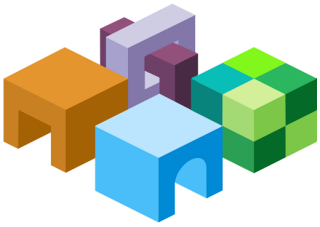


HYPERION

RELEASE 9.3.1

CONFIGURING ORACLE HYPERION WORKSPACE FOR KERBEROS AUTHENTICATION

ORACLE | Hyperion

Contents In Brief

About this Document

This document explains how to set up Oracle's Hyperion® Workspace to enable Single Sign-On (SSO) to a Kerberos realm using Windows Single Sign-on.

About Kerberos Single Sign-on

Kerberos SSO, also known as Windows Native Authentication, allows transparent Workspace access to Windows users. The credentials required for accessing Workspace are obtained from the Windows login credentials of the Hyperion user.

Kerberos is a trusted authentication service in which each Kerberos client trusts the identities of other Kerberos clients (users, network services, and so on) to be valid. Kerberos is centered around its Key Distribution Center (KDC), a database of its clients (users, computers, and services in the Kerberos realm). KDC maintains details of Kerberos clients and their private keys. Kerberos is based on the concept of tickets; data structures that wrap cryptographic keys and some other information. KDC distributes Kerberos tickets to authenticated clients. Computers on the network are configured to implicitly trust the KDC. Users gain access to network resources by presenting tickets with encrypted information from the KDC, which the server verifies. Because KDC is the only entity that knows every encryption key, it can securely verify the authenticity of its clients. Because each client trusts the KDC, the entire network is secure as long as the KDC is secure.

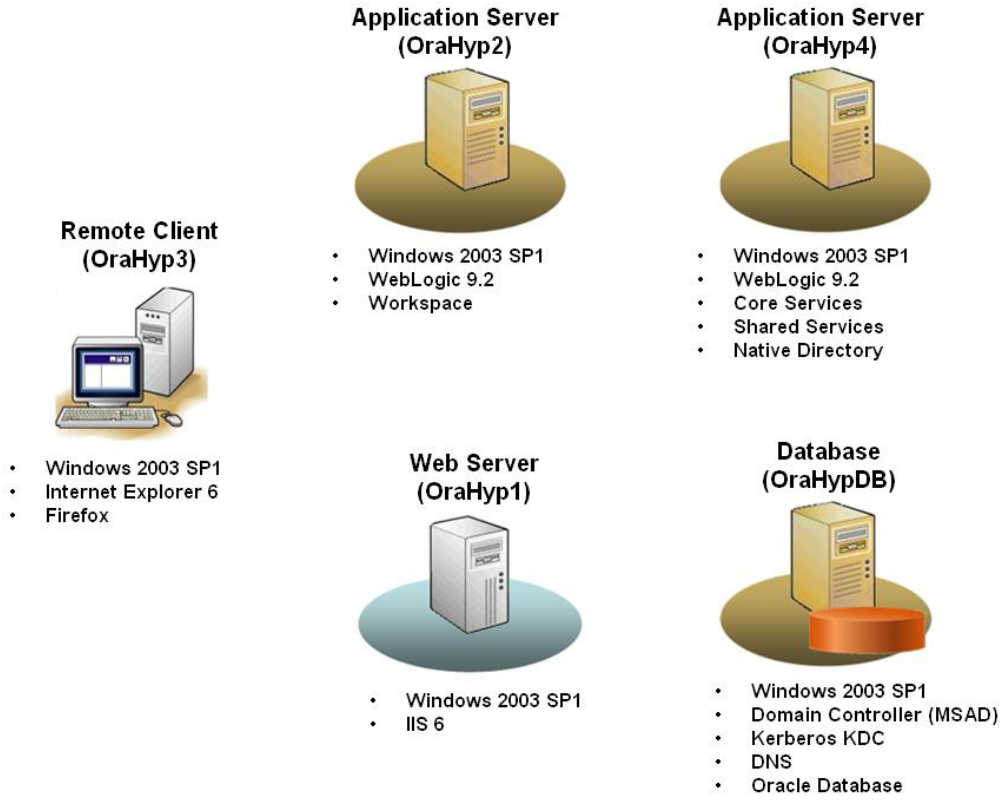
Browsers use the SPNEGO protocol to automatically pass the user's Kerberos credentials/tickets to a Kerberos-enabled server when the server request these credentials. The server decrypts the credentials and authenticates the user.

Technical Architecture and Prerequisites

This section presents a sample deployment architecture that is used to explore Hyperion products deployment in a Kerberos environment and deployment prerequisites.

Architecture

A sample deployment architecture and the components hosted by servers are indicated in the following illustration.



A supported browser that is capable of handling SPNEGO protocol should be used to access Workspace.

Prerequisites

- "Client Machines" on page 5
- "WebLogic Application Server" on page 5
- "Hyperion Products" on page 6

Kerberos

A fully functional Kerberos-enabled network environment is required to support Workspace. This document assumes the following conditions:

- Application servers (`OraHyp2.example.com`, and `OraHyp4.example.com`), Web server (`OraHyp1.example.com`), database server (`OraHypDB.example.com`), and client computers are members of a Kerberos realm (for example, `EXAMPLE.COM`).
- Make sure that fully qualified domain names (FQDN) are used across the configuration in all environments. Do not use `localhost`, IP address, or host name.
- Clients from which Workspace will be accessed are members of the Kerberos realm.
- Kerberos server and clients are in the same time zone and are synchronized to the same time and date.

- Connectivity among Kerberos server and clients is established using static IP.
- All connectivity issues have been resolved.
    - Connectivity was verified using name server lookup (`nslookup`) for forward and reverse DNS resolution.
    - The state and health of Active Directory domain controllers were verified using the Domain Controller Diagnostics Tool (`dcdiag.exe`).

Kerberos Conventions

The examples contained in this document assume the following Kerberos conventions:

- Service Principal Name (SPN) is created as `HTTP/`*`host_DNS_Name`*; for example, `HTTP/OraHypDB.example.com`.
- The Kerberos service class is HTTP.
- The Kerberos realm name is specified in all upper case; for example, `EXAMPLE.COM`.
- The Kerberos principal is specified as `HTTP/`*`host_DNS_Name@Kerberos_realm_name`*; for example `HTTP/OraHypDB.example.com@EXAMPLE.COM`.

Microsoft Active Directory

A Kerberos-enabled Active Directory is required to support the deployment of Workspace. Following are some of the steps involved in setting up Active Directory to support Kerberos. See Active Directory Documentation for detailed information.

- Install Active Directory on a server; for example, `OraHypDB`.
- Ensure that Kerberos KDC is running (preferably on port 88) on the server; for example, on `OraHypDB`.
- Promote the server hosting Active Directory to act as the domain controller.
- Install and configure DNS on the server that hosts Active Directory.
- Add all servers and clients to forward lookup zone.
- Add all servers and clients to reverse lookup.

You may need to use the following tools to work with the Kerberos server.

Kerbtray

Kerbtray is a graphic tool that displays ticket information for a computer running the Kerberos protocol.

Ksetup

This Microsoft utility configures clients to use UNIX-based Kerberos realm.

Ktpass

This Microsoft utility configures an Active Directory user for WebLogic server as Kerberos SPN and generates the keytab file that contains the SPN and the key.

Setspn

`Setspn` is a command-line tool that you can use to read, modify, and delete the SPN (which represents a running a service identity) property of an Active Directory service account.

Generally, you need not modify SPNs, because they are set up by a computer when it joins a domain and when services are installed on the computer. You do need to modify SPNs if they become stale. For instance, if the computer name is changed, the SPNs for installed services must be changed to match the new computer name.

Ldp

This Microsoft tool is an graphical LDAP client that allows users to perform operations such as connect, bind, search, modify, add, and delete against Active Directory. LDP is used to view objects stored in Active Directory along with their metadata, such as security descriptors and replication metadata.

LDIFDE

This Microsoft utility is used to import and export directory objects to Active Directory.

Client Machines

All servers and client machines should be configured to use the following:

- Kerberos server (for example; `OraHypDB`) as the primary DNS server.
- Domain controller on the Active Directory host; for example, `OraHypDB`.

Client machines have the following installations:

- A browser (Internet Explorer 6 or later, or Firefox 2 or later) that is capable of negotiating challenge and performing Kerberos authentication with the server.

  The browser must be configured for Kerberos authentication.
- Microsoft Kerberos utilities `kerbtray` to view tickets in the credential cache.

WebLogic Application Server

- WebLogic installations on the application server hosts `OraHyp2` (for Workspace) and `OraHyp4` (for Oracle's Hyperion® Shared Services).
- A keytab file that allows the WebLogic server to authenticate itself against the KDC. The keytab file contains its own principal and key derived from the password that is used by the Windows account that runs the WebLogic service.

Hyperion Products

Shared Services and Workspace are deployed to a WebLogic application server.

- A *HYPERION_HOME* on application server host (`OraHyp2`) where Workspaceis installed.
- A *HYPERION_HOME* on application server host (`OraHyp4`) where Shared Services, and Core Services are installed.
- A relational database (for example, on `OraHypDB`) to support Workspace and Shared Services.
- An Internet Information Services (IIS) installation on the Web server host (`OraHyp1`) that is used as the Workspace Web server.
- Shared Services, and Core Services are running. Check the availability, at these URLs:
  - Shared Services: `http://OraHyp4:58080/interop`
  - Workspace: `http://OraHyp2:19000/workspace`

Information Sources

- *Oracle Hyperion Security Administration Guide*
- *Oracle Hyperion Workspace Administrator's Guide*
- *Oracle Hyperion Installation Start Here*
- Microsoft Support (for the latest Active Directory and Kerberos tools and documentation)

Setup Procedures

Microsoft Active Directory

Complete these Active Directory procedures:

- "Create Service Principal Name and Keytab File" on page 10.
- "Creating the JAAS Configuration File" on page 12.

Note: In general, it is not a good practice to use the space character in user, group, role, and computer names. For example, names such as `test user`. and `my Computer` are not recommended. Also, do not use a hyphen (-) in user, group, role, or computer identifiers.

Opening the Microsoft Active Directory Console

The Active Directory Users and Computers window is used to perform most of the Active Directory configuration tasks detailed in this section.

➤ To open the Active Directory console:

1. Select **Start**, and then **Programs**,
2. Select **Administrative Tools**, and then **Active Directory Users and Computers**

   The Active Directory Users and Computers window opens.

Create the `wls_users` Group in Active Directory

Create a group called `wls_users` in Microsoft Active Directory.

➤ To create the `wls_users` group:

1. Open the Active Directory console. See "Opening the Microsoft Active Directory Console" on page 8.
2. Expand the node representing the Active Directory Domain Controller; for example, `OraHypDB.example.com`.
3. Right-click **Users**, then select **New**, and then **Group**.

   The New Object – Group window opens.
4. In **Group name** and **Group name (pre-Windows 2000)**, enter `wls_users`.
5. In **Group Scope**, select `Global`.
6. In **Group type**, select `Security`.
7. Click **OK**.

Create Active Directory Users

You must create the following active directory users:

- `bea_sso_ad` user as a member of the `wls_users` group
- *`WEBLOGIC_HOST`*`_WLS`; for example, `OraHyp2_WLS`, as a member of the `Users` group. This account represents the WebLogic Service user account in the Active Directory and will be mapped to Kerberos Service Principal.

See Microsoft Documentation for more information.

➤ To create Active Directory users:

1 Open the Active Directory console, if needed. See "Opening the Microsoft Active Directory Console" on page 8.

2 Expand the node representing the Active Directory Domain Controller; for example, `OraHypDB.oracle.com`.

3 Right-click **Users**, then select **New**, and then **User**.

The New Object – User window opens.

4 In **First name**, **Full name**, **User logon name**, and **User logon name (pre_windows 2000)**, enter `bea_sso_ad`.

5 Click **Next**.

6 In **Password** and **Confirm password**, enter a password for the user.

Caution! Do not select the `User must change password at next logon` option.

7 Click **Next**.

8 Click **Finish**.

9 Repeat the procedure to create the account for WebLogic server; for example, `OraHyp2_WLS`. To create `OraHyp2_WLS`, in step 4, you must use `OraHyp2_WLS` as the value in **First name** and **User logon name**.

Set Additional User Properties for WebLogic SSO User

You must set additional properties in the Active Directory account of WebLogic SSO user; for example, `bea_sso_ad`.

➤ To update user properties:

1 Open the Active Directory console, if necessary. See "Opening the Microsoft Active Directory Console" on page 8.

2 Expand the node representing the Active Directory Domain Controller; for example, `OraHypDB.example.com`

3 Click **Users**.

4 Right-click the user name; for example, `bea_sso_ad` in `wls_users` group, and then select `Properties`.

5 Select the **Account** tab.

6 Select the `Use DES encryption types for this account` option from **Account Options**.

Caution! Setting the encryption type can corrupt the password. If password corruption is detected, you should reset the Active Directory password of the user to the original password that you set while creating the user.

7 Verify that the `Do not require Kerberos preauthentication` option in **Account Options** is not selected.

8 Verify that **Password never expires** is selected.

9 Verify that **Account expires** is set to `Never`.

10 Click **OK**.

Trust Host Accounts for Delegation in AD

Ensure that all servers (`OraHyp1`, `OraHyp2`, and `OraHyp4`) are set up in the Active Directory with the `Trust computer for delegation` option enabled.

Set Additional User Properties for WebLogic Server User

You must update the Active Directory account of WebLogic server user; for example, `OraHyp2_WLS`, to set additional properties and to specify that it is trusted for delegation:

➤ To update a WebLogic server user account:

1 Open the Active Directory console, if needed. See "Opening the Microsoft Active Directory Console" on page 8.

2 Expand the node representing the Active Directory Domain Controller; for example, `OraHypDB.oracle.com`.

3 Select **Users**.

4 Right-click the WebLogic server user account; for example, `OraHyp2_WLS`, and select **Properties**.

5 Click **Account**.

6 Select the following options from **Account Options**:

- Use DES encryption types for this account
- Do not require Kerberos preauthentication
- Account is trusted for delegation

Caution! Setting the encryption type can corrupt the password. If password corruption is detected, reset the Active Directory password of the user to the original password that you set while creating the user.

7 Click **OK**.

Create Service Principal Name and Keytab File

Note: This procedure should be performed on the machine that hosts the WebLogic server; for example, `OraHyp2`.

The service principal name and keytab file are used to provide SSO between the browser and WebLogic SPNEGO filters. A keytab is a file that contains pairs of Kerberos principals and DES-encrypted keys derived from the Kerberos password. It is used to log into Kerberos without being challenged for a password. The keytab file is computer-independent. You can copy it from one computer to another.

Note: Oracle recommends that you use a global keytab file. Theoretically, you can create distinct keytab files at the WebLogic domain or WebLogic server level. See "Creating the JAAS Configuration File" on page 12 and "Adding Kerberos Java Options to WebLogic Startup Script" on page 16.

The following procedure requires that you use `ktpass` (see "`Ktpass`" on page 5) utility. Ensure that you have the latest version of `ktpass`, which can be obtained from Microsoft Support.

Using a tool such as `dcdiag`, verify that you can access the DNS server and the Active Directory from the server that hosts WebLogic server. You must correct any access errors before performing the following procedure.

Note: Ensure the SPN is created using the fully qualified domain name (FQDN) of the WebLogic server.

➤ To create the SPN and the keytab file:

1 **Using your windows credentials, log on to the machine that hosts the WebLogic server; for example, `OraHyp2`. This machine must be a Kerberos client.**

2 **Open a command prompt.**

3 **Note:** `-DesOnly` unsets the default setting of DesOnly encryption for the account. If you do not use the `-crypto` option, the message "KDC has no support for encryption type (14)" appears in the AdminS log.

See Microsoft documentation for a detailed procedures to execute the `ktpass` utility.

For instance, you can execute the following command to create `HTTP:/OraHyp2.example.com@EXAMPLE.COM` as the SPN and map it to `OraHyp2_WLS` user. The keytab file `bea.keytab` is generated and stored in `C:\bea`.

```
ktpass –princ HTTP/OraHypDB.example.com@EXAMPLE.COM –DesOnly –out C:\bea\bea.keytab
–pass myp@ssW0rd –mapuser OraHyp2_WLS -crypto DES-CBC-CRC
```

On executing the command, you should see a message similar to the following:

```
Targeting domain controller: OraHyp2.example.com
Using legacy password setting method
Successfully mapped HTTP/OraHyp2.example.com to OraHyp2_WLS.
WARNING: pType and account type do not match. This might cause problems.
Key created.
Output keytab to C:\bea\bea.keytab:
Keytab version: 0x502
```

```
keysize 67 HTTP/OraHyp2.example.com@EXAMPLE.COM ptype 0 (KRB5_NT_UNKNOWN)
vno 12 etype 0x1 (DES-CBC-CRC) keylength 8 (0x85b0eac10da4c7c8)
```

Creating the JAAS Configuration File

The JAAS login configuration file identifies the system properties and login modules that direct WebLogic server to allow Kerberos authentication to occur. You must create the JAAS configuration file*BEA_HOME*\krb5login.conf; for example, C:\bea\krb5login.conf.

➤ To create the JAAS configuration file:

1 **Open a text editor.**

2 **Enter directives such as the following. Be sure to modify the property values to suit your environment.**

Note: There is no line break within an entry. Make sure that the initiate and accept entries are specified without line breaks.

```
com.sun.security.jgss.initiate
{
com.sun.security.auth.module.Krb5LoginModule required principal="HTTP/
OraHyp2.example.com@EXAMPLE.COM" useKeyTab=true keyTab="C:\\bea\\bea.keytab"
storeKey=true debug=true;
};
com.sun.security.jgss.accept
{
com.sun.security.auth.module.Krb5LoginModule required principal="HTTP/
OraHyp2.example.com@EXAMPLE.COM" useKeyTab=true keyTab="C:\\bea\\bea.keytab"
storeKey=true debug=true;
};
```

3 **Save the file as** *BEA_HOME*\krb5login.conf.

WebLogic Procedures

- "Creating Custom Roles and Policies for the Workspace Web Application" on page 19

Note: Oracle recommends that you create a backup copy of your WebLogic configuration file before performing these operations. Back up these files: *HYPERION_HOME*`/deployments/WebLogic9/config/config.xml` and *HYPERION_HOME*`/deployments/WebLogic9/init-info/*.xml`. Also, back up any WebLogic file that you edit.

Creating WebLogic Domain for Workspace

You must create a WebLogic domain; for example, `ws_domain` for Workspace. See the *Hyperion Reporting and Analysis – System 9 Installation Guide* for instructions.

Creating Active Directory Authenticator

WebLogic security realm is a container for the users, groups, security policies, roles and providers that are used to protect WebLogic resources. You should configure a WebLogic realm and create an Active Directory authenticator.

➤ To create an Active Directory authenticator:

1 **Start WebLogic server and log in to the WebLogic Server Administration Console.**

2 **Open the Providers tab of the realm that you want to use to protect Oracle applications.**

a. In **Domain Structure**, click `Security Realms`.

Summary of Security Realms screen opens.

b. In the **Realms** list, click the default (active) realm; for example, `myrealm`, that you want to use to protect Oracle applications.

c. In the settings page, click the **Providers** tab.

3 **In Change Center, click Lock & Edit to activate buttons on the Providers tab.**

4 **Create an authentication provider.**

a. Under **Authentication Providers**, click **New**.

b. In **Name**, enter an authentication provider name; for example, `AD-AuthN`.

c. In **Type**, select `ActiveDirectoryAuthenticator`.

d. Click **OK**.

5 **Verify that the control flag of the authentication provider is set to OPTIONAL. The control flag specifies how the authentication fits into the login sequence.**

a. Under **Authentication Providers**, click the provider; for example, `AD-AuthN`, which you created in the preceding step.

The settings page for the selected provider opens.

b. In **Control Flag** on Common tab, select `OPTIONAL`.

c. Click **Save**.

d. Click **Provider Specific**.

e. In **Group Base DN**, enter the Distinguished Name (DN) of the group to which the `bea_sso_ad` user belongs. For example, if the `bea_sso_ad` user belongs to the `example.com/Users` group, enter `cn=Users,dc=example,dc=com`.

See "Create the `wls_users` Group in Active Directory" on page 8 and "Create Active Directory Users" on page 8.

f. In **Host**, enter the name or IP address of the Active Directory host machine; for example, `OraHypDB`.

g. In **Port**, verify that the Active Directory listen port is correctly set.

h. In **User Base DN**, enter the DN of the LDAP directory tree that contains users. For example, if users are defined in `example.com/Users`, enter `cn=Users,dc=example,dc=com`.

Note: User Base DN should contain all Hyperion users, service users, and the krbtgt user.

i. In **Principal**, enter the DN of the user (usually the Active Directory administrator) whose account WebLogic should use to connect to the Active Directory. For example, `cn=Administrator,cn=Users,dc=example,dc=com`

j. In **Credential** and **Confirm Credential**, enter the principal's password.

k. Click **Save**.

6 **Change the control flag of `DefaultAuthenticator`.**

a. Open the Providers tab of the active realm. See step 2 on page 13 for instructions.

b. In **Authentication Providers**, click `DefaultAuthenticator`.

c. In **Control Flag** on Common tab, select `OPTIONAL`.

d. Click **Save**.

7 **In Change Center, click Activate Changes.**

8 **Restart WebLogic.**

Verifying Authentication Provider

Before proceeding, verify that the authenticator you created in the preceding section can connect to the Active Directory to access user and group information.

➤ To verify the authentication provider:

1 **Log on to the WebLogic Server Administration Console.**

2 **Open the Users and Groups tab of the realm that you want to use to protect Oracle applications.**

a. In **Domain Structure**, click `Security Realms`.

Summary of Security Realms opens.

b. In **Realms**, click the default (active) realm; for example, `myrealm`, that you want to use to protect Oracle applications.

c. In the settings page, select the **Users and Groups** tab.

3 **Verify that `bea_sso_ad` user is listed in Users.**

Configuring Negotiate Asserter

➤ To configure an identity asserter:

1 **Log on to the WebLogic Server Administration Console.**

2 **Open the Providers tab of the realm that you want to use to protect Oracle applications.**

a. In **Domain Structure**, click `Security Realms`.

Summary of Security Realms opens.

b. In **Realms**, click the default (active) realm; for example, `myrealm`, that you want to use to protect Oracle applications.

c. On the **Settings** page, select the **Providers** tab.

3 **In Change Center, click Lock & Edit to activate buttons.**

4 **Create an authentication provider.**

a. In **Authentication Providers**, click **New**.

b. In **Name**, enter an asserter name; for example, `spnego-asserter`.

c. In **Type**, select `NegotiateIdentityAsserter`.

d. Click **OK**.

5 **Reorder the sequence in which authentication providers are called.**

a. In **Authentication Providers**, click **Reorder**.

b. In Reorder Authentication Providers page, move the authentication provider; for example, `AD-AuthN`; as the first provider and the asserter; for example, `spnego-asserter`; as the second provider in the **Available** list.

Ensure that the `DefaultAuthenticator` is third and the `DefaultIdentityAsserter` is fourth in the Available list.

c. Click **OK**.

d. In **Change Center**, click **Activate Changes**.

6 **Restart WebLogic server.**

Granting WebLogic Administrator Role to the SSO User

You must assign WebLogic administrator role to the `bea_sso_ad` Active Directory user who is a member of the `wls_users` group. See "Create Active Directory Users" on page 8.

This step allows `bea_sso_ad` user to log in to WebLogic Administrative Console without going through a log in process.

➤ To grant WebLogic Administrator role to `bea_sso_ad`:

1 **Log on to the WebLogic Server Administration Console.**

2 **Open the Edit Global Roles screen.**

a. In **Domain Structure**, click `Security Realms`.

Summary of Security Realms opens.

b. In the **Realms** list, click the default (active) realm; for example, `myrealm`.

c. On the settings page, click the **Roles and Policies** tab.

d. Expand the **Global Roles** node.

e. Expand the **Roles** node.

f. Select `View Role Conditions` for `Admin`.

The Edit Global Role screen opens.

3 **In Change Center, click Lock & Edit to activate buttons.**

4 **Add a global role condition.**

a. Select the group **Administrators** check box.

b. Click **Add Conditions**.

c. In **Predicate List**, select `Group`.

d. Click **Next**.

e. In **Group Argument Name**, enter `wls_users`, the Active Directory group to which `bea_sso_ad` belongs.

f. Click **Add**.

g. In **Group Argument Name**, enter `Administrators`.

h. Click **Add**.

i. Click **Finish**.

5 **Click Save in Global Role Conditions screen.**

Adding Kerberos Java Options to WebLogic Startup Script

You must edit the startup script for your WebLogic domain; for example, `C:\bea\user_projects\domains\ws_domain\bin\startWeblogic.cmd`, to include the following Kerberos options.

```
set KERB_options=-Djava.security.krb5.realm=KERBEROS_REALM_NAME
–Djava.security.krb5.kdc=IP_ADDRESS_OF_KDC
–Djava.security.auth.login.config=BEA_HOME\krb5Login.conf
–Djavax.security.auth.useSubjectCredsOnly=false
–Dweblogic.security.enableNegotiate=true

set JAVA_OPTIONS=%JAVA_OPTIONS% %KERB_OPTIONS%
```

Where:

- *KERBEROS_REALM_NAME* identifies the name of the Kerberos realm; for example, `EXAMPLE.COM`
- *IP_ADDRESS_OF_KDC* is the IP address of the server that hosts the KDC (for example, IP address of `OraHypDB`).
- *BEA_HOME* is the root directory where WebLogic is installed.

Enabling Security Debugging in WebLogic (Optional)

Enabling debugging helps you identify and correct issues.

➤ To enable security debugging in WebLogic:

1 **Log on to the WebLogic Server Administration Console.**

2 **Open the Debug Settings screen.**

   a. In **Domain Structure**, click `Environments`.

   b. Select **Servers**, and then **AdminServer**.

   c. Select the **Debug** tab.

   d. Expand **weblogic**, then **Security**.

3 **Expand atn.**

4 **Expand atz.**

5 **Select DebugSecurityAtn, DebugSecurityAtz, and DebugSecurity.**

6 **Click Enable.**

7 **In Global Role Conditions, click Activate Changes.**

Deploying Workspace Web Application

In a Kerberos environment, Workspace Web application uses custom WebLogic roles and policies, which are defined as a part of the deployment process. If you have already deployed Workspace Web application without defining the required custom settings, you must redeploy it.

The Workspace Web application redeployment sequence is as follows:

- Stop Workspace
- Delete the WorkspaceWeb application
- Deploy Workspace from the WebLogic Administration Console.

Before redeploying the Workspace Web application, verify:

- WebLogic server is running (usually at port 7001).
- Workspace Server is running (usually at port 45000)

See the *Hyperion Reporting and Analysis – System 9 Installation Guide* for manual deployment instructions.

Overview of Steps

Caution! You must deploy Workspace Web application using WebLogic Administration Console. Do not use Oracle's Hyperion® Configuration Utility™ to redeploy the Workspace Web application.

- While deploying Workspace , select the settings described in "Deployment Settings" on page 18.
- After deploying Workspace Web application, verify that the custom roles and policies settings that you specified during deployment are in effect. See "Custom Roles and Policies Settings" on page 18.
- Create custom roles and policies for the URL patterns specific to Workspace. See "Creating Custom Roles and Policies for the Workspace Web Application" on page 19.

Deployment Settings

During the deployment process, specify these options in the Optional Settings page of WebLogic Install Application Assistant.

- In Security, select `Custom Roles and Policies: Use only roles and policies that are defined in the Administration console`.
- In Source accessibility, select `I will make the deployment accessible from the following location`.
- In location, enter `C:\Hyperion\deployments\WebLogic9\servers\Workspace\webapps\workspace`.

Custom Roles and Policies Settings

After deploying the Workspace Web application, verify that the custom roles and policies settings that you specified during deployment are in effect.

➤ To verify custom roles and policies settings:

1 **Log on to the WebLogic Server Administration Console.**

2 **Open the Settings for Workspace page.**

a. In **Domain Structure**, click `Deployments`.

Summary of Deployments screen opens.

b. Select `Workspace`.

Settings for Workspace opens.

3 Verify that the value of the **Security Model** on **Overview** tab is set to `CustomRolesAndPolicies`.

Creating Custom Roles and Policies for the Workspace Web Application

You must create custom roles and policies for the URL patterns specific to Workspace Web application.

➤ To create custom roles and policies:

1 Log on to the WebLogic Server Administration Console.

2 Open the Settings for Workspace page.

a. In **Domain Structure**, click `Deployments`.

Summary of Deployments opens.

b. Select `Workspace`.

Settings for Workspace opens.

3 Select the **Security** tab.

4 In **Policies** tab, create URL pattern for the Workspace Web application, and add conditions.

a. Select the **Policies** tab.

b. Click **New**.

c. Add a URL Pattern for Workspace Web application using the following settings:

- URL Pattern: `/index.jsp`
- Provider Name: `XACMLAuthorizer`

d. Click **OK**.

e. In URL Pattern, click the pattern name; for example, `/index.jsp`.

f. Click **Add Conditions**.

g. In **Predicate List**, select **Group**, and click **Next**.

h. In **Group Argument Name**, enter `wls_users`.

i. Click **Add**.

j. Click **Finish**.

Workspace Procedures

Installing WebLogic Administration Server as a Windows Service (Optional)

WebLogic console should reflect the current status of Shared Services, Workspace, and JVMs if you install WebLogic Administration Server as a Windows service that can start before Oracle's Hyperion® Shared Services starts.

➤ To install WebLogic Administration Server as a Windows service:

1 **Copy the following files from** `C:\Hyperion\deployments\WebLogic9` **into as** `C:\bea\weblogic92\server\bin\installSvc7001.cmd`

- `installSvc.cmd`.
- `uninstallSvc.cmd` (This file is used to remove the Windows service.)

2 **Rename** `C:\bea\weblogic92\server\bin` **to** `installSvc7001.cmd`

3 **Using a text editor, open** `installSvc7001.cmd`.

4 **Add the following parameters:**

```
set SERVER_NAME=AdminServer
set WLS_USER=hyperion
set WLS_PW=hyperion
set DOMAIN_NAME=WebLogic9
set USERDOMAIN_HOME=C:\Hyperion\deployments\WebLogic9
```

5 **Save** `installSvc7001.cmd`.

6 **Execute** `installSvc7001.cmd` **to create the service.**

Setting Up Workspace for Single Sign-On

Workspace delegates the process of handling external authentication and SSO to Workspace Core Services. To enable this process, you must define the trusted password that is used to establish trust between Workspace and Workspace Core Services.

Configuring Workspace for Single Sign-On

You must change some configuration properties and define a trusted password to enable SSO.

SSO configuration and the encrypted trusted password are in Workspace deployment directory; for example, `C:\Hyperion\deployments\WebLogic9\servers\Workspace\webapps\workspace\WEB-INF\config`.

- `ws.conf` (Workspace SSO configuration file)
- `tp.conf` (trusted password configuration file)

SSO settings you define are used by Workspace foundation servlets.

➤ To configure Workspace for SSO:

1 **Set up Workspace for SSO.**

a. Open a command prompt and change directory to `C:\Hyperion\BIPlus\bin`.

b. Start the Servlet Configurator by executing `config.bat`.

c. Select **Properties**, then **User Interface**, and then **Login**.

d. Use these settings:

Table 1 SSO Settings

Property	Value
`LoginPolicy class for $CUSTOM_LOGIN$`	Leave blank.
`Custom username policy`	`$REMOTE_USER$`
`Custom password policy`	`$TRUSTEDPASS$`
`Set the remote server to`	Server where Core Services is running and core services port; for example, `OraHyp4.example.com:6800`
`Set the default Authentication system to`	Leave blank.

e. Save your settings and exit the Servlet Configurator.

2 **Specify a trusted password for authentication services.**

a. Open a command prompt and change directory to `C:\Hyperion\BIPlus\bin`.

b. Execute `settrustedpass.bat`.

c. For `Current Password`, enter the current trusted password; for example, `123456`, which is the default password.

d. For `New Password` and `Confirm Password`, enter a new password; for example, `hyperion`.

Setting the Trusted Password for Authentication Service

Set the trusted password from the preceding step as the password for the Authentication Service.

Note: You must be a user; for example, `admin`, with the Global Administrator role to set the trusted password for Authentication Service.

➤ To set a trusted password for Authentication Service:

1 **Open a command prompt and change directory to `C:\Hyperion\BIPlus\bin`**

2 **Execute `ServiceConfig.bat`.**

3 **Click Show host properties.**

4 **Open the Authentication tab.**

5 **Select `Use user's login credentials for pass-through`.**

6 **Set the password. The password you set must match the SSO trusted password. See "Configuring Workspace for Single Sign-On" on page 20.**

7 **Click OK.**

Updating JVM Arguments

The Java properties that you set for Kerberos (see "Adding Kerberos Java Options to WebLogic Startup Script" on page 16) should also be set for the JVM used by Web applications; for example, Workspace Server, that start as Windows services. You do this by updating the registry entries of the Web application by adding the required JVMOption and changing the `JVMOptionCount` data.

➤ To update JVM arguments in Windows registry:

1 **Start Windows Registry Editor.**

2 **Locate the registry entry** `HKEY_LOCAL_MACHINE\SOFTWARE\Hyperion Solutions\Workspace\HyS9Workspace`.

3 **Add the following keys. You must ensure that you use the settings suitable for your environment. See "Adding Kerberos Java Options to WebLogic Startup Script" on page 16 for information on how these properties were set in** `startWebLogic.cmd`.

Note: JVMOption names indicated in the following table are examples that assume that the next available `JVMOption` name is `JVMOption13`.

Table 2 Registry Entries to Support Kerberos

Name	Type	Data
`JVMOption13`	`REG_SZ`	*`-Djava.security.krb5.realm=`*_KERBEROS_REALM_NAME_
`JVMOption14`	`REG_SZ`	†`-Djava.security.krb5.kdc=`*IP_ADDRESS_OF_KDC*
`JVMOption15`	`REG_SZ`	‡`-Djava.security.auth.login.config=`*BEA_HOME*`\krb5login.conf`
`JVMOption16`	`REG_SZ`	`-Djavax.security.auth.useSubjectCredsOnly=false`
`JVMOption17`	`REG_SZ`	`-Dweblogic.security.enableNegotiate=true`

**KERBEROS_REALM_NAME* identifies the name of the Kerberos realm; for example, `EXAMPLE.COM`

†*IP_ADDRESS_OF_KDC* is the IP address of the server that hosts the KDC (for example, IP address of `OraHypDB`, 192.190.19.202).

‡*BEA_HOME* is the root directory where WebLogic is installed; for example, `C:\bea`

4 **Update the data of** `JVMOptionCount` **to reflect the added JVMOption keys.**

5 **Close the Windows Registry Editor.**

Web Server Settings

To configure the IIS Web server for Kerberos, complete the following procedures:

- "Configure Web Service Extensions" on page 23
- "Copy WebLogic Plug-ins for IIS" on page 23
- "Create Proxy Configuration File" on page 23

- "Update Default Web Site Properties" on page 24
- "Configure Web Service Extension" on page 24
- "Add ISAPI Filter" on page 25
- "Set the Negotiate Security Header" on page 25

Ensure that the following Windows services are running:

- HTTP SSL
- IIS Admin Service
- World Wide Web Publishing Service

Configure Web Service Extensions

You must allow `All Unknown ISAPI Extensions` in IIS.

Note: You must be a member of the Administrators group on IIS host machine; for example, `OraHyp1`, to perform this procedure.

➤ To configure a Web service extension:

1 **On the IIS host server, select Start, then Programs, then Administrative Tools, and Internet Information Services (IIS) Manager.**

Internet Information Services Manager opens.

2 **Expand the node representing the IIS host; for example,** `ORAHYP1 (local computer)`

3 **Select the** `Web Service Extension` **node.**

4 **Right-click** `All Unknown ISAPI Extensions` **in the Web Service Extension list.**

5 **Select Allow.**

Copy WebLogic Plug-ins for IIS

Copy the following IIS plug-in files from *BEA_HOME*`\weblogic92\server\plugin\win\32` into IIS directory; generally `C:\iisfiles`.

- `iisproxy.dll`
- `iisforward.dll`

Create Proxy Configuration File

Create `iisproxy.ini` in IIS directory; for example, `C:\iisfiles\iisproxy.ini`.

➤ To create `iisproxy.ini`:

1 **Open a text editor.**

2 Enter the following configuration information. Be sure to use your environment settings.

```
WebLogicCluster=OraHyp2.example.com:7001,OraHyp2.example.com:45000
ConnectTimeoutSecs=20
ConnectRetrySecs=2
WlForwardPath=/
Debug=ON
```

3 Save the file as `C:\iisfiles\iisproxy.ini`.

Update Default Web Site Properties

Configure Web Service Extension

You must configure a Web service extension that uses the WebLogic plug-in `C:\iisfiles\iisproxy.dll`.

➤ To configure a Web service extension:

1 On the IIS host server, select **Start**, then **Programs**, then **Administrative Tools**, and then **Internet Information Services (IIS) Manager**.

Internet Information Services Manager opens.

2 Expand the **Web Sites** node.

3 Right-click **Default Web Site** and select **Properties**.

Default Web Site Properties opens.

4 Select the **Home Directory** tab.

5 Click **Configuration**.

Application Configuration opens.

6 Create application extension:

a. Click **Add**.

b. Enter these values:

Table 3 Application Extension Properties

Field	Value
Executable	`C:\iisfiles\iisproxy.dll`
Extension	`wlforward`
Verbs	`All Verbs`

7 Click **OK** repeatedly to return to Default Web Site Properties.

Add ISAPI Filter

You must create an ISAPI Filter that uses the WebLogic plug-in `C:\iisfiles\iisproxy.dll`.

➤ To add ISAPI filter:

1 On the IIS host server, start Internet Information Services Manager, if necessary, by selecting **Start**, then **Programs**, then **Administrative Tools**, and then **Internet Information Services (IIS) Manager**.

2 Expand **Web Sites** node.

3 Right–click **Default Web Site** and select **Properties**.

Default Web Site Properties opens.

4 Select **ISAPI Filters** tab.

5 Add filter properties:

a. Click **Add**.

b. Enter these values:

Table 4 ISAPI Filters

Field	Value
Filter name	`iis2weblogic`
Executable	`C:\iisfiles\iisproxy.dll`

c. Click **OK**.

Set the Negotiate Security Header

You must ensure that IIS supports both Kerberos and NTLM protocols by setting the Negotiate security header in the NTAuthentication Providers metabase property.

➤ To confirm that the negotiate security header is set:

1 On the IIS host server, open a command prompt window.

2 Change directory to the directory that contains the `Adsutil.vbs` file. The default location of this file is `C:\Inetpub\Adminscripts`.

3 Execute the following command to get the current values of the `NTAuthenticationProviders` metabase property:

```
cscript adsutil.vbs get w3svc/NTAuthenticationProviders
```

4 Review the NTAuthenticationProviders metabase properties listed in `C:\WINDOWS\system32\inetsrv\MBSchema.xml`.

5 If necessary, execute the following command to set the `NTAuthenticationProviders` metabase property to `negotiate`:

```
cscript adsutil.vbs set w3svc/NTAuthenticationProviders "Negotiate"
```

Client Machine Settings

All computers—machines that host Workspace components and those that will be used to access Workspace—should be set up as Kerberos clients. For example, you must configure the Web server host (`OraHyp1`), remote clients (`OraHyp3`), and application server host (`OraHyp2`) as Kerberos clients.

To configure client machines, perform these procedures:

Mapping a Local User Account to the Kerberos User Principal (Optional)

Create a local user account; for example, `beauser`, on the client computer, and map it to the Kerberos user; for example, `bea_sso_ad`. See Windows help for instructions.

Use the following procedure to map the local user account to Kerberos user.

➤ To map a local user account to the Kerberos user principal:

1 **Open a command prompt on the client machine and navigate to the directory where you copied `ksetup` tools.**

2 **Execute the following command. Be sure to modify the command for your environment.**

```
ksetup /MapUser bea_sso_ad@EXAMPLE.COM beauser
```

This sample command maps the Active Directory user `bea_sso_ad@EXAMPLE.COM` to the local user `beauser`.

Configuring Browser on Client Computers

Browsers used to access Hyperion products should be configured for Integrated Windows Authentication. You must use a browser that is capable of handling SPNEGO protocol. Internet Explorer 6 or later and Firefox 2 or later support the SPNEGO protocol.

Configuring Firefox

See Firefox asks for user name and password on internal sites, a Mozilla article, for detailed instructions.

➤ To configure Firefox for Integrated Windows Authentication:

1 **Start Firefox.**

2 **Enter the following URL in the Location bar:**

```
about:config
```

Firefox configuration settings are displayed.

3 In **Filter**, type `network.negotiate-auth.delegation-uris`.

4 Click **Show All**.

5 Under **Preference Name**, double-click `network.negotiate-auth.delegation-uris`.

6 In the Enter string vale, enter WebLogic host machine URI; for example, `http://OraHyp2.example.com`.

7 Click **OK**.

Configuring Internet Explorer 6

➤ To configure Internet Explorer 6 for Integrated Windows Authentication:

1 On the client machine, log in as a Kerberos user; for example, `bea_sso_ad`, into the Kerberos realm; for example, `EXAMPLE.COM`.

2 Start a browser session.

3 Select **Tools**, and then **Internet Options**.

4 Add Web sites to the intranet zone.

   a. In Internet Options, select **Security**, then **Local Intranet**, and then **Sites**.

   b. In Local intranet, select **Advanced**

   c. Add WebLogic host machine URI to the zone:

      i. In **Add this Web site to the zone**, enter the WebLogic host machine URI; for example, `http://OraHyp2`

      ii. Click **Add**.

   d. Repeat step 4.c to add the IIS host machine URI; for example, `http://OraHyp1`, to the intranet zone.

   e. Click **OK** repeatedly till you return to the Internet Options window.

5 Set user authentication settings:

   a. In Internet Options, select **Security**, then **Local Intranet**, and then **Custom Level**.

   b. From **Logon** under **User Authentication** settings, select **Automatic logon only in Intranet zone**.

   c. Click **OK**.

6 Specify LAN settings.

   a. In Internet Options, select **Connections**, and then **LAN Settings**.

   b. If **Use a proxy server for your LAN** (**These settings will not apply to dial-up or VPN connections**) is selected, select **Bypass proxy server for local addresses** check box.

   c. Click **OK**.

7 Verify that Integrated Windows Authentication setting is selected.

   a. In Internet Options window, select **Advanced**.

b. Verify that **Enable Integrated Windows Authentication (requires restart)** is selected.

c. Click **OK**.

8 In Internet Options, click **OK**.

9 Restart Internet Explorer.

Testing Your Deployment

Log on to the client machine and access the Workspace URL to test your deployment. Your deployment is successful if you can access Workspace without being prompted for your credentials.

➤ To test your deployment:

1 Verify that Workspace Web applications, and IIS Web server are running.

2 Log on to the client machine using Windows credentials.

3 Start a browser session.

4 Access the following Oracle's Hyperion® Workspace URL:

```
http://server_name:port_number/workspace/index.jsp
```

In the URL, `server_name` indicates the name of the computer where IIS Web server is running, and `port_number` indicates the Web server port; for example, `http://OraHyp1:19000/workspace/index.jsp`.

COPYRIGHT NOTICE

Configuring Oracle Hyperion Workspace for Kerberos Authentication, 9.3.1

Authors: EPM Information Development Team