identify those users who should be granted access. In Studio, one way to do this is to use your existing LDAP system.

By having users log in with their existing LDAP credentials, instead of manually creating users within Studio, you have greater control over the access to Studio.

The LDAP integration can also be part of integration with a single sign-on (SSO) system, where users log in once and are then automatically logged in to all of the relevant applications, including Studio.

For details on how to integrating with an LDAP or SSO system to manage users in Studio, see the *Studio Administration and Customization Guide*.

Limiting the number of Studio administrators

In Studio, users with the Administrator user role have unlimited access to all Studio functions and applications. To reduce the possibility unwanted changes to your Studio configuration and applications, we recommend limiting the number of users who have the Administrator role.

By default, new users created in Studio have the Power User role.

When using LDAP to establish Studio users, you assign a user role to a Studio user group for the LDAP users. You should not assign the Administrator role to a user group.

For information on user roles and assigning roles to user groups, see the *Studio Administration and Customization Guide*.

Chapter 4

Controlling Access to Studio Applications and Data

In addition to restricting access to Studio as a whole, you should also restrict access to the applications and application data.

Restricting the data viewed by users

Controlling access to Studio applications

Restricting the data viewed by users

Studio provides filtering functions to ensure that users only see the data they should have access to.

Using a DataSourceFilter to restrict Studio data source data

Using role-based security to control access to Studio data sources

Using a DataSourceFilter to restrict Studio data source data

In Studio, data source definitions can include a `baseFunctions` setting to restrict the amount of data included in the data source, to help ensure that users only see the data they need.

The basic format for the `baseFunctions` setting is:

```
"baseFunctions": [
 {
  "class":"com.endeca.portal.data.functions.<class name>",
  "<property name>":"<property value>"
 }
]
```

Where:

- *<class name>* is the name of the filter class.
- *<property name>* is the name of a configuration property for the filter.
- *<property value>* is the value of the configuration property.

When a data source includes filters, all components that use that data source automatically use those filters.

While Studio supports multiple filters, the `DataSourceFilter` is the only permanent filter. When you use a `DataSourceFilter`, users do not know that the data has been filtered, and cannot remove the filter.

For additional details on configuring Studio data sources, including filters, see the *Studio User's Guide*.

DataSourceFilter syntax

A `DataSourceFilter` uses EQL to filter the data.

A `DataSourceFilter` is permanent. It is not added to the **Breadcrumbs** component and cannot be removed.

The properties for a `DataSourceFilter` are:

Property	Description
`filterString`	The EQL snippet containing the filter information. For a `DataFilter`, this would essentially be the content of a `WHERE` clause for an EQL statement. For details on EQL syntax, see the *Oracle Endeca Server EQL Guide*.

In the following example, the data is filtered to only show records from the Napa Valley region with a price lower than 40 dollars.

```
{
"class":"com.endeca.portal.data.functions.DataSourceFilter",
"filterString":"Regions='Napa Valley' and P_Price<40"
}
```

Example data source with a DataSourceFilter

The following example data source contains a `DataSourceFilter` to filter the to only include transaction records from the Midwest region.

```
{
 "server":"server01.lab.acme.com",
 "port":"7002",
 "dataDomainName":"acmeDB",
 "name":"Wine Transactions",
 "description":"Transaction data for the Midwest Region",
 "sslConfig": {
    "caFile":"endecaServerTrustStore.ks",
    "caPassword":"************",
    "certFile":"endecaServerClientCert.ks",
    "certPassword":"************"
  }
 "baseFunctions": [
  {
   "class":"com.endeca.portal.data.functions.DataSourceFilter",
   "filterString":"Regions='Midwest'"
  }
 ]
}
```

Using role-based security to control access to Studio data sources

By default, Studio provides data source role-based security.

You can configure a Studio application data source to control who can view the data based on user roles.

The data source settings related to role-based security are:

Setting	Description
`securityEnabled`	Whether to enable the security filters for queries to the data source. If set to `"true"`, then the data source uses the filters configured under `securityFilters`.
`securityFilters`	Defines all of the security filters to be used by the data source. **Note:** Data source filters are the only supported type of `securityFilter`.
`rolePermissionsMultiOr`	For users who have more than one security role, whether to use logical OR to combine the filters from each role into a single, combined security role filter. If set to `"true"`, then logical OR is used, and users have access to data that matches at least one of the filters for their security roles. If set to `"false"` (the default value), then logical AND is used, and users only have access to data that matches all of the filters associated with all of their security roles. Note that if logical OR is used, it is only used to combine filters from different security roles. The filters from each individual role are still applied using logical AND before they are combined with the filters from the other roles. Other non-security filters that apply to the entire data source, such as the filters under `baseFunctions`, are also applied using logical AND.
`rolePermissions`	Maps the user roles to the security filters. For each mapping, the format is: `"<role name>" : [<filter list>]` where: • *<role name>* is the name of the user role. • *<filter list>* is a comma-separated list of filter names to apply for the specified role. Each name is in quotes. For example, `["filter1", "filter2", "filter3"]`.
`inheritSecurity`	For a child data source, whether the data source should inherit the security filters configured in its ancestor data sources. If set to `"true"`, then the list of security filters to apply includes both: • All of the security filters for every ancestor data source • All of the security filters defined in the child data source

For example, in the data source shown below, users with the role "French Wine" can only see data from the Bordeaux and Burgundy regions, while users with the role "Austrian Wine" can only see data from the Austria, Burgenland, and Steiermark regions. Because `rolePermissionsMultiOr` is set to true, users who have both of these roles can view records from any of the five regions.

```
{
 "server":"server01.lab.acme.com",
 "port":"7002",
 "dataDomainName":"acmeDB",
 "name":"European Wines",
 "description":"Sales transactions for European wines",
 "sslConfig": {
    "caFile":"endecaServerTrustStore.ks",
    "caPassword":"************",
    "certFile":"endecaServerClientCert.ks",
    "certPassword":"************"
 }
 "securityEnabled":"true",
 "inheritSecurity":"true",
 "securityFilters": {
    "frenchFilter": {
        "class":"com.endeca.portal.data.functions.DataSourceFilter",
        "filterString":"Region='Bordeaux' OR Region='Burgundy'"
    },
    "austrianFilter": {
        "class":"com.endeca.portal.data.functions.DataSourceFilter",
        "filterString":"Region='Austria' OR Region='Burgenland' OR Region='Steiermark'"
    }
  },
 "rolePermissionsMultiOr":"true",
 "rolePermissions": {
    "French Wine": ["frenchFilter"],
    "Austrian Wine": ["austrianFilter"]
  }
}
```

For details on configuring Studio data sources, including role-based security filtering, see the *Studio User's Guide*.

If you require more than this default data source role-based security, you can create a custom Security Manager to filter Endeca Server data based on user profile details such as the user's role or group association.

For details on creating and configuring a Security Manager, see the *Studio Administration and Customization Guide*.

Controlling access to Studio applications

You can configure Studio applications to minimize the number of users who can view or configure each application.

Using private applications to manage access

Restricting who can configure applications

Using private applications to manage access

Studio applications can be either public or private. Public applications can be viewed by all logged-in users. Private applications can only be viewed by application members.

By default, all new Studio applications are private. For better access control, we recommend that you keep all applications private.

For details on application types and how to configure them, see the *Studio User's Guide*.

Restricting who can configure applications

Studio applications can only be configured by Studio administrators and by users assigned as application administrators for that application.

When a new application is created, only the application creator is assigned as an application administrator.

When configuring the application membership, most members should be application members only, able to view the application but not change its configuration. Only assign a user as an application administrator if they absolutely need to be able to configure the application.

Note that an application administrator does not need to be a Studio administrator (in other words, does not need to have the Administrator user role).

For details on configuring access to applications, see the *Studio User's Guide*.

Chapter 5

Controlling Access to the Studio Databases and File Systems

As part of a secure Studio configuration, you should make sure to control access to the Studio and Provisioning Service databases and file systems.

Restricting access to the Studio and Provisioning Service databases

Restricting access to the Studio and Provisioning Service file systems

Restricting access to the Studio and Provisioning Service databases

The Studio database stores the Studio data sources, applications, and configuration. The Provisioning Service also has an associated database. Access to these databases should be restricted to prevent corruption of the data.

By default:

- Studio uses a Hypersonic database. For a production environment, you must change to either an Oracle or a MySQL database.
- The Provisioning Service uses a Derby database. For a production environment, you must change to an Oracle database. You should then also disable the Derby database in WebLogic Server.

For both Studio and the Provisioning Service, only the account used to communicate with the respective databases should have write access.

The only exception to this is when you create a new database schema in order to change to a different database. Once the database is up and running, the write access should be removed.

See the *Studio Installation Guide* for information on configuring permissions before and after changing the Studio database.

For details on securing an Oracle database, see the *Oracle Database Security Guide*.

For details on securing a MySQL database, see the *MySQL Security Guide*.

Restricting access to the Studio and Provisioning Service file systems

For the application server, for additional security, you should restrict access to the file system.

In general, only the owner should have full access to create or update files on the system.

Also, for the Provisioning Service, the file upload directory is by default the temp directory for your system. The directory is configured using the `upload-file-directory` variable in `plan.xml`. You should change this setting to be a directory appropriate for your installation, and set the directory permissions to only allow the owner to have read and write access.

Index