

Oracle® Enterprise Performance Management System 安全配置指南



11.2 版
F28776-21
2023 年 12 月

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered within a solid red square.

ORACLE®

版权所有 © 2005, 2023, Oracle 和/或其附属公司。

第一作者：EPM Information Development Team

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, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

目录

文档可访问性

文档反馈

1 关于 EPM System 安全

关于 EPM System	1-1
基础知识	1-1
安全基础结构组件	1-2
用户身份验证	1-2
设置（基于角色的授权）	1-5
启动 Shared Services Console	1-8

2 为 EPM System 组件启用 SSL

假设	2-1
信息源	2-1
位置引用	2-2
关于为 EPM System 产品启用 SSL	2-2
支持的 SSL 方案	2-3
必需的证书	2-3
在 SSL 卸载器上终止 SSL。	2-4
EPM System 的完全 SSL 部署	2-6
部署体系结构	2-7
假设	2-8
为 EPM System 配置完全 SSL	2-8
重新配置 EPM System 公共设置	2-9
可选：为 WebLogic Server 安装根 CA 证书	2-10
在 WebLogic Sever 上安装证书	2-11
配置 WebLogic 服务器	2-12
启用 HFM 服务器与启用了 SSL 的 Oracle 数据库的连接	2-14
Oracle HTTP Server 过程	2-19

配置 WebLogic 服务器上部署的 EPM System Web 组件	2-22
更新域配置	2-23
重新启动服务器和 EPM System	2-25
测试部署	2-25
配置已启用 SSL 的外部用户目录	2-25
在 Web 服务器上终止 SSL	2-26
用于 Essbase 11.1.2.4 的 SSL	2-28
安装并部署 Essbase 组件	2-30
为 Essbase 使用受信任的第三方 CA 证书	2-31
建立每个会话的 SSL 连接	2-37
用于 Essbase 21c 的 SSL	2-38
安装并部署 Essbase 组件	2-40
为 Essbase 使用受信任的第三方 CA 证书	2-40
建立每个会话的 SSL 连接	2-46

3 通过安全代理启用 SSO

支持的 SSO 方法	3-1
从 Oracle Access Manager 单点登录	3-3
OracleAS Single Sign-on	3-4
测试部署	3-6
为 EPM System 启用 OSSO	3-6
针对 SSO 保护 EPM System 产品	3-10
使用身份管理产品配置基于头的 SSO	3-15
使用 Oracle Identity Cloud Service 为 EPM System 配置基于头的 SSO	3-16
先决条件和示例 URL	3-16
为 EPM System 启用基于头的身份验证	3-17
将 EPM System 应用程序和网关添加到 Oracle Identity Cloud Service	3-17
配置应用程序网关	3-22
配置用户目录以进行授权	3-22
在 EPM System 中启用 SSO	3-22
更新 EPM Workspace 设置	3-22
SiteMinder SSO	3-23
Kerberos 单点登录	3-25
针对 SSO 配置 EPM System	3-38
Smart View 的单点登录选项	3-39

4 配置用户目录

用户目录和 EPM System 安全	4-1
与用户目录配置相关的操作	4-2

Oracle Identity Manager 和 EPM System	4-2
Active Directory 信息	4-2
配置 OID、Active Directory 和其他基于 LDAP 的用户目录	4-3
将关系数据库配置为用户目录	4-15
测试用户目录连接	4-17
编辑用户目录设置	4-17
删除用户目录配置	4-18
管理用户目录搜索顺序	4-19
设置安全选项	4-20
重新生成加密密钥	4-23
使用特殊字符	4-24

5 使用自定义身份验证模块

概览	5-1
用例示例和限制	5-2
先决条件	5-3
设计和编码注意事项	5-3
部署自定义身份验证模块	5-7

6 EPM System 安全准则

实施 SSL	6-1
更改管理员密码	6-1
重新生成加密密钥	6-1
更改数据库密码	6-2
保护 Cookie 的安全	6-3
缩短 SSO 令牌超时	6-3
审核安全报表	6-3
自定义身份验证系统以实现强身份验证	6-3
禁用 EPM Workspace 调试实用程序	6-4
更改默认 Web 服务器错误页面	6-4
对第三方软件的支持	6-4

A 自定义身份验证样本代码

样本代码 1	A-1
样本代码 2	A-2
样本代码 2 的数据文件	A-4

B 实施自定义登录类

自定义登录类样本代码	B-1
部署自定义登录类	B-4

C 在用户目录之间迁移用户和组

概览	C-1
先决条件	C-1
迁移过程	C-1
特定于产品的更新	C-4

文档可访问性

有关 Oracle 对可访问性的承诺，请访问 Oracle Accessibility Program 网站 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>。

获得 Oracle 支持

购买了支持服务的 Oracle 客户可通过 My Oracle Support 获得电子支持。有关信息，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>；如果您听力受损，请访问 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>。

文档反馈

要提供有关此文档的反馈，请单击任意 Oracle 帮助中心主题中页面底部的“反馈”按钮。
还可以向 epmdoc_ww@oracle.com 发送电子邮件。

1

关于 EPM System 安全

另请参阅：

- [关于 EPM System](#)
- [基础知识](#)
- [安全基础结构组件](#)
- [用户身份验证](#)
- [设置（基于角色的授权）](#)
- [启动 Shared Services Console](#)

关于 EPM System

Oracle Enterprise Performance Management System 产品形成了一套综合的企业系统，整合了财务管理和规划应用程序模块化套件以及用于报表和分析最全面的商业智能功能。EPM System 产品的主要组件包括：

- Oracle Hyperion Foundation Services
- Oracle Essbase
- Oracle Hyperion Financial Management
- Oracle Hyperion Planning

有关上述每个产品系列中的产品和组件的信息，请参阅《*Oracle Enterprise Performance Management System 安装入门*》。

基础知识

本指南的目标读者是配置、保护和管理 Oracle Enterprise Performance Management System 组件的系统管理员，并要求具备以下基础知识：

- 深入了解组织的安全基础结构，包括：
 - 目录服务器；例如，Oracle Internet Directory、Sun Java System Directory Server 和 Microsoft Active Directory
 - 使用安全套接字层 (SSL) 保障通信通道安全
 - 访问管理系统；例如，Oracle Access Manager 和 SiteMinder
 - 单点登录 (SSO) 基础结构；例如，Kerberos
- 了解与您组织相关的 EPM System 安全概念

安全基础结构组件

Oracle Enterprise Performance Management System 集成多个安全组件，以确保可靠的应用程序安全性。集成到安全基础结构后，EPM System 提供了一整套高度安全的应用程序，可确保数据和访问安全。可用于保护 EPM System 的基础结构组件包括：

- 可选的访问管理系统；例如，Oracle Access Manager 提供对 EPM System 组件的 SSO 访问
- 使用集成的 SSO 基础结构；例如，Kerberos)
可以对访问管理系统 (SiteMinder) 使用 Kerberos 身份验证，确保 Windows 用户可以透明地登录到 SiteMinder 和 EPM System 组件。
- 使用安全套接字层 (SSL) 保护 EPM System 组件和客户端之间的通信通道

用户身份验证

用户身份验证通过验证每个用户的登录信息来确定通过验证的用户，从而实现了跨 Oracle Enterprise Performance Management System 产品的单点登录 (SSO) 功能。用户身份验证与特定于组件的授权一起向用户授予对 EPM System 组件的访问权限。授权过程被称为设置。

身份验证组件

以下各节描述支持 SSO 的各个组件：

- [Native Directory](#)
- [外部用户目录](#)

Native Directory

Native Directory 是 Oracle Hyperion Shared Services 用于支持设置和存储初始数据（如默认用户帐户）的关系数据库。

Native Directory 功能：

- 维护和管理默认 EPM System 用户帐户
- 存储所有 EPM System 设置信息（用户、组及角色之间的关系）

使用 Oracle Hyperion Shared Services Console 访问和管理 Native Directory。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“管理 Native Directory”。

外部用户目录

用户目录是与 EPM System 组件兼容的企业用户和身份管理系统。

EPM System 组件支持包括基于 LDAP 的用户目录在内的多个用户目录；例如，Oracle Internet Directory、Sun Java System Directory Server（以前称为 SunONE Directory Server）和 Microsoft Active Directory。还支持将关系数据库作为用户目录。在本文档中，除 Native Directory 之外的用户目录都指外部用户目录。

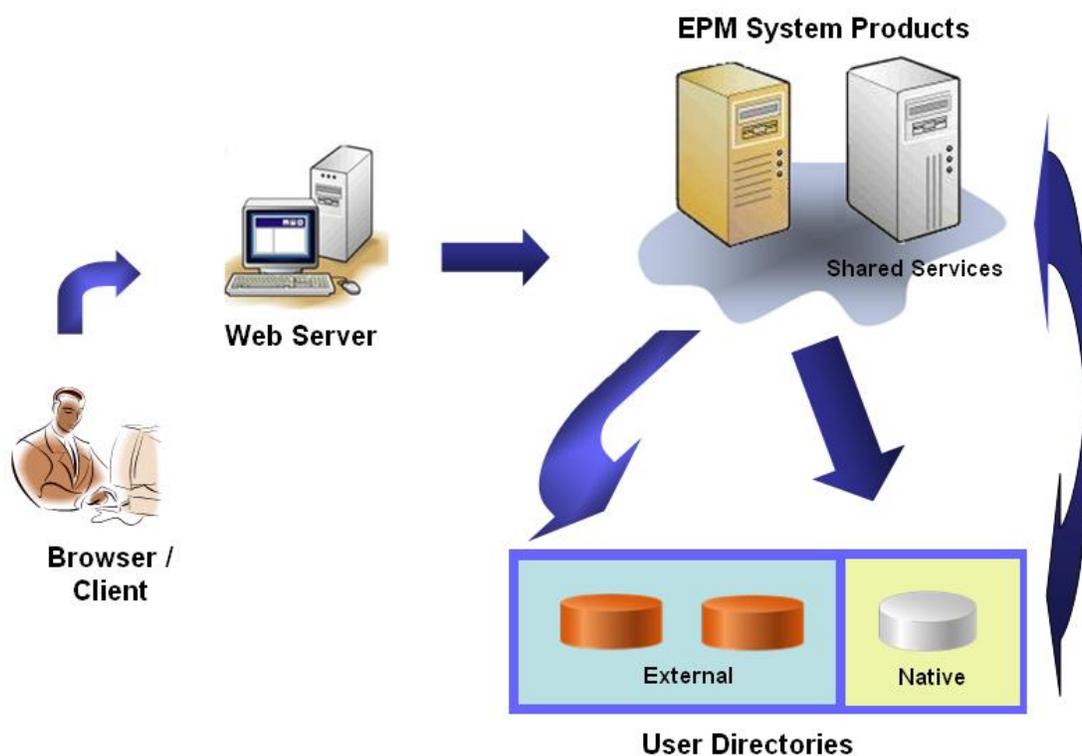
有关支持的用户目录的列表，请参阅 Oracle 技术网 (OTN) 上 "[Oracle Fusion Middleware Supported System Configurations](#)" 页面上发布的 "Oracle Enterprise Performance Management System Certification Matrix"。

从 Shared Services Console 中，您可以配置多个外部用户目录作为 EPM System 用户和组的源。每个 EPM System 用户的帐户必须在配置的用户目录中唯一。通常，可以将 EPM System 用户分配给组来简化设置过程。

默认 EPM System 单点登录

EPM System 支持跨多个 EPM System Web 应用程序使用 SSO，允许在一个应用程序中已验证身份的用户无需重新输入凭据，即可无缝导航至其他应用程序。通过集成一个公共安全环境来处理跨多个 EPM System 组件的用户身份验证和设置（基于角色的授权），即可实现 SSO。

默认的 SSO 过程如下图所示。



1. 用户通过浏览器访问 EPM System 组件登录屏幕并输入用户名和密码。

EPM System 组件查询配置的用户目录（包括 Native Directory）以验证用户凭据。一旦在用户目录中找到了匹配的用户帐户，即终止搜索，并将用户的信息返回给 EPM System 组件。

如果在配置的任何用户目录中都找不到用户帐户，则拒绝访问。

2. EPM System 组件使用检索到的用户信息查询 Native Directory，以获取该用户的设置详细信息。
3. EPM System 组件检查组件中的访问控制列表 (ACL)，以确定用户可以访问的应用程序对象。

一旦收到了来自 Native Directory 的设置信息，即会允许用户访问 EPM System 组件。此时，将为针对其设置了用户的所有 EPM System 组件启用 SSO。

来自访问管理系统的单点登录

为了进一步保障 EPM System 组件的安全，您可以实施支持的访问管理系统（例如 Oracle Access Manager 或 SiteMinder），该系统可以将已验证身份的用户的凭据提供给 EPM System 组件，并能根据预定义的访问权限控制访问。

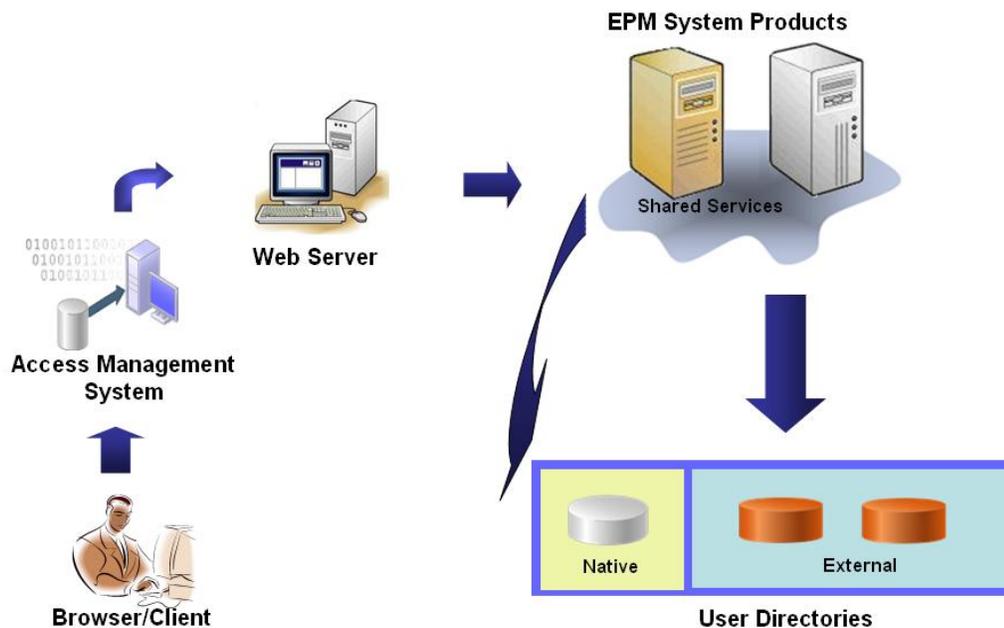
来自安全代理的 SSO 仅可用于 EPM System Web 应用程序。在此方案中，EPM System 组件使用安全代理提供的用户信息来确定用户的访问权限。为了增强安全性，Oracle 建议通过防火墙阻止对服务器的直接访问，以让所有请求都通过 SSO 门户进行路由。

通过可接受的 SSO 机制接受已验证身份的用户的凭据，可以支持来自访问管理系统的 SSO。请参阅“支持的 SSO 方法”。访问管理系统对用户进行身份验证，并将登录名传递给 EPM System。EPM System 根据配置的用户目录验证该登录名。

请参阅以下主题。

- [从 Oracle Access Manager 单点登录](#)
- [OracleAS Single Sign-on](#)
- [SiteMinder SSO](#)
- [Kerberos 单点登录](#)

概念如下图所示：



1. 用户可使用浏览器请求访问受访问管理系统（例如 Oracle Access Manager 或 SiteMinder）保护的资源。

注：

EPM System 组件被定义为受访问管理系统保护的资源。

访问管理系统将拦截请求并显示登录屏幕。用户输入用户名和密码，访问管理系统将依据配置的用户目录对用户名和密码进行验证，以核实用户身份。EPM System 组件也被配置为可以使用这些用户目录。

已验证身份的用户的个人信息将传递给 EPM System 组件，后者将接受信息并视为有效。

访问管理系统使用可接受的 SSO 机制将用户登录名（登录属性值）传递给 EPM System 组件。请参阅“支持的 SSO 方法”。

2. 为了验证用户凭据，EPM System 组件将尝试在用户目录中查找该用户。如果找到匹配的用户帐户，则将用户信息返回给 EPM System 组件。EPM System 安全性设置用于在 EPM System 组件之间启用 SSO 的 SSO 令牌。
3. EPM System 组件使用检索到的用户信息查询 Native Directory，以获取该用户的设置详细信息。

一旦收到用户设置信息，即允许用户访问 EPM System 组件。将为针对其设置了用户的所有 EPM System 组件启用 SSO。

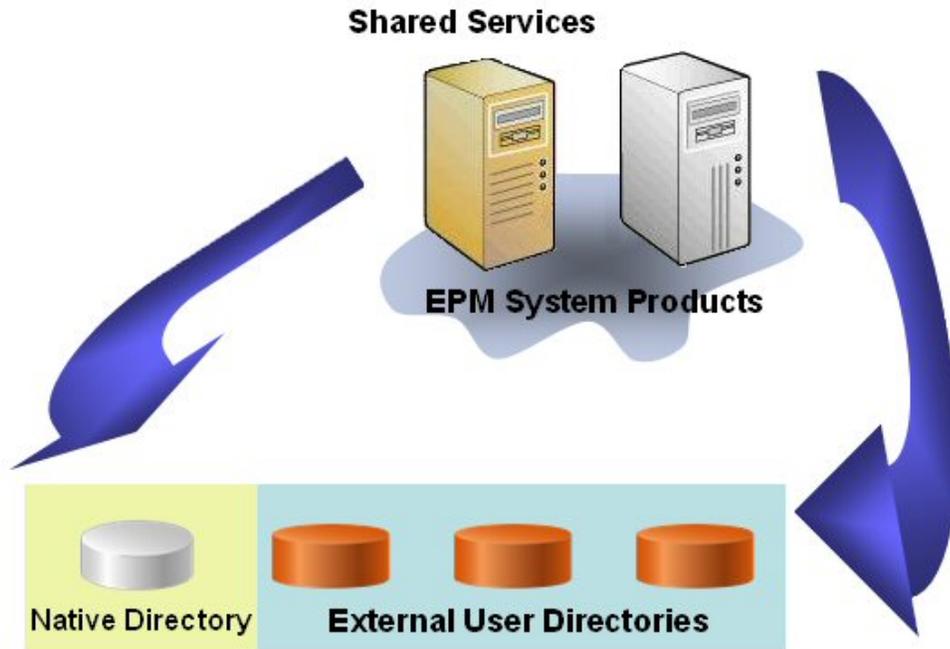
设置（基于角色的授权）

Oracle Enterprise Performance Management System 安全性使用角色这一概念来确定用户对应用程序的访问权限。角色是确定用户能否访问应用程序功能的权限。某些 EPM System 组件会强制实施对象级 ACL，以进一步细化用户对其对象（例如报表和成员）的访问权限。

每个 EPM System 组件都提供了若干针对不同业务需求定制的默认角色。属于 EPM System 组件的每个应用程序都继承了这些角色。已注册到 Oracle Hyperion Shared Services 的应用程序中的预定义角色均可从 Oracle Hyperion Shared Services Console 获得。您还可以创建另外一些聚合了默认角色的角色，以满足特定需求。这些角色可用于设置。向属于 EPM System 应用程序及其资源的用户和组授予特定角色的过程称为设置。

Native Directory 和已配置的用户目录是设置过程中使用的用户和组信息的源。您可以在 Shared Services Console 中浏览和设置所有配置的用户目录中的用户和组。您还可以使用在设置过程中在 Native Directory 中创建的特定于应用程序的聚合角色。

授权过程如下图概述：



1. 对用户进行身份验证后，EPM System 组件将查询用户目录以确定用户所在的组。
2. EPM System 组件使用组和用户信息从 Shared Services 中检索用户的设置数据。该组件使用此数据来确定用户可访问哪些资源。
为每个产品完成特定于产品的设置任务，例如，设置特定于产品的访问控制。将结合此数据和设置数据来确定用户的产品访问权限。

EPM System 产品基于角色的设置使用这些概念。

角色

角色是一种结构（类似于访问控制列表），它定义了授予用户和组的访问权限，使其可以在 EPM System 资源上执行各项功能。角色由资源或资源类型（用户可访问的内容，例如报表）与用户可对资源执行的操作（例如，查看和编辑）组合而成。

对 EPM System 应用程序资源的访问受到限制。只有在为用户或用户所属的组分配了提供访问权限的角色之后，用户才能访问这些资源。利用基于角色的访问限制，管理员可以控制和管理应用程序访问。

全局角色

全局角色是跨多个产品的 Shared Services 角色，使用户可以跨多个 EPM System 产品执行某些任务。例如，Shared Services 管理员可以为所有 EPM System 应用程序设置用户。

预定义角色

预定义角色是 EPM System 产品中的内置角色。不能删除它们。属于 EPM System 产品的每个应用程序实例都继承了该产品的预定义角色。对于每个应用程序，这些角色在应用程序创建时就注册到 Shared Services 中。

聚合角色

聚合角色，亦称自定义角色，它聚合属于一个应用程序的多个预定义角色。聚合角色可以包含其他聚合角色。例如，Shared Services 管理员或设置管理员可以创建一个聚合角色，其中同时包含 Oracle Hyperion Planning 应用程序的“规划者”和“查看用户”角色。聚合角色可以简化包括若干精细角色的应用程序的管理。聚合角色中可以包括全局 Shared Services 角色。您无法创建跨多个应用程序或产品的聚合角色。

用户

用户目录存储有关可访问 EPM System 产品的用户的信息。身份验证和授权过程都使用用户信息。您只能从 Shared Services Console 中创建和管理 Native Directory 用户。

可以在 Shared Services Console 中查看所有已配置的用户目录中的用户。可以单独设置这些用户，以便授予其对 Shared Services 中注册的 EPM System 应用程序的访问权限。Oracle 不建议分别设置各个用户。

默认 EPM System 管理员

部署过程中，会在 Native Directory 中创建一个管理员帐户，默认名称为 admin。这是权限最高的 EPM System 帐户，只应用来设置系统管理员，后者是负责管理 EPM System 安全性和环境的信息技术专家。

EPM System 管理员的用户名和密码在 Oracle Hyperion Foundation Services 部署期间设置。由于此帐户不受企业帐户密码策略限制，因此 Oracle 建议在创建系统管理员帐户后禁用它。

通常，默认 EPM System 管理员帐户用于执行以下任务：

- 将企业目录配置为外部用户目录。请参阅“[配置用户目录](#)”。
- 为企业信息技术专家设置 Shared Services 管理员角色，以创建系统管理员帐户。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“设置用户和组”。

系统管理员

系统管理员通常是对 EPM System 部署中涉及的所有服务器均具有读取、写入和执行访问权限的企业信息技术专家。

通常，系统管理员执行以下任务：

- 禁用默认 EPM System 管理员帐户。
- 至少创建一个功能管理员。
- 使用 Shared Services Console 为 EPM System 设置安全配置。
- （可选）将用户目录配置为外部用户目录。
- 通过定期运行日志分析工具来监控 EPM System。

本指南中介绍了功能管理员执行的任务。

创建功能管理员的过程如下：

- 将企业目录配置为外部用户目录。请参阅“[配置用户目录](#)”。
- 为用户或组设置所需的角色，以创建功能管理员。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“设置用户和组”。

必须为功能管理员设置以下角色：

- Shared Services 的 LCM 管理员角色
- 部署的每个 EPM System 组件的管理员和设置管理员角色

功能管理员

功能管理员是一个企业用户，并且是 EPM System 专家。该用户通常在企业目录中定义，企业目录在 Shared Services 中被配置为外部用户目录。

功能管理员执行 EPM System 管理任务，例如，创建其他功能管理员、设置授权管理、创建和设置应用程序及对象，以及设置 EPM System 审核。《Oracle Enterprise Performance Management System 用户安全管理指南》中介绍了功能管理员执行的任务。

组

组是包含用户或其他组的容器。您可以从 Shared Services Console 中创建和管理 Native Directory 组。所有已配置用户目录中的组都显示在 Shared Services Console 中。您可以设置这些组，以便授予针对向 Shared Services 注册的 EPM System 产品的权限。

启动 Shared Services Console

可以使用 Oracle Hyperion Enterprise Performance Management Workspace 中的菜单选项访问 Oracle Hyperion Shared Services Console。

要启动 Shared Services Console：

1. 转至：

`http://web_server_name:port_number/workspace`

在此 URL 中，`web_server_name` 表示运行 Oracle Hyperion Foundation Services 使用的 Web 服务器的计算机的名称，`port_number` 表示 Web 服务器端口；例如，`http://myWebserver:19000/workspace`。

注：

如果您在安全环境中访问 EPM Workspace，请使用 `https` 协议（而非 `http`）和安全的 Web 服务器端口号。例如，键入一个 URL，如：
`https://myserver:19043/workspace`。

2. 单击启动应用程序。

注：

弹出窗口阻止程序可能会阻止 EPM Workspace 打开。

3. 在登录中，输入您的用户名和密码。

最初，唯一可访问 Shared Services Console 的用户是 Oracle Enterprise Performance Management System 管理员，其用户名和密码是在部署过程中指定的。

4. 单击登录。
5. 依次选择导航、管理和 **Shared Services Console**。

2

为 EPM System 组件启用 SSL

另请参阅：

- [假设](#)
- [信息源](#)
- [位置引用](#)
- [关于为 EPM System 产品启用 SSL](#)
- [支持的 SSL 方案](#)
- [必需的证书](#)
- [在 SSL 卸载器上终止 SSL](#)
- [EPM System 的完全 SSL 部署](#)
- [在 Web 服务器上终止 SSL](#)
- [用于 Essbase 11.1.2.4 的 SSL](#)
- [用于 Essbase 21c 的 SSL](#)

假设

- 您已经确定了部署拓扑结构，并确定了要使用 SSL 进行安全保护的通信链接。
- 您已经从知名的或自己的证书颁发机构 (CA) 获得了需要的证书，或者创建了自签名证书。请参阅[“必需的证书”](#)。
- 您熟悉 SSL 概念以及诸如导入证书等过程。
有关参考文档的列表，请参阅[“信息源”](#)。

信息源

对 Oracle Enterprise Performance Management System 启用 SSL 前，要求您为各个组件（例如，应用程序服务器、Web 服务器、数据库和用户目录）做好使用 SSL 进行通信的准备。本文档假定您熟悉对这些组件启用 SSL 时涉及的任务。

- **Oracle WebLogic Server:** 请参阅《*Securing WebLogic Server Guide*》中的“[Configuring SSL](#)”。
- **Oracle HTTP Server:** 请参阅《*Oracle HTTP Server Administrator's Guide*》中的以下主题：
 - [Managing Security](#)
 - [Enabling SSL for Oracle HTTP Server](#)
- **用户目录:** 请参阅用户目录供应商提供的文档。相关链接如下：
 - **Oracle Internet Directory:** 请参阅《*Oracle Internet Directory Administrator's Guide*》

- **Sun Java System Directory Server:** 请参阅《*Sun Java System Directory Server Administration Guide*》中的 "Directory Server Security"
- **Active Directory:** 请参阅 Microsoft 文档。
- **数据库:** 请参阅数据库供应商提供的文档。

位置引用

本文档引用以下安装和部署位置：

- **MIDDLEWARE_HOME** 是指中间件组件（如 Oracle WebLogic Server）的位置，也可以是指一个或多个 **EPM_ORACLE_HOME**。**MIDDLEWARE_HOME** 是在 Oracle Enterprise Performance Management System 产品安装过程中定义的。默认的 **MIDDLEWARE_HOME** 目录为 `Oracle/Middleware`。
- **EPM_ORACLE_HOME** 是指包含支持 EPM System 产品所需文件的安装目录。**EPM_ORACLE_HOME** 位于 **MIDDLEWARE_HOME** 中。默认的 **EPM_ORACLE_HOME** 是 `MIDDLEWARE_HOME/EPMSys11R1`；例如，`Oracle/Middleware/EPMSys11R1`。
EPM System 产品安装在 **EPM_ORACLE_HOME/** 产品目录中；例如，`Oracle/Middleware/EPMSys11R1/products`。
除此之外，在 EPM System 产品配置过程中，一些产品会将组件部署到 `MIDDLEWARE_HOME/user_projects/epmsys11`；例如，`Oracle/Middleware/user_projects/epmsys11`。
- **EPM_ORACLE_INSTANCE** 表示在配置过程中定义的某些产品部署组件的位置。**EPM_ORACLE_INSTANCE** 的默认位置为 `MIDDLEWARE_HOME/user_projects/epmsys11`；例如，`Oracle/Middleware/user_projects/epmsys11`。

关于为 EPM System 产品启用 SSL

Oracle Enterprise Performance Management System 部署过程会自动以 SSL 和非 SSL 两种模式部署 Oracle EPM System 产品。

注：

- EPM System 仅支持基于 HTTP 和 JDBC 的 SSL，它不支持使用其他标准（例如 Thrift 和 ODBC）进行安全通信。
- 为了防御在 SSLv3 协议上发起攻击的 Poodle (Padding Oracle On Downgraded Legacy Encryption) 漏洞，您必须在服务器以及用于访问 EPM System 组件的浏览器中禁用 SSLv3 支持。有关禁用 SSLv3 支持的信息，请参阅服务器和浏览器文档。
- 如果您在配置 SSL 后禁用了非 SSL 模式，则 EPM System 服务器无法启动。
对域中的所有 EPM System 服务器启用安全复制，确保它们在非 SSL 模式被禁用后能够启动。

在指定 EPM System 的通用设置时，您可指定是否对部署中的所有服务器到服务器通信启用 SSL。

在部署过程中选择 SSL 设置不会自动为您的环境配置 SSL。此操作只是在 Oracle Hyperion Shared Services Registry 中设置一个标志，以指出所有使用 Shared Services Registry 的 EPM System 组件必须采用安全协议 (HTTPS) 进行服务器到服务器的通信。要为您的环境启用 SSL，必须完成额外的过程。这些步骤在本文档中有详细的讨论。

 **注：**

重新部署应用程序可删除您指定以启用 SSL 的自定义应用程序服务器和 Web 服务器设置。

 **注：**

在 Enterprise Performance Management System 11.2.x 版中，不支持存储库创建实用程序 (Repository Creation Utility, RCU) 中的 MS SQL Server 安全套接字层 (Secure Sockets Layer, SSL)。

支持的 SSL 方案

支持以下 SSL 方案：

- 终止 SSL 卸载器中的 SSL。请参阅“[在 SSL 卸载器上终止 SSL](#)”。
- 完全 SSL 部署。请参阅“[EPM System 的完全 SSL 部署](#)”。

必需的证书

SSL 通信使用证书在组件之间建立信任。Oracle 建议您在生产环境中将来自知名第三方的 CA 证书运用到已启用 SSL 的 Oracle Enterprise Performance Management System 中。

 **注：**

EPM System 支持使用通配符证书，它可以是一个 SSL 证书保护多个子域。使用通配符证书可以减少管理时间和成本。

如果您使用通配符证书加密通信，则必须在 Oracle WebLogic Server 中禁用主机名验证。

您需要为托管 EPM System 组件的每个服务器提供以下证书：

- 根 CA 证书

 注：

如果您使用的是来自知名第三方 CA 证书（其根证书已安装在 Java 密钥库中），则无需在 Java 密钥库中安装根 CA 证书。

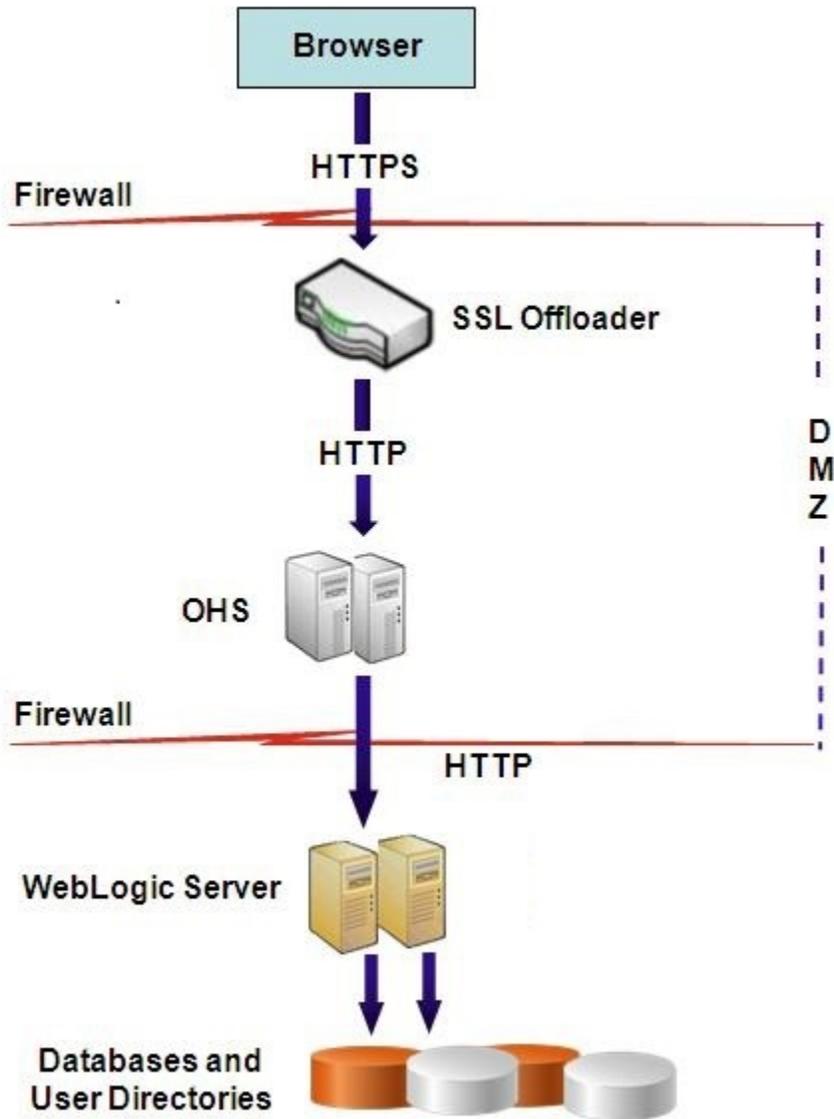
Firefox 和 Internet Explorer 已预加载知名第三方 CA 的证书。如果您充当自身的 CA，则必须将您的 CA 根证书导入从此类浏览器访问的客户端使用的密钥库中。如果您充当自身的 CA，但访问客户端的浏览器不能使用您的根 CA 证书，则 Web 客户端无法与服务器建立 SSL 握手。

- 为部署中的每个 Oracle HTTP Sever 提供的签名证书
- 为 WebLogic Server 主机提供的签名证书此计算机上的受管服务器也可以使用此证书
- 适用于 SSL 卸载器/负载均衡器的两个证书。上述证书之一用于外部通信，另一证书用于内部通信

在 SSL 卸载器上终止 SSL。

部署体系结构

在此方案中，使用 SSL 保护 Oracle Enterprise Performance Management System 客户端（例如浏览器）和 SSL 卸载器之间通信链路的安全。概念如下图所示：



假设

SSL 卸载器和负载均衡器

部署环境中必须具有完全配置的 SSL 卸载器和负载均衡器。

负载均衡器必须配置为将虚拟主机收到的所有请求转发给 Oracle HTTP Server。

当 SSL 在 Oracle HTTP Server (OHS) 或负载均衡器上终止时，您必须：

- 将每个逻辑 Web 应用程序设置为负载均衡器或 Oracle HTTP Server 的非 SSL 虚拟主机（例如 `empinternal.myCompany.com:80`，其中 80 是非 SSL 端口）。打开“配置”屏幕，完成以下步骤：
 1. 展开 **Hyperion Foundation** 配置任务。
 2. 选择配置 Web 应用程序的逻辑地址。
 3. 指定主机名、非 SSL 端口号和 SSL 端口号。

- 将外部 URL 设置为负载均衡器或 Oracle HTTP Server 的启用 SSL 的虚拟主机（例如，`empexternal.myCompany.com:443`，其中 443 是 SSL 端口）。打开“配置”屏幕，完成以下步骤：
 1. 展开 **Hyperion Foundation** 配置任务。
 2. 选择配置公共设置。
 3. 选择“外部 URL 详细信息”下的启用 **SSL** 卸载。
 4. 指定外部 *URL* 主机和外部 *URL* 端口。

 **注：**

使用 **configtool** 重新部署 Web 应用程序或重新配置 Web 服务器将替换逻辑 Web 应用程序和外部 URL 的设置。

虚拟主机

在 SSL 卸载器中终止 SSL 的配置在 SSL 卸载器/负载均衡器上使用两个服务器别名（例如，`epm.myCompany.com` 和 `empinternal.myCompany.com`）：一个用于卸载器和浏览器之间的外部通信，另一个用于各 EPM System 服务器之间的内部通信。确保服务器别名指向该计算机的 IP 地址，且可通过 DNS 解析。

卸载器/负载均衡器上必须安装支持卸载器和浏览器之间外部通信（通过 `epm.myCompany.com`）的签名证书。

配置 EPM System

EPM System 组件的默认部署支持在 SSL 卸载器上终止 SSL。无需其他操作。

配置 EPM System 时，确保逻辑 Web 应用程序指向为内部通信创建的别名（例如 `empinternal.myCompany.com`）。请查看以下信息源以安装和配置 EPM System：

- 《Oracle Enterprise Performance Management System 安装与配置指南》
- 《Oracle Enterprise Performance Management System 安装入门》
- 《Oracle Enterprise Performance Management System 安装与配置故障排除指南》

测试部署

完成部署过程后，通过连接到安全的 Oracle Hyperion Enterprise Performance Management Workspace URL 验证一切是否正常：

`https://virtual_host_external:SSL_PORT/workspace/index.jsp`

例如，`https://epm.myCompany.com:443/workspace/index.jsp`，其中 443 为 SSL 端口。

EPM System 的完全 SSL 部署

另请参阅：

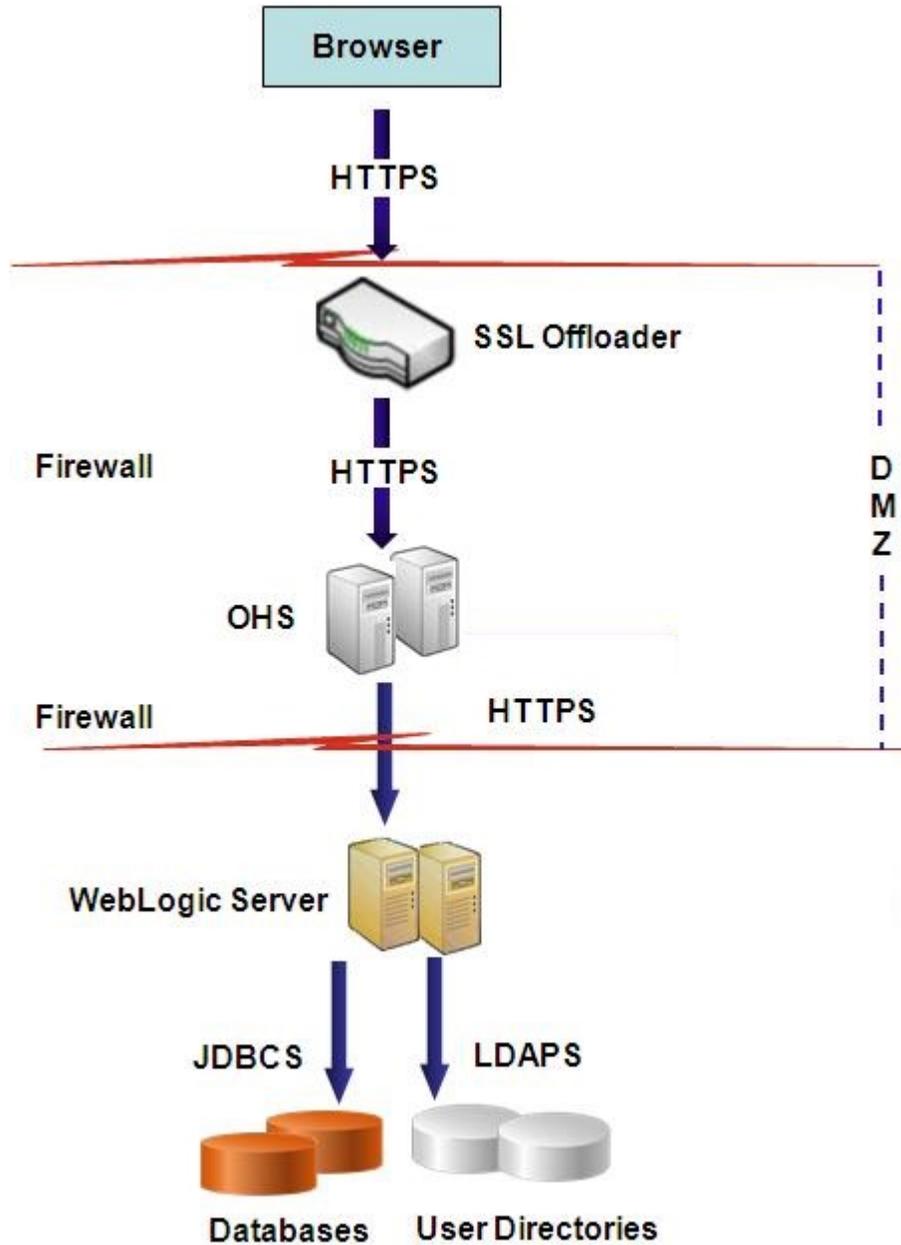
- [部署体系结构](#)

- 假设
- 针对完全 SSL 配置 EPM System

部署体系结构

在完全 SSL 模式中，使用 SSL 保护所有安全通道之间的通信。这种 Oracle Enterprise Performance Management System 部署方案极其安全。

概念如下图所示：



假设

数据库

已为数据库服务器和客户端启用 SSL。有关为数据库服务器和客户端启用 SSL 的信息，请参阅数据库文档。

EPM System

已安装和部署 Oracle Enterprise Performance Management System 组件（包括 Oracle WebLogic Server 和 Oracle HTTP Server）。此外，EPM System 环境已经过测试，确保所有组件均在非 SSL 模式中正常运行。查看以下信息源：

- 《Oracle Enterprise Performance Management System 安装与配置指南》
- 《Oracle Enterprise Performance Management System 安装入门》
- 《Oracle Enterprise Performance Management System 安装与配置故障排除指南》

如果您计划在配置过程中为数据库连接启用 SSL，则必须在每个数据库配置屏幕中选择高级选项链接，然后指定包括如下所示的必需设置：

- 选择使用到数据库的安全连接(SSL) 并输入安全的数据库 URL；例如：
`jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=myDBhost)(PORT=1529)(CONNECT_DATA=(SERVICE_NAME=myDBhost.myCompany.com)))`
- 可信的密钥库
- 可信的密钥库密码

有关详细信息，请参阅《Oracle Enterprise Performance Management System 安装与配置指南》。

SSL 卸载器和负载平衡器

部署环境中必须具有完全配置的 SSL 卸载器和负载平衡器。

完整的 SSL 配置在 SSL 卸载器上使用两个服务器别名，例如 `epm.myCompany.com` 和 `empinternal.myCompany.com`。一个用于卸载器和浏览器之间的外部通信，另一个用于 EPM System 服务器间的内部通信。确保服务器别名指向该计算机的 IP 地址，且可通过 DNS 解析。

负载均衡器必须配置为将虚拟主机收到的所有请求转发给 Oracle HTTP Server。

卸载器/负载平衡器上必须安装两个签名证书：一个用于支持卸载器和浏览器之间的外部通信（通过 `epm.myCompany.com`），另一个用于支持应用程序间的内部通信（通过 `empinternal.myCompany.com`）。Oracle 建议将这些证书绑定到服务器别名，以防止暴露服务器名称并提高安全性。

为 EPM System 配置完全 SSL

另请参阅：

- [重新配置 EPM System 公共设置](#)
- [可选：为 WebLogic Server 安装根 CA 证书](#)
- [在 WebLogic Server 上安装证书](#)

- 配置 WebLogic Server
- 启用 HFM 服务器与启用了 SSL 的 Oracle 数据库的连接
- Oracle HTTP Server 过程
- 配置已在 WebLogic Server 上部署的 EPM System Web 组件
- 更新域配置
- 重新启动服务器和 EPM System
- 测试部署
- 配置已启用 SSL 的外部用户目录

重新配置 EPM System 公共设置

在此过程中，您选择强制 Oracle Enterprise Performance Management System 组件使用 SSL 通信的设置。

注：

如果您要为 **Oracle Hyperion Financial Management Web** 服务器启用 SSL：在配置 Financial Management 之前，必须编辑 `weblogic.xml` 中 HFM WebApp 的会话描述符以确保 cookie 的安全。

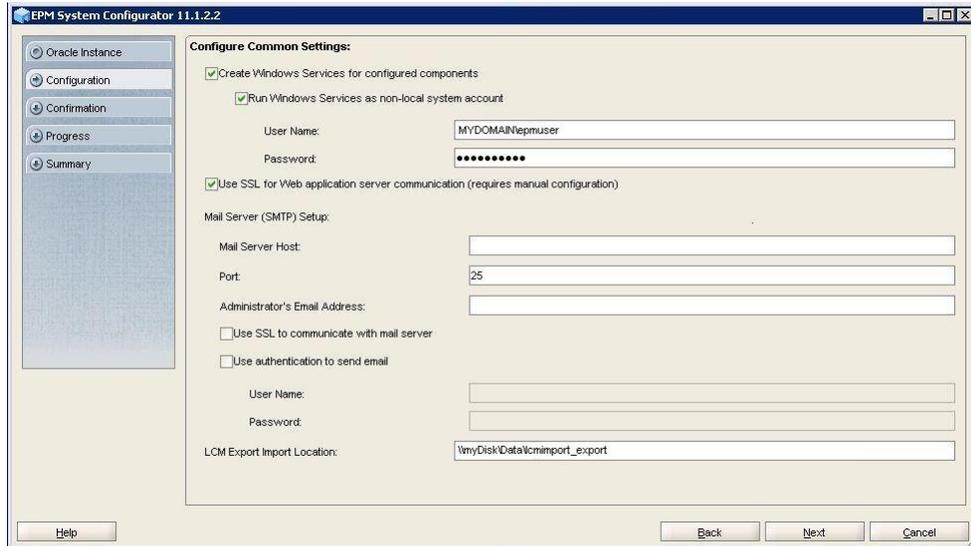
1. 使用 7-Zip 等工具展开 Financial Management Web 存档。`weblogic.xml` 在存档中的位置为
`EPM_ORACLE_HOME\products\FinancialManagement\AppServer\InstallableApps\HFMWebApplication.ear\HFMWeb.war\WEB-INF\weblogic.xml`
2. `weblogic.xml` 中的 HFM WebApp 会话描述符包含以下指令：
`<cookie-secure>true</cookie-secure>`
3. 保存 `weblogic.xml`
4. 当 7-Zip 询问是否更新存档时，单击是。

要为 EPM System 重新配置 SSL：

1. 启动 EPM System Configurator。
2. 在请选择要将配置应用到的 **EPM Oracle** 实例中，完成以下步骤：
 - a. 在 **EPM Oracle** 实例名称中，输入最初配置 EPM System 组件时使用的实例名称。
 - b. 单击下一步。
3. 在“配置”屏幕上，完成以下步骤：
 - a. 清除取消全选。
 - b. 展开 **Hyperion Foundation** 配置任务，然后选择配置公共设置。
 - c. 单击下一步。
4. 在配置公共设置中，完成以下步骤：

注意：

在选择该设置来使用 SSL 与电子邮件服务器进行通信之前，确保已针对 SSL 配置电子邮件服务器。



- a. 选择对 **Java Web** 应用程序服务器通信使用 **SSL**（需要手动配置），以指定 EPM System 应使用 SSL 进行通信。
 - b. 可选：在邮件服务器主机和端口中输入信息。为支持 SSL 通信，您必须指定 SMTP 邮件服务器使用的安全端口。
 - c. 可选：要支持与 SMTP 邮件服务器进行 SSL 通信，请选择使用 **SSL** 与邮件服务器进行通信。
 - d. 在其余字段中选择或输入设置。
 - e. 单击下一步。
5. 在后续的 EPM System Configurator 屏幕上，单击下一步。
 6. 完成部署过程后，将显示“摘要”屏幕。单击完成。

可选：为 WebLogic Server 安装根 CA 证书

大多数知名第三方 CA 的根证书已安装在 JVM 密钥库中。如果您没有使用来自知名第三方 CA 的证书（不推荐），请完成本节中所述的过程。默认的 JVM 密钥库位置为 `MIDDLEWARE_HOME/jdk/jre/lib/security/cacerts`。

注：

在每个 Oracle Enterprise Performance Management System 服务器上执行此过程。

要安装根 CA 证书：

1. 将根 CA 证书复制到安装 Oracle WebLogic Server 的计算机的本地目录中。
2. 从控制台中，将目录转至 `MIDDLEWARE_HOME/jdk/jre/bin`。
3. 按如下所示执行 `keytool` 命令，将根 CA 证书安装到 JVM 密钥库中：

```
keytool -import -alias ALIAS -file CA_CERT_FILE -keystore KEYSTORE -  
storepass KEYSTORE_PASSWORD -trustcacerts
```

例如，您可以使用以下命令将当前目录中存储的证书 `CAcert.crt` 添加到 JVM 密钥库中，并将 `Blister` 用作密钥库中的证书别名。假定为 `Storepassexample_pwd`。

```
keytool -import -alias Blister -file CAcert.crt -keystore ../lib/security/  
cacerts -storepass example_pwd -trustcacerts
```

 注：

上述命令和示例使用了利用 `keytool` 导入证书的一些语法。有关导入语法的完整列表，请参阅 `keytool` 文档。

在 WebLogic Sever 上安装证书

默认 Oracle WebLogic Server 安装使用演示证书支持 SSL。Oracle 建议您安装知名第三方的证书，以增强环境安全性。

在托管 WebLogic Server 的每个计算机上，使用工具（例如 `keytool`）创建一个自定义密钥库，以存储 WebLogic Server 和 Oracle Enterprise Performance Management System Web 组件的签名证书。

要创建自定义密钥库并导入证书：

1. 从控制台中，将目录转至 `MIDDLEWARE_HOME/jdk/jre/bin`。
2. 执行类似如下的 `keytool` 命令，在现有目录中创建自定义密钥库（由命令中的 `-keystore` 指令标识）：

```
keytool -genkey -dname "cn=myserver, ou=EPM, o=myCompany, c=US" -alias  
epm_ssl -keypass password -keystore  
C:\oracle\Middleware\EPMSysstem11R1\ssl\keystore -storepass password -  
validity 365 -keyalg RSA
```

 注：

您设置的通用名 (cn) 必须与服务器名称匹配。如果您使用完全限定域名 (FQDN) 作为 cn，则部署 Web 组件时必须使用 FQDN。

3. 生成证书请求。

```
keytool -certreq -alias epm_ssl -file C:/certs/epmssl_csr -keypass  
password -storetype jks -keystore  
C:\oracle\Middleware\EPMSysystem11R1\ssl\keystore -storepass password
```

4. 获取 WebLogic Server 计算机的签名证书。

5. 将签名证书导入密钥库：

```
keytool -import -alias epm_ssl -file C:/certs/epmssl_cert -keypass  
password -keystore C:\Oracle\Middleware\EPMSysystem11R1\ssl\keystore -  
storepass password
```

配置 WebLogic 服务器

部署 Oracle Enterprise Performance Management System Web 组件之后，必须进行相应配置以实现 SSL 通信。

要为 Web 组件配置 SSL：

1. 执行 `MIDDLEWARE_HOME/user_projects/domains/EPMSysystem/bin/startWebLogic.cmd` 以启动 Oracle WebLogic Server：
2. 通过访问以下 URL 启动 WebLogic Server 管理控制台：

```
http://SERVER_NAME:Port/console
```

例如，要访问部署在 `myServer` 默认端口上的 WebLogic Server 控制台，应使用 `http://myServer:7001/console`。

3. 在“欢迎”屏幕上，输入在 EPM System Configurator 中指定的 WebLogic Server 管理员用户名和密码。
4. 在更改中心内，单击锁定并编辑。
5. 在控制台的左窗格中，展开环境，然后选择服务器。
6. 在“服务器摘要”屏幕中，单击您要为其启用 SSL 的服务器的名称。

例如，要为 Oracle Hyperion Foundation Services 组件启用 SSL，可以使用 `EPMServer0` 服务器。

7. 清除已启用侦听端口以禁用 HTTP 侦听端口。
8. 确保已选择已启用 SSL 侦听端口。
9. 在 SSL 侦听端口中，输入此服务器应侦听请求的 SSL 侦听端口。
10. 要指定将使用的标识和信任密钥库，选择密钥库以打开“密钥库”选项卡。
11. 单击更改。
12. 请选择一个选项：
 - 自定义标识和自定义信任（如果您未使用来自知名第三方 CA 的服务器证书）
 - 自定义标识和 Java 标准信任（如果您使用了来自知名第三方 CA 的服务器证书）
13. 单击保存。

14. 在自定义标识密钥库中，输入签名的 WebLogic Server 证书安装到的密钥库的路径。
15. 在自定义标识密钥库类型中，输入 `jks`。
16. 在自定义标识密钥库密码短语和确认自定义标识密钥库密码短语中，输入密钥库密码。
17. 如果您已在密钥库中选择自定义标识和自定义信任：
 - 在自定义信任密钥库中，输入为服务器证书签名的 CA 的根证书所在的自定义密钥库的路径。
 - 在自定义信任密钥库类型中，输入 `jks`。
 - 在自定义信任密钥库密码短语和确认自定义信任密钥库密码短语中，输入密钥库密码。
18. 单击保存。
19. 指定 SSL 设置：
 - 选择 **SSL**。
 - 在私钥别名中，输入您在导入已签名的 WebLogic Server 证书时指定的别名。
 - 在私钥密码短语和确认私钥密码短语中，输入用于检索私钥的密码。
 - 单击保存。

 **注：**

如果您使用 SHA-2 证书，则必须为用于支持 EPM System 的每个受管服务器选择使用 **JSSE SSL** 设置。此设置位于 SSL 页的“高级”选项卡中。您需要重新启动 WebLogic 服务器以激活此更改。

20. 对服务器启用安全复制：
 - a. 在控制台的左窗格中，展开环境，然后单击群集。
 - b. 在“群集摘要”中，单击您要为其启用安全复制的服务器的名称，例如 `Foundation Services`。
此时，将显示选定服务器“设置”屏幕的“配置”选项卡。
 - c. 单击复制以打开“复制”选项卡。
 - d. 选择已启用安全复制。您可能需要先单击锁定并编辑，然后才能选择此选项。
 - e. 单击保存。
21. 对属于此主机的每个受管服务器，完成步骤 6 到 20。
22. 启用安全复制，为群集的复制调用提供通道。
有关详细信息，请参阅 Oracle Metalink 文档 1319381.1。
 - 在管理员控制台中，展开环境，然后选择群集。
 - 选择复制。
 - 在复制上，选择（选中）已启用安全复制。
 - 单击保存。
23. 在更改中心内，单击激活更改。

启用 HFM 服务器与启用了 SSL 的 Oracle 数据库的连接

HFM 数据源和 Oracle 数据库之间的网络连接可以使用 SSL 进行加密。为此，必须按照 [Oracle 文档](#) 中的说明配置 Oracle Wallet。TNS 侦听器还必须配置为侦听 SSL 加密连接的新端口。最后，需要将相应的证书加载到托管 HFM 数据源的服务器上的密钥库和信任库中。以下说明引用自 [Oracle 数据库文档](#)。

先决条件

在继续执行以下步骤之前，请确保满足以下先决条件：

- 正常运行的数据库服务器。
- 确保没有本地或网络防火墙阻止在运行已启用 SSL 的 TNS 侦听器的端口上与服务器进行通信。

在下面的示例中，使用了在 MS Windows Server 2016 上运行的 Oracle 12c (12.1.0.2) 版本。如果为 wallet 文件指定的路径是 Linux 文件系统路径，并且已针对数据库服务器上使用的 shell 正确更改了环境变量替代项，则这些说明在 Linux 安装上同样适用。这些相同的说明已成功用于 19c 开发和支持实例。

本文中的示例使用自签名证书，但如果您愿意，也可以使用适当的证书颁发机构证书。要了解安装证书颁发机构颁发的证书时要遵循的确切步骤，请参阅 [Oracle 数据库文档](#)。

配置 Oracle 数据库

要配置 Oracle 数据库，请按照以下步骤操作：

1. 在数据库服务器上创建新的自动登录 wallet。

注：

仅当之前未创建 Oracle Wallet 时才需要执行这些步骤。如果在数据库服务器上使用 GUI Oracle Wallet 工具，则无需执行以下步骤。

```
C:\> cd %ORACLE_HOME%
C:\oracledb\12.1.0\home> mkdir wallet
C:\oracledb\12.1.0\home> orapki wallet create -wallet wallet -pwd
password1 -auto_login
```

可以忽略任何提示您在 orapki 命令行上使用 -auto_login_local 的消息。如果遇到 SSL 身份验证失败错误，请参阅 [文档 ID 2238096.1](#) 以解决此问题。此外，请检查 cwallet.sso 文件（在 wallet 目录下）的安全权限，并确保 Oracle 侦听器服务用户对此文件有读取权限。如果没有读取权限，稍后进行 SSL 握手将失败。如果使用不允许登录的建议 Oracle 用户安装 Oracle 数据库，则会出现这种情况。如果使用 Oracle 用户安装 Oracle 数据库，则 TNS 侦听器必须以不同的用户身份运行。

2. 创建自签名证书并将其加载到 wallet 中

```
C:\oracledb\12.1.0\home> orapki wallet add -wallet wallet -pwd password1 -  
dn "CN={FQDN of db server}" -  
keysize 1024 -self_signed -validity 3650
```

以上示例中的 password1 密码必须与步骤 1 中指定的密码匹配。

3. 导出新创建的自签名证书

```
C:\oracledb\12.1.0\home> orapki wallet export -wallet wallet -pwd  
password1 -dn "CN={FQDN of db server}"  
-cert %COMPUTERNAME%-certificate.crt
```

4. 将导出的 Base64 证书文件复制到 HFM 服务器。

5. 配置 SQL*NET 和 TNS 侦听器：

- a. 确定数据库服务器上未使用的端口。以下示例在端口 1522 上创建新侦听器。用于 SSL 连接的典型端口是 2484，您可以使用任何可用端口。在继续操作之前，您必须检查要使用的端口在数据库服务器上是否可用，并根据需要进行调整。
- b. 更新 SQLNET.ORA。WALLET_LOCATION 声明的 DIRECTORY 元素必须指向上面的步骤 1 中创建的 wallet。

```
SQLNET.AUTHENTICATION_SERVICES= (TCPS, NTS, BEQ)  
NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)  
WALLET_LOCATION=  
  (SOURCE =  
   (METHOD = FILE)  
   (METHOD_DATA =  
    (DIRECTORY = C:\oracledb\12.1.0\home\wallet)  
   )  
  )  
SSL_CLIENT_AUTHENTICATION = FALSE
```

- c. 更新 LISTENER.ORA 以定义新侦听器。使用在上面的步骤 5a 中确定的端口。

```
SID_LIST_LISTENER =  
  (SID_LIST =  
   (SID_DESC =  
    (SID_NAME = CLRExtProc)  
    (ORACLE_HOME = C:\oracledb\12.1.0\home)  
    (PROGRAM = extproc)  
    (ENVS = "EXTPROC_DLLS=ONLY:C:\oracledb\12.1.0\home\bin\oraclr12.dll")  
   )  
  )  
SSL_CLIENT_AUTHENTICATION = FALSE  
WALLET_LOCATION=  
  (SOURCE =  
   (METHOD = FILE)  
   (METHOD_DATA =  
    (DIRECTORY = C:\oracledb\12.1.0\home\wallet)  
   )  
  )
```

```

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST = myServer) (PORT = 1521))
    )
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCPS) (HOST = myServer) (PORT = 1522))
    )
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
    )
  )
ADR_BASE_LISTENER = C:\oracledb

```

- d. 在 TNSNAMES.ORA 中为新端口创建新条目。

```

ORCL_SSL =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCPS) (HOST = myServer) (PORT = 1522))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = myServer_service)
    )
  )

```

必须指定在上面的步骤 5a 中确定并在步骤 5c 中使用的相同端口。

- e. 重新启动 TNS 侦听器。

```

C:\oracledb\12.1.0\home>lsnrctl stop
C:\oracledb\12.1.0\home>lsnrctl start

```

- f. 验证新 TNS 侦听器是否正常运行

```

C:\oracledb\12.1.0\home>tnsping orcl_ssl
TNS Ping Utility for 64-bit Windows: Version 12.1.0.2.0 -
Production on 10-SEP-2019 15:43:22
Copyright (c) 1997, 2014, Oracle. All rights reserved.
Used parameter files:
C:\oracledb\12.1.0\home\network\admin\sqlnet.ora
Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCPS)
(HOST = myServer)
(PORT = 1522)) (CONNECT_DATA = (SERVER = DEDICATED)
(SERVICE_NAME = myServer_service)))
OK (130 msec)

```

将 HFM 服务器配置为使用 SSL 数据库连接

将数据库的证书添加到 HFM 服务器上的信任库

必须在运行 HFM 数据源的每个 EPM 服务器上执行以下步骤。下面使用的 `%MW_HOME%` 环境变量是 Oracle Middleware 安装的位置。在 EPM 安装期间默认不创建此环境变量，此处用于显示 EPM 安装的父目录。

EPM 安装的位置由 `EMP_ORACLE_HOME` 环境变量指定。下面的示例将密钥库和信任库放置在与 EPM 安装位置相同的目录中。密钥库和信任库文件可以位于 HFM 服务器文件系统的任何位置。

1. 在 `%MW_HOME%` 下创建一个新目录来存储 Java 密钥库和 PKCS12 信任库。
 - a. `cd %MW_HOME%`
 - b. `mkdir certs`
2. 从 JDK 复制 Java 密钥库文件 `cacerts`。
 - a. `cd %MW_HOME%\certs`
 - b. 复制 `%MW_HOME%\jdk1.8.0_181\jre\lib\security\cacerts` `testing_cacerts`
复制并使用该 JDK 密钥库而非使用 JDK 默认密钥库的原因是，如果升级 JDK 并删除了之前的 JDK，插入到默认密钥库中的密钥和证书将丢失。
3. 将此 Base 64 证书复制到 `%MW_HOME%\certs`。
4. 将证书导入到 Java 密钥库文件 `testing_cacerts`。
 - a. 例如，`keytool -importcert -file bur00cbb-certificate.crt -keystore testing_cacerts -alias "myserver"`
 - i. 您将必须指定密钥库的密码。
 - ii. 您应该将 "myserver" 替换为数据库服务器的完全限定域。
 - b. 当系统提示您是否应信任证书的问题时，请指定 `y`。
5. 从 JDK 的 Java 密钥库文件创建 PKCS12 格式的信任库。例如：

```
keytool -importkeystore -srckeystore testing_cacerts -srcstoretype JKS -  
deststoretype PKCS12 -destkeystore testing_cacerts.pfx
```

将 HFM JDBC 连接更新为使用 SSL

1. 将 HFM 数据库 JDBC 连接重新配置为使用 SSL。
 - a. 启动 EPM 配置工具。
 - i. 选择 **Financial Management** 节点下的配置数据库和部署到应用程序服务器节点。
 - ii. 单击下一步。
 - iii. 为 HFM JDBC 连接执行以下每个步骤
 - i. 在端口、服务名称、用户名和密码列中输入连接的 SSL 端口、服务名称、用户名和密码。
 - ii. 单击 (+) 以打开高级数据库选项。
 - iii. 选中使用安全连接复选框。
 - iv. 输入在步骤 2 中创建的 Java 密钥库的位置。
 - v. 单击应用。
 - vi. 单击 (+) 以打开高级数据库选项。
 - vii. 单击编辑并使用修改后的 **JDBC URL**。请注意，不应显示对 JDBC URL 进行任何更改。
 - viii. 单击应用。

- ix. 单击下一步。
 - b. 按照 EPM 文档中所述，完成其余步骤以部署 HFM 应用程序。
2. 打开命令窗口或 shell 以手动更新 EPM 注册表，以便数据源使用的 ODBC 连接可以启用 SSL。
执行下列每个命令：

```
epmsys_registry.bat addproperty FINANCIAL_MANAGEMENT_PRODUCT/
DATABASE_CONN/@ODBC_TRUSTSTORE "C:
\Oracle\Middleware\certs\testing_cacerts.pfx"
epmsys_registry.bat addencryptedproperty
FINANCIAL_MANAGEMENT_PRODUCT/DATABASE_CONN
/@ODBC_TRUSTSTOREPASSWORD <truststorepassword>
epmsys_registry.bat addproperty FINANCIAL_MANAGEMENT_PRODUCT/
DATABASE_CONN
/@ODBC_VALIDATESERVERCERTIFICATE false
```

在以上示例中，路径 C:\Oracle\Middleware 是步骤 1、2 和 3 中的 %MW_HOME% 值。

仅当使用自签名证书时，才应将属性 FINANCIAL_MANAGEMENT_PRODUCT/DATABASE_CONN/@ODBC_VALIDATESERVERCERTIFICATE 设置为 false。
FINANCIAL_MANAGEMENT_PRODUCT/DATABASE_CONN/@ODBC_TRUSTSTOREPASSWORD 的值应该是步骤 2 中复制的原始 Java 密钥库的密码。

更新 HFM 使用的 TNS 名称条目

编辑 TNSNAMES.ORA 可创建新条目并重命名旧条目。以下示例显示了 HFM 服务器上已应用必要更改的更新后 TNSNAMES.ORA 文件。进行这些更改的原因是 HFM 查找并使用名为 HFMTNS 的 TNS 名称条目。必须更改此条目的协议和端口，XFMDDataSource 才能正常工作。

```
HFMTNS_UNENC =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = myserver)(PORT = 1521))
)
(CONNECT_DATA =
(SERVICE_NAME = myserver_service)
(SERVER = DEDICATED)
)
)
)
HFMTNS =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCPS)(HOST = myserver)(PORT = 1522))
)
(CONNECT_DATA =
(SERVICE_NAME = myserver_service)
(SERVER = DEDICATED)
)
)
)
```

原始 HFMTNS 条目已重命名为 HFMTNS_UNENC。通过复制 HFMTNS_UNENC 条目并将其重命名为 HFMTNS 生成了新的 HFMTNS。然后，协议更新为 TCPS，并且端口更改为 1522。指定的端口必须与 TNS_LISTENER.ORA 文件中指定的端口相同。

Oracle HTTP Server 过程

为 Oracle HTTP Server 创建 Wallet 并安装证书

默认 Wallet 随 Oracle HTTP Server 自动安装。您必须为部署中的每个 Oracle HTTP Server 配置一个实际的 Wallet。

注意：从 11.2.x 开始，Oracle Wallet Manager 不再随 Oracle HTTP Server 一起安装。仅当您安装 Oracle Database Client 时，才会安装 Oracle Wallet Manager。您必须使用数据库客户端提供的 wallet 管理器来创建 wallet 并导入证书。如果要为 SSL 配置 Oracle HTTP Server，请确保在 EPM 系统产品安装期间始终安装 64 位 Oracle Database Client。

创建并安装 Oracle HTTP Server 证书：

1. 在托管 Oracle HTTP Server 的每台计算机上，启动 Wallet Manager。
依次选择开始、所有程序、**Oracle-OHxxxxxx**、集成管理工具和 **Wallet Manager**。
xxxxxx 是 Oracle HTTP Server 的实例号。
2. 创建一个新的空 Wallet。
 - a. 在 Oracle Wallet Manager 中，依次选择 **Wallet** 和新建。
 - b. 单击是以创建默认 Wallet 目录，或者单击否以在您选定的位置创建 Wallet 文件。
 - c. 在“新建 Wallet”屏幕上的 **Wallet** 密码和确认密码选项中，输入您要使用的密码。
 - d. 单击确定。
 - e. 在确认对话框中，单击否。
3. 可选：如果没有使用 Oracle HTTP Server 已知的 CA，请将根 CA 证书导入该 Wallet。
 - a. 在 Oracle Wallet Manager 中，右键单击信任证书并选择导入信任证书。
 - b. 浏览并选择根 CA 证书。
 - c. 选择打开。
4. 创建证书请求。
 - a. 在 Oracle Wallet Manager 中，右键单击证书：[空] 并选择添加证书请求。
 - b. 在“创建证书请求”中，输入必需信息。
对于公共名称，输入完全限定的服务器别名（例如 epm.myCompany.com 或 epminternal.myCompany.com），可从系统上的 hosts 文件中找到此信息。
 - c. 单击确定。
 - d. 在确认对话框中，单击确定。
 - e. 右键单击创建的证书请求，然后选择导出证书请求。
 - f. 指定证书请求文件的名称。
5. 使用证书请求文件，从 CA 获得签名证书。
6. 导入签名证书

- a. 在 Oracle Wallet Manager 中，右键单击用于获取签名证书的证书请求，然后选择导入用户证书。
 - b. 在“导入证书”中，单击确定以从文件导入证书。
 - c. 在“导入证书”中，选择“证书文件”，然后单击打开。
7. 将 Wallet 保存到方便的位置；例如，`EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/keystores/epmsystem`。
 8. 选择 **Wallet**，然后选择自动登录以激活自动登录。

使用 ORAPKI 设置 Oracle Wallet (在 Linux 上)

要使用 ORAPKI 命令行设置 Oracle Wallet，请完成以下步骤：

1. 创建用于存放 wallet 的文件夹：

```
$ mkdir /MIDDLEWARE_HOME/oracle_common/wallet
```

2. 将 orapki 实用程序的位置添加到您的路径中：

```
$ export PATH=$PATH:$MIDDLEWARE_HOME/oracle_common/bin
```

3. 创建 wallet 以存放您的证书：

```
>$ MIDDLEWARE_HOME/oracle_common/bin/orapki wallet create -wallet  
[wallet_location] -auto_login
```

此命令提示您输入和重新输入 wallet 密码（如果未在命令行上指定任何密码）。它将在为 `-wallet` 指定的位置创建 wallet。

4. 生成证书签名请求 (certificate signing request, CSR) 并将其添加到您的 wallet：

```
$ MIDDLEWARE_HOME/oracle_common/bin/orapki wallet add -wallet  
[wallet_location] -dn 'CN=<CommonName>,OU=<OrganizationUnit>,  
O=<Company>, L=<Location>, ST=<State>, C=<Country>' -keysize 512|  
1024|2048|4096 -pwd [Wallet_Password]
```

5. 将根证书和中间证书添加到可信的密钥库中

```
$ MIDDLEWARE_HOME/oracle_common/bin/orapki wallet add -wallet  
[wallet_location] -trusted_cert -cert [certificate_location] [-pwd]
```

6. 使用您的 CA (Certificate Authority, 证书颁发机构) 为 CSR (Certificate Signing Request, 证书签名请求) 签名。要从 Oracle Wallet 导出证书请求：

```
$ MIDDLEWARE_HOME/oracle_common/bin/orapki wallet export -wallet  
[wallet_location] -dn 'CN=<CommonName>,OU=<OrganizationUnit>,  
O=<Company>, L=<Location>, ST=<State>, C=<Country>' -request  
[certificate_request_filename] [-pwd]
```

7. 将签名的 CSR 导入到 wallet 中:

```
$ MIDDLEWARE_HOME/oracle_common/bin/orapki wallet add -wallet  
[wallet_location] -user_cert -cert [certificate_location] [-pwd]
```

8. 要显示 wallet 的内容:

```
$ MIDDLEWARE_HOME/oracle_common/bin/orapki wallet display -wallet  
[wallet_location] [-pwd]
```

为 Oracle HTTP Server 启用 SSL

在托管 Oracle HTTP Server 的每台计算机上重新配置 Web 服务器之后,更新 Oracle HTTP Server 配置文件,将其中的默认 Wallet 位置替换为您创建的 Wallet 的位置。

要为 Oracle HTTP Server 配置 SSL:

1. 在部署中的每台 Oracle HTTP Server 主机上重新配置 Web 服务器。
2. 为实例启动 EPM System Configurator。
3. 在配置任务选择屏幕中,完成以下步骤,然后单击下一步。
 - a. 通过取消全选清除所选内容。
 - b. 展开 **Hyperion Foundation** 任务组,然后选择配置 **Web 服务器**。
4. 在配置 **Web 服务器**中,单击下一步。
5. 在确认中,单击下一步。
6. 在摘要中,单击完成。
7. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/fmwconfig/components/OHS/ohs_component/ssl.conf`。
8. 确保使用的 SSL 端口已列在 OHS Listen port 下,类似如下所示:
如果使用 19443 作为 SSL 通信端口,条目内容应如下所示:

```
Listen 19443
```

9. 将 SSLSessionCache 参数值设置为 none。
10. 更新部署中每个 Oracle HTTP Server 的配置设置。
 - a. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/httpConfig//ohs/config/fmwconfig/components/OHS/ohs_component/ssl.conf`。
 - b. 找到 SSLWallet 指令并更改其值,使其指向安装证书所在的 Wallet。如果您已在 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/keystores/epmsystem` 中创建 Wallet,则 SSLWallet 指令可能如下所示:

```
SSLWallet "${ORACLE_INSTANCE}/config/${COMPONENT_TYPE}/${  
COMPONENT_NAME}/keystores/epmsystem"
```

- c. 保存并关闭 `sl.conf`。
11. 更新部署中每个 Oracle HTTP Server 上的 `mod_wl_ohs.conf`。

- a. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/fmwconfig/components/OHS/ohs_component/mod_wl_ohs.conf`。

- b. 确保 `WLSSLWallet` 指令指向存储 SSL 证书的 Oracle Wallet。

```
WLSSLWallet MIDDLEWARE_HOME/ohs/bin/wallets/myWallet
```

例如 `C:/Oracle/Middleware/ohs/bin/wallets/myWallet`

- c. 将 `SecureProxy` 指令的值设置为 `ON`。

```
SecureProxy ON
```

- d. 确保已部署的 Oracle Enterprise Performance Management System 组件的 `LocationMatch` 定义类似以下 Oracle Hyperion Shared Services 示例。该示例假定有一个 Oracle WebLogic Server 群集（位于 `myserver1` 和 `myserver2` 上且使用 SSL 端口 28443）：

```
<LocationMatch /interop/>
  SetHandler weblogic-handler
  pathTrim /
  WeblogicCluster myServer1:28443,myServer2:28443
  WLProxySSL ON
</LocationMatch>
```

- e. 保存并关闭 `mod_wl_ohs.conf`。

配置 WebLogic 服务器上部署的 EPM System Web 组件

部署 Oracle Enterprise Performance Management System Web 组件之后，必须进行相应配置以实现 SSL 通信。

要为 Web 组件配置 SSL：

1. 通过执行存储在 `EPM_ORACLE_INSTANCE/domains/EPMSysystem/bin/startWebLogic.cmd` 中的文件启动 Oracle WebLogic Server：
2. 通过访问以下 URL 启动 WebLogic Server 管理控制台：

```
http://SERVER_NAME:Port/console
```

例如，要访问部署在 `myServer` 默认端口上的 WebLogic Server 控制台，应使用 `http://myServer:7001/console`。

3. 在“欢迎”屏幕上输入用户名和密码，以访问 `EPMSysystem`。该用户名和密码是在配置期间在 EPM System Configurator 中指定的。
4. 在更改中心内，单击锁定并编辑。
5. 在控制台的左窗格中，展开环境，然后选择服务器。
6. 在“服务器摘要”屏幕中，单击您要为其启用 SSL 的服务器的名称。

例如，如果您已安装所有 Oracle Hyperion Foundation Services 组件，则可为以下服务器启用 SSL：

- CalcManager

- FoundationServices
7. 清除已启用侦听端口以禁用 HTTP 侦听端口。
 8. 确保已选择已启用 **SSL** 侦听端口。
 9. 在 **SSL 侦听端口** 中，输入 WebLogic Server SSL 侦听端口。
 10. 指定要使用的标识和信任密钥库。
 - 选择密钥库以打开“密钥库”选项卡。
 - 在密钥库选项卡中，选择一个选项：
 - a. 选择密钥库以打开“密钥库”选项卡。
 - b. 在密钥库选项卡中，选择一个选项：
 - 自定义标识和自定义信任（如果您未使用来自知名第三方 CA 的服务器证书）
 - 自定义标识和 **Java** 标准信任（如果您使用了来自知名第三方 CA 的服务器证书）
 - c. 在自定义标识密钥库中，输入签名的 WebLogic Server 证书安装到的密钥库的路径。
 - d. 在自定义标识密钥库类型中，输入 `jks`。
 - e. 在自定义标识密钥库密码短语和确认自定义标识密钥库密码短语中，输入密钥库密码。
 - f. 如果您已在密钥库中选择自定义标识和自定义信任：
 - 在自定义信任密钥库中，输入为服务器证书签名的 CA 的根证书所在的自定义密钥库的路径。
 - 在自定义信任密钥库类型中，输入 `jks`。
 - 在自定义信任密钥库密码短语和确认自定义信任密钥库密码短语中，输入密钥库密码。
 - g. 单击保存。
 11. 指定 SSL 设置。
 - 选择 **SSL**。
 - 在私钥别名中，输入您在导入已签名的 WebLogic Server 证书时指定的别名。
 - 在私钥密码短语和确认私钥密码短语中，输入用于检索私钥的密码。
 - 仅限 **Oracle Hyperion Provider Services Web 应用程序**：如果您使用通配符证书对 WebLogic Server 和其他 EPM System 服务器组件之间的通信进行加密，则需禁用 Provider Services Web 应用程序的主机名验证。
 - 选择高级。
 - 在主机名验证中，选择无。
 - 单击保存。
 12. 在更改中心内，单击激活更改。

更新域配置

此过程会更新域配置。在启动此过程之前，请创建部署的完整备份。Oracle 建议在对生产部署进行更改之前先在测试部署上测试此过程。

要更新域配置：

1. 导航到 MIDDLEWARE_HOME/oracle_common/bin directory 目录：
cd MIDDLEWARE_HOME/oracle_common/bin

2. 设置 ORACLE_HOME、WL_HOME 和 JAVA_HOME。
set ORACLE_HOME= /Oracle/Middleware

set WL_HOME= /Oracle/Middleware/wlserver

set JAVA_HOME= /Oracle/Middleware/jdk

3. 在 WebLogic 控制台中，为管理服务器启用 HTTP 端口。

4. 使用类似如下的命令创建密钥库：

```
libovdconfig.bat -host HOSTNAME -port 7001 -userName USERNAME -
domainPath %MWH%\user_projects\domains\EPMSysstem -createKeystore
```

在此命令中，将 *HOSTNAME* 替换为 WebLogic 服务器的主机名，将 *USERNAME* 替换为管理员的用户名。确保输出内容报告 OVD 密钥库创建成功。

5. 从 AdminServer 中导出 SSL 证书。

 **Note:**

此步骤仅适用于嵌入式 LDAP（默认身份验证器）。对于其他 LDAP，必须使用特定于 LDAP 的相应命令导出证书。证书格式必须为 **Base 64 编码的 x.509**

- a. 使用 Internet Explorer 通过连接到 <https://HOSTNAME:7002/console> 来访问 WebLogic 管理控制台
 - b. 依次单击查看证书和详细信息，然后选择复制到文件以导出 SSL 证书。
 - c. 将证书以 **Base 64 编码的 x.509** 证书文件形式保存到本地目录，例如 C:\certificate\slc17rby.cer。
 - d. 将证书移到服务器上。
6. 使用 keytool，将证书导入到步骤 4 中创建的密钥库中。使用类似如下的命令（假设 *JAVA_HOME*（和 keytool 可执行文件）在相应的路径中）：

```
export PATH=$JAVA_HOME/bin:$PATH
```

```
keytool -importcert -keystore
DOMAIN_HOME\config\fmwconfig\ovd\default\keystores/adapters.jks -
storepass PASSWORD -alias wcp_ssl -file CERTIFICATE_PATH -noprompt, 例
如:
```

```
keytool -importcert -keystore %MWH%
\user_projects\domains\EPMSysstem\config\fmwconfig\ovd\default\keystore
s/adapters.jks -storepass examplePWD -alias wcp_ssl -file
C:\certificate\slc17rby.cer -noprompt
```

 **Note:**

- 此命令中使用的密码必须与在步骤 4 中生成密钥库时所使用的密码一致。
- `CERTIFICATE_PATH` 是证书的位置和名称
- `alias` 可以是您选择的任何别名。

在成功导入证书后，`keytool` 会显示消息 `Certificate was added to keystore`（证书已添加到密钥库）。

7. 在 WebLogic 控制台中，除了 HTTP 端口外，还为管理服务器启用 SSL 端口。
8. 重新启动 Weblogic 管理服务器和受管服务器。
9. 使用安全连接登录到 Oracle Hyperion Enterprise Performance Management Workspace 以验证一切是否正常。

重新启动服务器和 EPM System

重新启动部署中的所有服务器，然后在每台服务器上启动 Oracle Enterprise Performance Management System。

测试部署

完成 SSL 部署之后，验证一切是否都能正常运行。

要测试部署：

1. 使用浏览器访问安全 Oracle Hyperion Enterprise Performance Management Workspace URL：

如果使用 `epm.myCompany.com` 作为外部通信的服务器别名，并使用 4443 作为 SSL 端口，则 EPM Workspace URL 为

```
https://epm.myCompany.com:4443/workspace/index.jsp
```

2. 在“登录”屏幕上，输入用户名和密码。
3. 单击登录。
4. 验证您能否安全访问部署的 Oracle Enterprise Performance Management System 组件。

配置已启用 SSL 的外部用户目录

假设

- 您打算在 Oracle Hyperion Shared Services Console 中配置的外部用户目录已启用 SSL。
- 如果您没有使用来自知名第三方 CA 的证书为用户目录启用 SSL，则您具有为服务器证书签名的 CA 的根证书副本。

导入根 CA 证书

如果您没有使用来自知名第三方 CA 的证书为用户目录启用 SSL，则您必须将为服务器证书签名的 CA 的根证书导入以下密钥库：

 注:

在应用程序部署期间，WebLogic 将添加 `-Djavax.net.ssl.trustStore` 指令，该指令指向 `setDomainEnv.sh` 或 `setDomainEnv.cmd` 中的 `DemoTrust.jks`。如果您使用的不是默认的 WebLogic 证书，则从 `setDomainEnv.sh` 或 `setDomainEnv.cmd` 中删除 `-Djavax.net.ssl.trustStore`。

使用工具（例如 `keytool`）导入根 CA 证书。

- 所有 Oracle Enterprise Performance Management System 服务器：
JVM 密钥库： `MIDDLEWARE_HOME/jdk/jre/lib/security/cacerts`
- 每个 EPM System 组件主机上的 JVM 使用的密钥库。默认情况下，EPM System 组件使用以下密钥库：

`MIDDLEWARE_HOME/jdk/jre/lib/security/cacerts`

配置外部用户目录

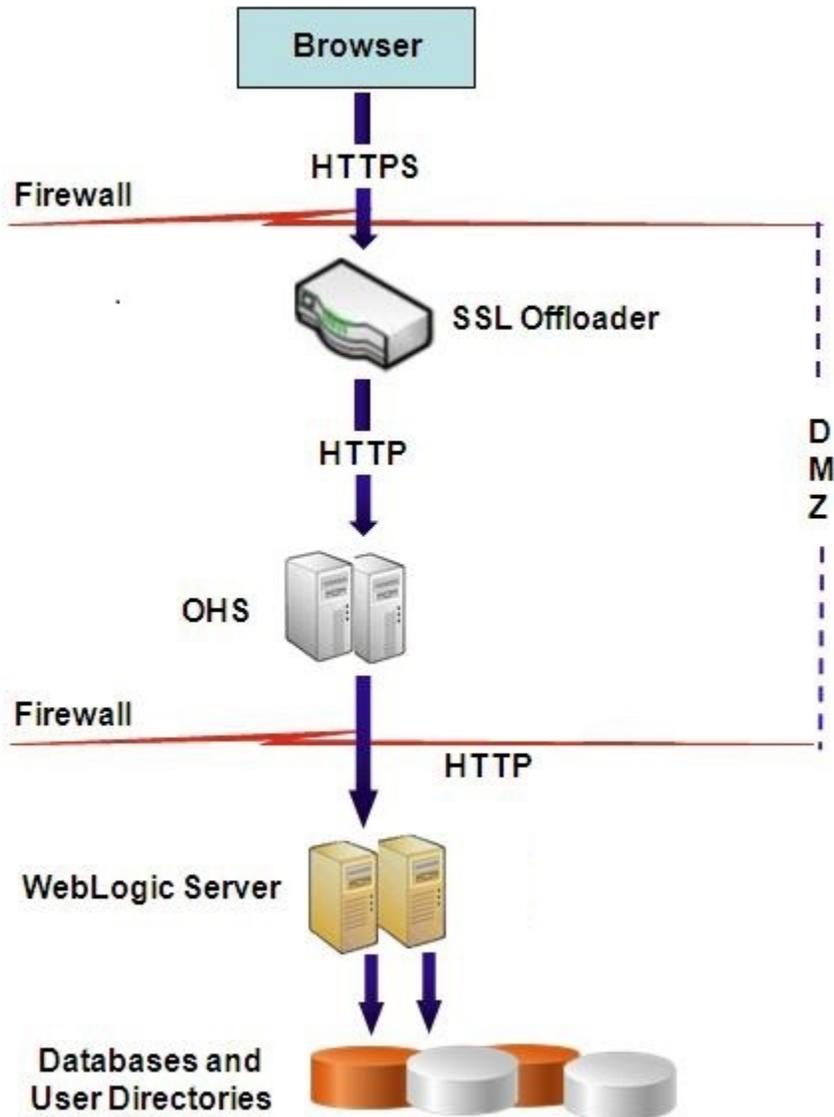
您可通过使用 Shared Services Console 来配置用户目录。配置用户目录时，您必须选择已启用 `SSL` 选项，该选项指示 EPM System 安全性使用安全协议与用户目录通信。您可以为 EPM System 安全性和已启用 LDAP 的用户目录（例如 Oracle Internet Directory 和 Microsoft Active Directory）之间的连接启用 SSL。

请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置用户目录”。

在 Web 服务器上终止 SSL

部署体系结构

在此方案中，使用 SSL 保护 Oracle Enterprise Performance Management System 客户端（例如浏览器）和 Oracle HTTP Server 之间通信链路的安全。概念如下图所示：



假设

此配置在 Web 服务器上使用两个服务器别名（例如 `epm.myCompany.com` 和 `empinternal.myCompany.com`）：一个用于 Web 服务器和浏览器之间的外部通信，另一个用于各 EPM System 服务器之间的内部通信。确保服务器别名指向该计算机的 IP 地址，且可通过 DNS 解析。

Web 服务器（其中定义了支持安全外部通信的虚拟主机）上必须已安装支持使用浏览器（例如，通过 `epm.myCompany.com`）进行外部通信的签名证书。此虚拟主机应终止 SSL 并将 HTTP 请求转发到 Oracle HTTP Server。

当 SSL 在 Oracle HTTP Server (OHS) 或负载均衡器上终止时，您必须：

- 将每个逻辑 Web 应用程序设置为负载均衡器或 Oracle HTTP Server 的非 SSL 虚拟主机（例如 `empinternal.myCompany.com:80`，其中 80 是非 SSL 端口）。打开“配置”屏幕，完成以下步骤：
 1. 展开 **Hyperion Foundation** 配置任务。

2. 选择配置 **Web** 应用程序的逻辑地址。
 3. 指定主机名、非 SSL 端口号和 SSL 端口号。
- 将外部 URL 设置为负载均衡器或 Oracle HTTP Server 的启用 SSL 的虚拟主机（例如，`empexternal.myCompany.com:443`，其中 443 是 SSL 端口）。打开“配置”屏幕，完成以下步骤：
 1. 展开 **Hyperion Foundation** 配置任务。
 2. 选择配置公共设置。
 3. 选择“外部 URL 详细信息”下的启用 **SSL** 卸载。
 4. 指定外部 *URL* 主机和外部 *URL* 端口。

 **注：**

使用 **configtool** 重新部署 Web 应用程序或重新配置 Web 服务器将替换逻辑 Web 应用程序和外部 URL 的设置。

配置 EPM System

EPM System 组件的默认部署支持在 Web 服务器上终止 SSL。无需其他操作。

配置 EPM System 时，确保逻辑 Web 应用程序指向为内部通信创建的虚拟主机（例如 `empinternal.myCompany.com`）。请查看以下信息源以安装和配置 EPM System：

- 《Oracle Enterprise Performance Management System 安装与配置指南》
- 《Oracle Enterprise Performance Management System 安装入门》

测试部署

完成部署过程后，通过连接到安全的 Oracle Hyperion Enterprise Performance Management Workspace URL 验证一切是否正常：

```
https://virtual_host_external:SSL_PORT/workspace/index.jsp
```

例如，`https://epm.myCompany.com:443/workspace/index.jsp`，其中 443 为 SSL 端口。

用于 Essbase 11.1.2.4 的 SSL

概览

本节介绍替换默认证书的过程，这些证书用于保护 Oracle Essbase 实例和组件之间通信，例如以下组件和实例：MaxL、Oracle Essbase Administration Services 服务器、Oracle Essbase Studio 服务器、Oracle Hyperion Provider Services、Oracle Hyperion Foundation Services、Oracle Hyperion Planning、Oracle Hyperion Financial Management 和 Oracle Hyperion Shared Services Registry。

默认部署

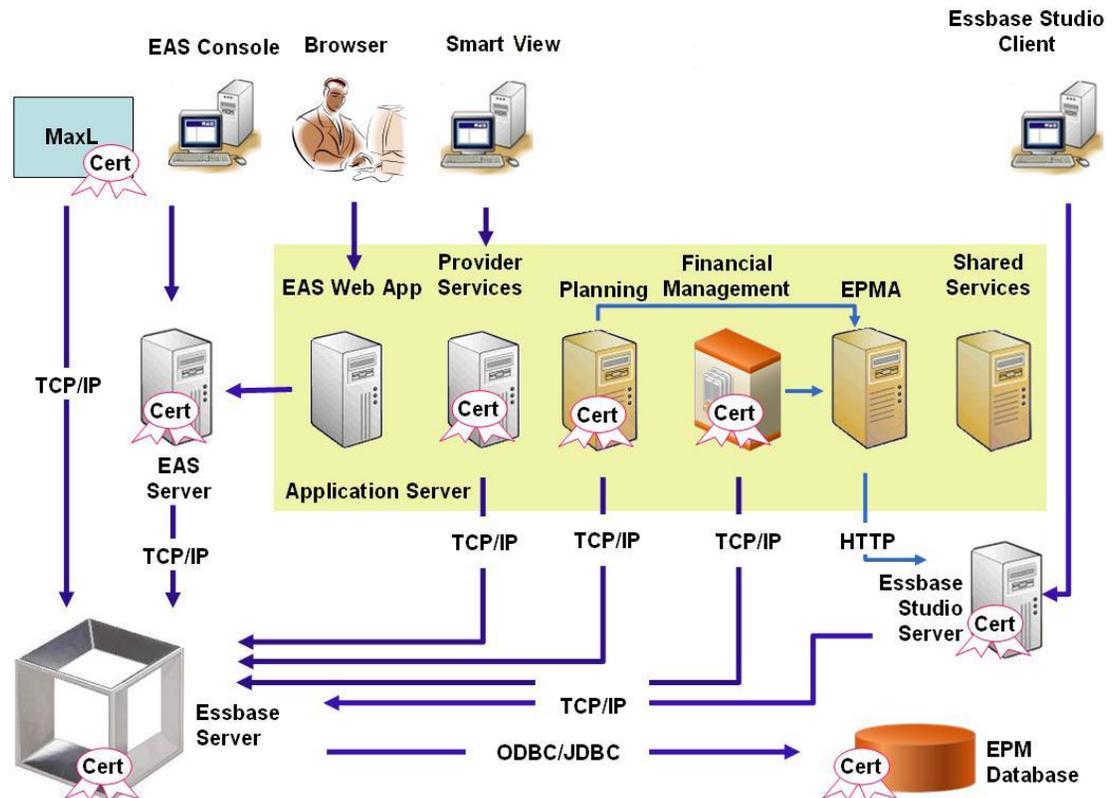
Essbase 部署后可以在 SSL 和非 SSL 模式下工作。Essbase 代理侦听非安全端口；也可以配置为侦听安全端口。访问安全端口的所有连接均被视为 SSL 连接。如果客户端

连接到非 SSL 端口上的 Essbase 代理，则该连接将被视为非 SSL 连接。组件可以与 Essbase 代理建立并发的非 SSL 连接和 SSL 连接。

您可以通过在登录时指定安全协议和端口来控制每个会话的 SSL。请参阅“[建立每个会话的 SSL 连接](#)”。

如果已启用 SSL，则 Essbase 实例内的所有通信均会加密，以确保数据安全性。

Essbase 组件在安全模式下的默认部署使用自签名证书来启用 SSL 通信，主要用于测试目的。Oracle 建议您在生产环境中对已启用 SSL 的 Essbase 使用来自知名第三方的 CA 证书。



通常，用于与使用 Essbase RTC 的客户端进行 SSL 通信的证书存储在 Oracle Wallet 中，而用于与使用 JAPI 通信的组件进行 SSL 通信的证书存储在 Java 密钥库中。为了建立 SSL 通信，Essbase 客户端和工具会存储对 Essbase 服务器和代理证书进行签名的 CA 的根证书。请参阅“[必需的证书及其位置](#)”。

必需的证书及其位置

Oracle 建议您在生产环境中对已启用 SSL 的 Essbase 使用来自知名第三方的 CA 证书。您可以使用默认的自签名证书进行测试。

注：

Essbase 支持使用通配符证书，通过一个 SSL 证书即可保护多个子域。使用通配符证书可以减少管理时间和成本。

如果已启用主机名检查，则不能使用通配符证书。

您需要以下证书：

- 根 CA 证书。
使用 Essbase RTC 与 Essbase 建立连接的组件要求根 CA 证书存储在 Oracle Wallet 中。使用 JAPI 建立连接的组件要求根 CA 证书存储在 Java 密钥库中。下表列出了必需的证书及其位置。

 注：

如果您使用来自某知名第三方 CA 的证书而且已将其根证书安装在 Oracle Wallet 中，则无需安装根 CA 证书。

- 用于 Essbase 服务器和 Essbase 代理的签名证书。

表 2-1 必需的证书及其位置

组件 ¹	密钥库	证书 ²
MaxL	Oracle Wallet	根 CA 证书
Administration Services 服务器	Oracle Wallet	根 CA 证书
Provider Services	Oracle Wallet	根 CA 证书
Oracle Enterprise Performance Management System 数据库	Oracle Wallet	根 CA 证书
Essbase Studio 服务器	Java 密钥库	根 CA 证书
Planning	<ul style="list-style-type: none"> • Oracle Wallet • Java 密钥库 	根 CA 证书
Financial Management	Java 密钥库	根 CA 证书
Essbase (服务器和代理) ³	<ul style="list-style-type: none"> • Oracle Wallet • Java 密钥库 	<ul style="list-style-type: none"> • 根 CA 证书 • 用于 Essbase 服务器和代理的签名证书。
Oracle Hyperion Shared Services 存储库		

¹ 您仅需一个密钥库实例即可支持多个使用类似密钥库的组件。
² 多个组件可以使用已安装到密钥库中的同一个根证书。
³ 证书必须安装在默认的 Oracle Wallet 和 Java 密钥库中。

安装并部署 Essbase 组件

在配置过程中，您可以选择安全代理端口（默认为 6423）；配置 Oracle Essbase 时可以更改该端口。默认情况下，部署过程将安装创建测试功能安全部署所需的自签名证书。

如果已安装 Oracle HTTP Server，则 EPM System 安装程序将在托管 Essbase 实例的计算机上的 `ARBOR_PATH` 内安装 Oracle Wallet 和自签名证书。在单个主机部署中，所有 Essbase 组件均共享此证书。

为 Essbase 使用受信任的第三方 CA 证书

创建证书请求并获取证书

生成证书请求，为托管 Oracle Essbase 服务器和 Essbase 代理的服务器获取证书。证书请求包含特定于您的可分辨名称 (DN) 的加密信息。您向签名机构提交证书请求以获取 SSL 证书。

您可以使用 keytool 或 Oracle Wallet Manager 等工具创建证书请求。有关创建证书请求的详细信息，请参阅您所用工具的相关文档。

如果使用的是 keytool，请使用类似如下的命令创建证书请求：

```
keytool -certreq -alias essbase_ssl -file C:/certs/essbase_server_csr -  
keypass password -storetype jks -keystore  
C:\oracle\Middleware\EPMSys11R1\Essbase_ssl\keystore -storepass password
```

获取和安装根 CA 证书

根 CA 证书验证用于支持 SSL 的证书的有效性。它包含公钥，为证书签名时使用的私钥将与公钥进行匹配以验证证书。您可以从为 SSL 证书签名的证书颁发机构获取根 CA 证书。

在连接到 Essbase 服务器或代理的客户端上，安装为 Essbase 服务器证书签名的 CA 的根证书。确保根证书安装在客户端的相应密钥库中。请参阅[必需的证书及其位置](#)。

注：

多个组件可以使用服务器计算机上安装的同一个根 CA 证书。

Oracle Wallet

有关 Oracle Wallet 中需要 CA 根证书的组件列表，请参阅[必需的证书及其位置](#)。您可以创建 Wallet，或将证书安装在安装了默认自签名证书的演示 Wallet 中。

有关创建 Wallet 和导入根 CA 证书的详细过程，请参阅 Oracle Wallet Manager 文档。

Java 密钥库

有关 Java 密钥库中需要根 CA 证书的组件列表，请参阅[必需的证书及其位置](#)。您可以将证书添加到已安装默认自签名证书的密钥库中，或者创建一个密钥库来存储证书。

注：

许多知名第三方 CA 的根 CA 证书已安装在 Java 密钥库中。

有关详细说明，请参阅您所用工具的相关文档。如果使用的是 keytool，请使用类似如下的命令导入根证书：

```
keytool -import -alias blister_CA -file c:/certs/CA.crt -keypass  
password -trustcacerts -keystore
```

```
C:\Oracle\Middleware\EPMSys11R1\Essbase_ssl  
\keystore -storepass password
```

安装签名证书

在托管 Essbase 服务器和 Essbase 代理的服务器上，安装签名的 SSL 证书。使用 Essbase RTC (C API) 建立与 Essbase 服务器和代理连接的组件要求该证书与根 CA 证书一起存储在 Oracle Wallet 中。使用 JAPI 与 Essbase 服务器或代理建立连接的组件要求 Java 密钥库中存储有根 CA 证书和签名的 SSL 证书。有关详细过程，请参阅以下信息源：

- Oracle Wallet Manager 文档
- 工具文档或联机帮助；例如，用于导入证书的 keytool

如果使用的是 keytool，请使用类似如下的命令导入证书：

```
keytool -import -alias essbase_ssl -file C:/certs/essbase_ssl.crt -  
keypass password -keystore  
C:\Oracle\Middleware\EPMSys11R1\Essbase_ssl\keystore -storepass  
password
```

更新 Essbase 服务器注册表值

Windows

1. 在命令提示符下，将目录转至 `EPM_ORACLE_INSTANCE/epmsystem1/bin`。
2. 运行以下命令来更新 Windows 注册表：

```
epmsys_registry.bat updateproperty "#<Object ID>/@EnableSecureMode"  
true  
  
epmsys_registry.bat updateproperty "#<Object ID>/@EnableClearMode"  
false
```

请务必将 `<Object ID>` 替换为 Essbase 服务器组件 ID，该 ID 存在于您完成 Essbase 服务器配置过程后生成的注册表报表中。

Linux

1. 在控制台中，将目录转至 `EPM_ORACLE_INSTANCE/epmsystem1/bin`。
2. 运行以下命令来更新注册表：

```
epmsys_registry.sh updateproperty "#<Object ID>/@EnableSecureMode"  
true  
  
epmsys_registry.sh updateproperty "#<Object ID>/@EnableClearMode"  
false
```

请务必将 `<Object ID>` 替换为 Essbase 服务器组件 ID，该 ID 存在于您完成 Essbase 服务器配置过程后生成的注册表报表中。

更新 Essbase SSL 设置

要为 Essbase 服务器和客户端自定义 SSL 设置，可以在 `essbase.cfg` 中指定以下项的值。

- 用于启用安全模式的设置
- 用于启用清除模式的设置

- 与客户端通信的首选模式（仅限客户端使用）
- 安全端口
- 密码套件
- Oracle Wallet 路径

 注:

在 `essbase.cfg` 中，确保添加任何缺少的必需参数（具体来说，包括 `EnableSecureMode` 和 `AgentSecurePort`），并设置其值。

要更新 `essbase.cfg`:

1. 为 Essbase 服务器复制带有证书的 Oracle Wallet，将其复制到 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/wallet`。
这是 Essbase 服务器唯一接受的 Oracle Wallet 位置。
2. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg`。
3. 根据需要进行设置。默认的 Essbase 设置为隐式设置。如果需要更改默认行为，请在 `essbase.cfg` 中添加自定义行为的设置。例如，默认情况下强制实施 `EnableClearMode`，以支持 Essbase 服务器通过非加密通道进行通信。要禁止 Essbase 服务器通过未加密通道进行通信，应在 `essbase.cfg` 中指定 `EnableClearMode FALSE`。请参阅下表。

表 2-2 Essbase SSL 设置

设置	描述 ¹
<code>EnableClearMode</code> ²	在 Essbase 应用程序和 Essbase 代理之间启用未加密通信。如果此属性设置为 <code>FALSE</code> ，则 Essbase 不会处理非 SSL 请求。 默认值: <code>EnableClearMode TRUE</code> 示例: <code>EnableClearMode FALSE</code>
<code>EnableSecureMode</code>	在 Essbase 客户端和 Essbase 代理之间启用 SSL 加密通信。此属性必须设置为 <code>TRUE</code> 才能支持 SSL。 默认值: <code>FALSE</code> 示例: <code>EnableSecureMode TRUE</code>
<code>SSLCipherSuites</code>	按优先顺序列出的用于 SSL 通信的密码套件列表。Essbase 代理使用这些密码套件之一进行 SSL 通信。当代理选择密码套件时，列表中的第一个密码套件被赋予最高优先级。 默认值: <code>SSL_RSA_WITH_RC4_128_MD5</code> 示例: <code>SSLCipherSuites SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</code>
<code>APSRESOLVER</code>	Oracle Hyperion Provider Services 的 URL。如果要使用多个 Provider Services 服务器，则使用分号分隔每个 URL。 示例: <code>APSRESOLVER https://exampleAPShost1:PORT/aps;https://exampleAPShost2:PORT/aps</code>

表 2-2 (续) Essbase SSL 设置

设置	描述 ¹
AgentSecurePort	代理侦听的安全端口。 默认值: 6423 示例: AgentSecurePort 16001
WalletPath	存储根 CA 证书和签名证书的 Oracle Wallet 的位置 (少于 1,024 个字符)。 默认值: ARBORPATH/bin/wallet 示例: WalletPath/usr/local/wallet
ClientPreferredMode ³	客户端会话的模式 (“安全”或“清除”)。如果此属性设置为“安全”, 则所有会话均使用 SSL 模式。 如果此属性设置为“清除”, 则根据客户端登录请求是否包含安全传输关键字来选择传输。请参阅 建立每个会话的 SSL 连接 。 默认值: CLEAR 示例: ClientPreferredMode SECURE

¹ 如果 `essbase.cfg` 中未提供这些属性, 则强制实施默认值。
² 如果 `EnableClearMode` 和 `EnableSecureMode` 都设置为 `FALSE`, 则 Essbase 将变为不可操作。
³ 客户端使用此设置来确定是否应与 Essbase 建立安全连接或非安全连接。

4. 保存并关闭 `essbase.cfg`。

更新用于 SSL 的分布式 Essbase 节点



注:
本节仅适用于 Essbase 的分布式部署

确保包含根 CA 证书和签名证书的 Wallet 文件夹 (例如 `WalletPath/usr/local/wallet`) 位于每个分布式节点上的所需位置。

- 将 Wallet 文件夹复制到每个分布式节点中的以下位置:
 - `EPM_ORACLE_HOME/common/EssbaseRTC/11.1.2.0/bin`
 - `EPM_ORACLE_HOME/common/EssbaseRTC-64/11.1.2.0/bin`
- 将 Wallet 文件夹复制到每个分布式节点中的以下位置 (如果存在):
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer-32/bin`
 - `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin`
- 将 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg` 复制到每个分布式节点上的以下位置:
 - `EPM_ORACLE_HOME/common/EssbaseRTC/11.1.2.0/bin`
 - `EPM_ORACLE_HOME/common/EssbaseRTC-64/11.1.2.0/bin`

4. 将 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg` 复制到每个分布式节点上的以下位置（如果存在）：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer-32/bin`
 - `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin`
5. 将 Wallet 文件夹复制到每个分布式节点上的以下 Essbase 客户端安装位置：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient-32/bin`
6. 将 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg` 复制到每个分布式节点上的以下 Essbase 客户端安装位置：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient-32/bin`
7. 将以下属性添加到 `essbase.properties` 文件：
 - `essbase.ssleverywhere=true`
 - `olap.server.ssl.alwaysSecure=true`
 - `APSRESOLVER=http[s]://host:httpsPort/aps`
请务必将此值替换为相应的 URL。

必须更新每个分布式节点中以下位置的 `essbase.properties` 文件（如果存在）：

 - `EPM_ORACLE_HOME/common/EssbaseJavaAPI/11.2.0/bin/essbase.properties`
 - `EPM_ORACLE_HOME/products/Essbase/aps/bin/essbase.properties`
 - `EPM_ORACLE_INSTANCE/aps/bin/essbase.properties`
8. 将 `EPM_ORACLE_HOME/products/Essbase/aps/bin/essbase.properties` 复制到每个分布式节点上的 `EPM_ORACLE_HOME/products/Essbase/eas` 目录（如果可用）。
9. 仅限 **Oracle Hyperion Planning**：将以下三个属性添加到 `essbase.properties` 文件：
 - `essbase.ssleverywhere=true`
 - `olap.server.ssl.alwaysSecure=true`
 - `APSRESOLVER=APS_URL`
将 `APS_URL` 替换为 Provider Services URL。如果要使用多个 Provider Services 服务器，则使用分号分隔每个 URL。例如，`https://exampleAPShost1:PORT/aps;https://exampleAPShost2:PORT/aps`。

必须更新每个分布式节点中以下位置的 `essbase.properties` 文件：

 - `EPM_ORACLE_HOME/products/Planning/config/essbase.properties`
 - `EPM_ORACLE_HOME/products/Planning/lib/essbase.properties`
10. 仅限 **Oracle Hyperion Financial Reporting**：将以下三个属性添加到 `EPM_ORACLE_HOME/products/financialreporting/bin/EssbaseJAPI/bin/essbase.properties` 文件：
 - `essbase.ssleverywhere=true`
 - `olap.server.ssl.alwaysSecure=true`
 - `APSRESOLVER=APS_URL`

将 `APS_URL` 替换为 Provider Services URL。如果要使用多个 Provider Services 服务器，则使用分号分隔每个 URL。例如，`https://exampleAPShost1:PORT/aps;https://exampleAPShost2:PORT/aps`。

 **注：**

在完全 SSL 环境中，Financial Reporting 需要使用 Essbase 群集名称来建立连接。如果将主机名用于连接，连接将失败。

11. a. 设置环境变量：

- **Windows：** 创建名为 `API_DISABLE_PEER_VERIFICATION` 的新系统变量并将其值设置为 1。
- **Linux：** 在 `setCustomParamsPlanning.sh` 中添加指令
`API_DISABLE_PEER_VERIFICATION=1`。

- b.** 在 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/setEssbaseenv.bat` 或 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/setEssbaseenv.sh` 中添加指令
`API_DISABLE_PEER_VERIFICATION=1`。

设置环境变量：

自定义 JAPI 客户端的 SSL 属性。

对于依赖 JAPI 的 Essbase 组件，已预定义多个默认属性。可以通过在 `essbase.properties` 中包含属性来覆盖默认属性。

 **注：**

下表中标识的 SSL 属性中只有少数在 `essbase.properties` 中外置化。您应添加尚未外部化的属性。

要更新 JAPI 客户端的 SSL 属性：

1. 使用文本编辑器打开 `EPM_ORACLE_HOME/common/EssbaseJavaAPI/11.1.2.0/bin/essbase.properties`。
2. 根据需要更新属性。有关可自定义的 JAPI 客户端属性的说明，请参阅下表。如果 `essbase.properties` 中未包含需要的属性，请添加。

表 2-3 JAPI 客户端的默认 SSL 属性

属性	说明
<code>olap.server.ssl.alwaysSecure</code>	设置客户端应对所有 Essbase 实例使用的模式。将此属性值更改为 <code>true</code> ，以强制实施 SSL 模式。 默认值： <code>false</code>
<code>olap.server.ssl.securityHandler</code>	用于处理协议的程序包名称。您可以更改此值以指示另一个处理程序。 默认值： <code>java.protocol.handler.pkgs</code>

表 2-3 (续) JAPI 客户端的默认 SSL 属性

属性	说明
<code>olap.server.ssl.securityProvider</code>	Oracle 使用 Sun SSL 协议实施。您可以更改此值以指示另一个提供程序。 默认值： <code>com.sun.net.ssl.internal.www.protocol</code>
<code>olap.server.ssl.supportedCiphers</code>	可启用以实现安全通信的其他密码的逗号分隔列表。只能指定 Essbase 支持的密码。 示例： <code>SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</code>
<code>olap.server.ssl.trustManagerClass</code>	用于通过验证签名和检查证书到期日期来验证 SSL 证书的 <code>TrustManager</code> 类。 默认情况下，此属性未设置为强制执行所有验证检查。 若不强制执行验证检查，请将此参数的值设置为 <code>com.essbase.services.olap.security.EssDefaultTrustManager</code> ，这是默认 <code>TrustManager</code> 类，允许所有验证检查成功进行。 要实施自定义的 <code>TrustManager</code> ，请指定实施 <code>javax.net.ssl.X509TrustManager</code> 接口的 <code>TrustManager</code> 类的完全限定类名称。 示例： <code>com.essbase.services.olap.security.EssDefaultTrustManager</code>

3. 保存并关闭 `essbase.properties`。
4. 重新启动所有 Essbase 组件。

建立每个会话的 SSL 连接

Oracle Essbase 组件（例如 MaxL）可以在会话级别控制 SSL，方法是使用 `secure` 作为传输关键字以连接到 Essbase 代理。例如，您可以从 MaxL Console 中执行以下命令之一，在 MaxL 和 Essbase 代理之间建立安全连接：

```
login admin example_password on hostA:PORT:secure
```

```
login admin example_password on hostA:secure
```

每个会话控制均优先于 `essbase.cfg` 中指定的配置设置。如果未指定传输关键字，Essbase 客户端将使用为 `ClientPreferredMode` 设置的值来确定是否启动与 Essbase 的安全连接。如果 `ClientPreferredMode` 设置未设置为安全连接，则通信在不安全的通道中进行。

用于 Essbase 21c 的 SSL

概览

本节介绍替换默认证书的过程，这些证书用于保护 Oracle Essbase 实例和组件之间通信，例如以下组件和实例：MaxL、Oracle Essbase Administration Services 服务器、Oracle Hyperion Provider Services、Oracle Hyperion Foundation Services、Oracle Hyperion Planning、Oracle Hyperion Financial Management 和 Oracle Hyperion Shared Services Registry。

注：

Essbase Administration Services (EAS) Lite 不使用通过 EPM Configurator 配置的 HTTP 服务器 SSL 端口（例如 443）。`easconsole.jnlp` 文件中的安全 URL 默认为使用非 SSL 端口（80）。

解决方法：用更新的安全 URL 替换 `easconsole.jnlp` 中标识的安全 URL 中的默认非 SSL 端口：

默认安全 URL：`https://myserver:SECURE_PORT/easconsole/console.html`。例如，`https://myserver:80/easconsole/console.html`

更新的安全 URL：`https://myserver:SECURE_PORT/easconsole/console.html`。例如，`https://myserver:443/easconsole/console.html`

有关详细信息，请参阅 My Oracle Support (MOS) 文章 - [文档 ID 1926558.1 - SSL 端口未包括在 EAS Web 控制台的 easconsole.jnlp 中](#)。

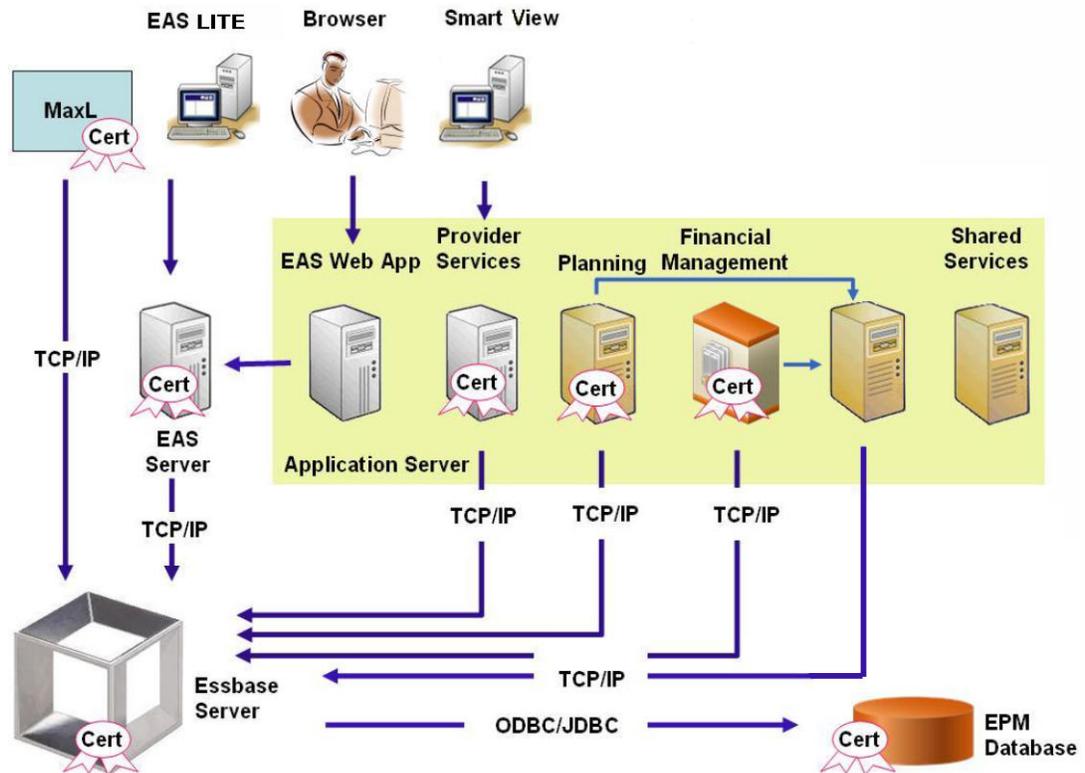
默认部署

Essbase 部署后可以在 SSL 和非 SSL 模式下工作。Essbase 代理侦听非安全端口；也可以配置为侦听安全端口。访问安全端口的所有连接均被视为 SSL 连接。如果客户端连接到非 SSL 端口上的 Essbase 代理，则该连接将被视为非 SSL 连接。组件可以与 Essbase 代理建立并发的非 SSL 连接和 SSL 连接。

您可以通过在登录时指定安全协议和端口来控制每个会话的 SSL。请参阅[“建立每个会话的 SSL 连接”](#)。

如果已启用 SSL，则 Essbase 实例内的所有通信均会加密，以确保数据安全性。

Essbase 组件在安全模式下的默认部署使用自签名证书来启用 SSL 通信，主要用于测试目的。Oracle 建议您在生产环境中对已启用 SSL 的 Essbase 使用来自知名第三方的 CA 证书。



通常，用于与使用 Essbase RTC 的客户端进行 SSL 通信的证书存储在 Oracle Wallet 中，而用于与使用 JAPI 通信的组件进行 SSL 通信的证书存储在 Java 密钥库中。为了建立 SSL 通信，Essbase 客户端和工具会存储对 Essbase 服务器和代理证书进行签名的 CA 的根证书。

必需的证书及其位置

Oracle 建议您在生产环境中对已启用 SSL 的 Essbase 使用来自知名第三方的 CA 证书。您可以使用默认的自签名证书进行测试。

注：

Essbase 支持使用通配符证书，通过一个 SSL 证书即可保护多个子域。使用通配符证书可以减少管理时间和成本。

如果已启用主机名检查，则不能使用通配符证书。

您需要以下证书：

- 根 CA 证书。
使用 Essbase RTC 与 Essbase 建立连接的组件要求根 CA 证书存储在 Oracle Wallet 中。使用 JAPI 建立连接的组件要求根 CA 证书存储在 Java 密钥库中。下表列出了必需的证书及其位置。

 注:

如果您使用来自某知名第三方 CA 的证书而且已将其根证书安装在 Oracle Wallet 中，则无需安装根 CA 证书。

- 用于 Essbase 服务器和 Essbase 代理的签名证书。

表 2-4 必需的证书及其位置

组件 ¹	密钥库	证书 ²
MaxL	Oracle Wallet	根 CA 证书
Administration Services 服务器	Oracle Wallet	根 CA 证书
Provider Services	Oracle Wallet	根 CA 证书
Oracle Enterprise Performance Management System 数据库	Oracle Wallet	根 CA 证书
Planning	<ul style="list-style-type: none"> • Oracle Wallet • Java 密钥库 	根 CA 证书
Financial Management	Java 密钥库	根 CA 证书
Essbase (服务器和代理) ³	<ul style="list-style-type: none"> • Oracle Wallet • Java 密钥库 	<ul style="list-style-type: none"> • 根 CA 证书 • 用于 Essbase 服务器和代理的签名证书。

Oracle Hyperion Shared Services
存储库

¹ 您仅需一个密钥库实例即可支持多个使用类似密钥库的组件。

² 多个组件可以使用已安装到密钥库中的同一个根证书。

³ 证书必须安装在默认的 Oracle Wallet 和 Java 密钥库中。

安装并部署 Essbase 组件

在配置过程中，您可以选择安全代理端口（默认为 6423）；配置 Oracle Essbase 时可以更改该端口。默认情况下，部署过程将安装创建测试功能安全部署所需的自签名证书。

如果已安装 Oracle HTTP Server，则 EPM System 安装程序将在托管 Essbase 实例的计算机上的 `ARBOR_PATH` 内安装 Oracle Wallet 和自签名证书。在单个主机部署中，所有 Essbase 组件均共享此证书。

为 Essbase 使用受信任的第三方 CA 证书

创建证书请求并获取证书

生成证书请求，为托管 Oracle Essbase 服务器和 Essbase 代理的服务器获取证书。证书请求包含特定于您服务器的通用名 (CN=) 的加密信息。您向签名机构提交证书请求以获取 SSL 证书。

您可以使用 `keytool` 或 Oracle Wallet Manager 等工具创建证书请求。有关创建证书请求的详细信息，请参阅您所用工具的相关文档。

keytool 使用示例：

创建 Java 密钥库 (JKS) 并生成私钥：

```
keytool.exe -genkey -dname "cn=myserver, ou=EPM, o=Oracle, c=US"  
-alias essbase_ssl -keypass password -keystore  
C:\oracle\Middleware\EPMSysstem11R1\ssl\EPM.JKS -storepass password  
-validity 365 -keyalg RSA -keysize 2048 -sigalg SHA256withRSA -noprompt
```

生成证书请求：

```
keytool -certreq -alias essbase_ssl -file  
C:\oracle\Middleware\EPMSysstem11R1\ssl\essbase_server.csr -keypass password  
-keystore C:\oracle\Middleware\EPMSysstem11R1\ssl\EPM.JKS -storepass password
```

导出您的私钥（您将需要 openssl 实用程序来执行这些步骤）：

1. openssl.exe pkcs12 -in C:\oracle\Middleware\EPMSysstem11R1\ssl\EPM.JKS -passin pass:password -legacy -nocerts -out c:\Apache24\ssl\Apache24.key -passout pass:password
2. 使用您的 CA（认证机构）为您新生成的证书请求签名，并将其粘贴到以下文件中：
C:\oracle\Middleware\EPMSysstem11R1\ssl\essbase.cer。

获取和安装根 CA 证书

根 CA 证书验证用于支持 SSL 的证书的有效性。它包含公钥，为证书签名时使用的私钥将与公钥进行匹配以验证证书。您可以从为 SSL 证书签名的证书颁发机构获取根 CA 证书。

在连接到 Essbase 服务器或代理的客户端上，安装为 Essbase 服务器证书签名的 CA 的根证书。确保根证书安装在客户端的相应密钥库中。请参阅“必需的证书及其位置”。

注：

多个组件可以使用服务器计算机上安装的同一个人根 CA 证书。

安装 CA 签名证书

要安装 CA 签名证书，请参阅以下链接：

- [为 Essbase 设置 Weblogic TLS 连接](#)
- [更新 TLS 证书](#)

更新以下位置下的 `tls.properties` 文件

```
%EPM_HOME%\essbase\bin\tls_tools.properties:  
certCA=c:\\ssl\\ca.crt;c:\\ssl\\intermediate.crt;c:\\ssl\\essbase.key;c:\\  
\\ssl\\essbase.cer;
```

其中：

```
C:\\ssl\\ca.crt - root CA certificate.  
C:\\ssl\\intermediate.crt - intermediate CA certificate.
```

C:\ssl\essbase.key - your private key generated in the previous step.
C:\ssl\essbase.cer - your server's signed certificate issued by your CA.

执行以下项以使用新证书更新 Essbase 服务器：

```
set ORACLE_HOME=c:\OracleSSL
set EPM_HOME=%ORACLE_HOME%
set WL_HOME=%ORACLE_HOME%\wlserver
set JAVA_HOME=%ORACLE_HOME%\jdk
set DOMAIN_HOME=%ORACLE_HOME%\user_projects\domains\essbase_domain
%EPM_HOME%\essbase\bin\tls_tools.properties:
%ORACLE_HOME%\jdk\bin\java.exe -Xmx256m -jar %ORACLE_HOME%
\essbase\lib\tlsTools.jar %EPM_HOME%\essbase\bin\tls_tools.properties
```

更新 Essbase SSL 设置

要为 Essbase 服务器和客户端自定义 SSL 设置，可以在 `essbase.cfg` 中指定以下项的值。

- 用于启用安全模式的设置
- 用于启用清除模式的设置
- 与客户端通信的首选模式（仅限客户端使用）
- 安全端口
- 密码套件
- Oracle Wallet 路径

注：

在 `essbase.cfg` 中，确保添加任何缺少的必需参数（具体来说，包括 `EnableSecureMode` 和 `AgentSecurePort`），并设置其值。

要更新以下位置下的 `essbase.cfg`：

`ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\essbase`

1. 根据需要输入设置。默认的 Essbase 设置为隐式设置。如果需要更改默认行为，请在 `essbase.cfg` 中添加自定义行为的设置。例如，默认情况下强制实施 `EnableClearMode`，以支持 Essbase 服务器通过非加密通道进行通信。要禁止 Essbase 服务器通过未加密通道进行通信，应在 `essbase.cfg` 中指定 `EnableClearMode FALSE`。请参阅下表：

表 2-5 Essbase SSL 设置

设置	描述 ¹
EnableClearMode ²	在 Essbase 应用程序和 Essbase 代理之间启用未加密通信。如果此属性设置为 FALSE，则 Essbase 不会处理非 SSL 请求。 默认值: EnableClearMode TRUE 示例: EnableClearMode FALSE
EnableSecureMode	在 Essbase 客户端和 Essbase 代理之间启用 SSL 加密通信。此属性必须设置为 TRUE 才能支持 SSL。 默认值: FALSE 示例: EnableSecureMode TRUE
SSLCipherSuites	按优先顺序列出的用于 SSL 通信的密码套件列表。Essbase 代理使用这些密码套件之一进行 SSL 通信。当代理选择密码套件时，列表中的第一个密码套件被赋予最高优先级。 默认值: SSL_RSA_WITH_RC4_128_MD5 示例: SSLCipherSuites SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384
APRESOLVER	Oracle Hyperion Provider Services 的 URL。如果要使用多个 Provider Services 服务器，则使用分号分隔每个 URL。 示例: https://exampleAPShost1:PORT/essbase;https://exampleAPShost2:PORT/essbase
AgentSecurePort	代理侦听的安全端口。 默认值: 6423 示例: AgentSecurePort 16001
WalletPath	存储根 CA 证书和签名证书的 Oracle Wallet 的位置（少于 1,024 个字符）。 默认值: ARBORPATH/bin/wallet 示例: WalletPath/usr/local/wallet
ClientPreferredMode ³	客户端会话的模式（“安全”或“清除”）。如果此属性设置为“安全”，则所有会话均使用 SSL 模式。如果此属性设置为“清除”，则根据客户端登录请求是否包含安全传输关键字来选择传输。请参阅“ 建立每个会话的 SSL 连接 ”。 默认值: CLEAR 示例: ClientPreferredMode SECURE

- ¹ 如果 essbase.cfg 中未提供这些属性，则强制实施默认值。
- ² 如果 EnableClearMode 和 EnableSecureMode 都设置为 FALSE，则 Essbase 将变为不可操作。
- ³ 客户端使用此设置来确定是否应与 Essbase 建立安全连接或非安全连接。

2. 保存并关闭 essbase.cfg。

更新用于 SSL 的分布式 Essbase 节点

注:

本节仅适用于 Essbase 的分布式部署

确保包含根 CA 证书和签名证书的 Wallet 文件夹（例如 `WalletPath/usr/local/wallet`）位于每个分布式节点上的所需位置。

1. 使用 TLS 工具导入所有新 CA 证书。

有关进一步的信息，请参阅以下链接：

- [为 Essbase 设置 Weblogic TLS 连接](#)
- [更新 TLS 证书](#)

2. 转到源位置：`ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\essbase` 并修改 `essbase.properties` 文件中的以下属性：

- `essbase.ssleverywhere=true`
- `olap.server.ssl.alwaysSecure=true`
- `APSRESOLVER=APS_URL`
将 `APS_URL` 替换为 Provider Services URL。如果要使用多个 Provider Services 服务器，则使用分号分隔每个 URL。

```
https://exampleAPShost1:PORT/essbase;https://exampleAPShost2:PORT/essbase。
```

3. 将 `Wallet` 文件夹、`Walletssl` 文件夹、`essbase.cfg` 文件和 `essbase.properties` 文件复制到以下目标路径。

表 2-6 目标路径

目标路径	Wallet	Walletssl	essbase.cfg	essbase.properties
<code>EPM_ORACLE_HOME\common\EssbaseRTC-21c\11.1.2.0\bin</code>	是	是	是	是
<code>EPM_ORACLE_HOME\common\EssbaseJavaAPI-21c\11.1.2.0\bin</code>	是	是	是	是
<code>ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\aps</code>	是	是	是	是
<code>ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\essbase</code>	是	是	是	是
<code>MIDDLEWARE_HOME\essbase\products\Essbase\template_files\essbase</code>	是	是	是	是
<code>MIDDLEWARE_HOME\essbase\products\Essbase\EssbaseServer\bin</code>	是	是	是	是
<code>MIDDLEWARE_HOME\essbase\products\Essbase\aps\bin</code>	是	是	是	是
<code>MIDDLEWARE_HOME\essbase\products\Essbase\ears</code>	是	是	是	是
<code>MIDDLEWARE_HOME\essbase\common\EssbaseJavaAPI\bin</code>	是	是	是	是

表 2-6 (续) 目标路径

目标路径	Walle t	Walle tssl	essb ase.c fg	essbas e. properti es
仅用于 Oracle Hyperion Financial Reporting EPM_ORACLE_HOME/products/ financialreporting/bin/EssbaseJAPI/bin/ 注意：在完全 SSL 环境中，Financial Reporting 需要使用 Essbase 群集名称来建立连接。如果将主机名用于连接，连接将失败。	是	是	是	是
仅用于 Oracle Hyperion Planning EPM_ORACLE_HOME/products/Planning/config/ EPM_ORACLE_HOME/products/Planning/lib/	是	是	是	是

4. 设置环境变量：

- **Windows：**创建名为 API_DISABLE_PEER_VERIFICATION 的新系统变量并将其值设置为 1。
- **Linux：**在 setCustomParamsPlanning.sh 中添加指令
API_DISABLE_PEER_VERIFICATION=1。

自定义 JAPI 客户端的 SSL 属性。

对于依赖 JAPI 的 Essbase 组件，已预定义多个默认属性。可以通过在 `essbase.properties` 中包含属性来覆盖默认属性。

 注：

下表中标识的 SSL 属性中只有少数在 `essbase.properties` 中外外部化。您应添加尚未外部化的属性。

要更新 JAPI 客户端的 SSL 属性：

1. 使用文本编辑器打开 `EPM_ORACLE_HOME/common/EssbaseJavaAPI-21C/11.2.0/bin/essbase.properties`。
2. 根据需要更新属性。有关可自定义的 JAPI 客户端属性的说明，请参阅下表。如果 `essbase.properties` 中未包含需要的属性，请添加。

表 2-7 JAPI 客户端的默认 SSL 属性

属性	说明
<code>olap.server.ssl.alwaysSecure</code>	设置客户端应对所有 Essbase 实例使用的模式。将此属性值更改为 <code>true</code> ，以强制实施 SSL 模式。 默认值： <code>false</code>
<code>olap.server.ssl.securityHandler</code>	用于处理协议的程序包名称。您可以更改此值以指示另一个处理程序。 默认值： <code>java.protocol.handler.pkgs</code>

表 2-7 (续) JAPI 客户端的默认 SSL 属性

属性	说明
<code>olap.server.ssl.securityProvider</code>	Oracle 使用 Sun SSL 协议实施。您可以更改此值以指示另一个提供程序。 默认值: <code>com.sun.net.ssl.internal.www.protocol</code>
<code>olap.server.ssl.supportedCiphers</code>	可启用以实现安全通信的其他密码的逗号分隔列表。只能指定 Essbase 支持的密码。 示例: <code>SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</code>
<code>olap.server.ssl.trustManagerClass</code>	用于通过验证签名和检查证书到期日期来验证 SSL 证书的 TrustManager 类。 默认情况下, 此属性未设置为强制执行所有验证检查。 若不强制执行验证检查, 请将此参数的值设置为 <code>com.essbase.services.olap.security.EssDefaultTrustManager</code> , 这是默认 TrustManager 类, 允许所有验证检查成功进行。 要实施自定义的 TrustManager, 请指定实施 <code>javax.net.ssl.X509TrustManager</code> 接口的 TrustManager 类的完全限定类名称。 示例: <code>com.essbase.services.olap.security.EssDefaultTrustManager</code>

3. 保存并关闭 `essbase.properties`。

4. 重新启动所有 Essbase 组件。

建立每个会话的 SSL 连接

Oracle Essbase 组件 (例如 MaxL) 可以在会话级别控制 SSL, 方法是使用 `secure` 作为传输关键字以连接到 Essbase 代理。例如, 您可以从 MaxL Console 中执行以下命令之一, 在 MaxL 和 Essbase 代理之间建立安全连接:

```
login admin example_password on hostA:PORT:secure
```

```
login admin example_password on hostA:secure
```

每个会话控制均优先于 `essbase.cfg` 中指定的配置设置。如果未指定传输关键字, Essbase 客户端将使用为 `ClientPreferredMode` 设置的值来确定是否启动与 Essbase 的安全连接。如果 `ClientPreferredMode` 设置未设置为安全连接, 则通信在不安全的通道中进行。

3

通过安全代理启用 SSO

另请参阅：

- [支持的 SSO 方法](#)
- [从 Oracle Access Manager 单点登录](#)
- [OracleAS Single Sign-on](#)
- [针对 SSO 保护 EPM System 产品](#)
- [使用身份管理产品配置基于头的 SSO](#)
- [使用 Oracle Identity Cloud Service 为 EPM System 配置基于头的 SSO](#)
- [SiteMinder SSO](#)
- [Kerberos 单点登录](#)
- [针对 SSO 配置 EPM System](#)
- [Smart View 的单点登录选项](#)

支持的 SSO 方法

SSO 要求 Web 标识管理解决方案将已验证身份的用户的登录名传递给 Oracle Enterprise Performance Management System 产品。您可以使用以下标准 EPM System 方法将 EPM System 与基于 Web 的商业和自定义 SSO 解决方案集成。

- [HTTP 头](#)
- [自定义登录类](#)
- [HTTP 授权头](#)
- [从 HTTP 请求获取远程用户](#)
- [使用身份管理产品配置基于头的身份验证](#)

▲ 注意：

作为一项安全措施，如果贵组织使用在头中包含用户标识以实现标识传播的方法，Oracle 建议您在 Web 服务器和应用程序服务器之间实施客户端证书身份验证（双向 SSL）。

HTTP 头

如果使用 Oracle Single Sign-on (OSSO)、SiteMinder 或 Oracle Access Manager 作为 Web 标识管理解决方案，EPM System 安全性将会自动选择自定义 HTTP 头将已验证身份的用户的登录名传递给 EPM System 组件。

EPM System 产品用户的登录名由您在配置 Oracle Hyperion Shared Services 中的用户目录时所指定的登录属性确定。有关登录属性的简短说明，请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。

HTTP 头必须包含设置为登录属性的属性的值。例如，如果登录属性的值为 uid，则 HTTP 头必须包含 uid 属性的值。

有关定义和发出自定义 HTTP 头的详细信息，请参阅 Web 标识管理解决方案文档。

EPM System 安全性将解析 HTTP 头，并根据 Shared Services 上配置的用户目录验证头中包含的登录名。

自定义登录类

当用户登录时，Web 标识管理解决方案依据目录服务器验证用户的身份，并将已验证身份的用户的凭据封装在 SSO 机制中，以便为下游系统启用 SSO。如果 Web 标识管理解决方案使用的机制不受 EPM System 产品的支持，或者 SSO 机制中没有登录属性的值，则可使用自定义登录类派生登录属性的值，并将该值传递给 EPM System 产品。

通过使用自定义登录类，EPM System 可以与使用基于 X509 证书的身份验证的安全代理集成。使用这种身份验证机制时，要求实施标准 Shared Services API 来定义 EPM System 组件和 Web 标识管理解决方案之间的 SSO 接口。自定义登录类必须将 Login Attribute 的值传递到 EPM System 产品。有关登录属性的简短说明，请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。有关样本代码和实施步骤，请参阅[“实施自定义登录类”](#)。

要使用自定义登录类（默认名称为 `com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl`），必须在类路径中实施 `com.hyperion.css.CSSSecurityAgentIF` 接口。`CSSSecurityAgentIF` 定义用于检索用户名和密码（可选）的 `getter` 方法。如果该接口返回 Null 密码，则安全身份验证将提供程序视为受信任，并验证用户是否存在于配置的提供程序中。如果该接口为密码返回非 Null 值，则 EPM System 将尝试使用此实施返回的用户名和密码对请求进行身份验证。

`CSSSecurityAgentIF` 包含两个方法：`getUserName` 和 `getPassword`。

`getUserName` 方法

此方法返回用于身份验证的用户名。

```
java.lang.String getUserName (  
    javax.servlet.http.HttpServletRequest req,  
    javax.servlet.http.HttpServletResponse res)  
    throws java.lang.Exception
```

`req` 参数标识 HTTP 请求，该请求包含用于确定用户名的信息。不使用 `res` 参数（预置以实现向后兼容）。

`getPassword` 方法

此方法返回用于身份验证的明文密码。密码检索是可选的。

```
java.lang.String getPassword(  

```

```
javax.servlet.http.HttpServletRequest req,  
javax.servlet.http.HttpServletResponse res)  
throws java.lang.Exception
```

`req` 参数标识 HTTP 请求，该请求包含用于确定密码的信息。不使用 `res` 参数（预置以实现向后兼容）。

HTTP 授权头

EPM System 安全性支持使用 HTTP 授权头将 Web 标识管理解决方案中登录属性的值传递给 EPM System 产品。EPM System 产品将解析授权头来检索用户的登录名。

从 HTTP 请求获取远程用户

EPM System 安全性支持使用 HTTP 请求将 Web 标识管理解决方案中登录属性的值传递给 EPM System 产品。如果 Web 标识管理解决方案传递了一个包含登录属性值（使用 `setRemoteUser` 函数进行设置）的 HTTP 请求，则使用此 SSO 方法。

使用身份管理产品配置基于头的身份验证

EPM System 支持任何支持基于头的身份验证的身份管理产品，例如 Oracle Identity Cloud Service、Microsoft Azure AD 和 Okta。概念工作流程如下所示：

- 充当反向代理的网关应用程序通过限制未经身份验证的网络访问来保护 EPM System 组件。
- 网关应用程序拦截对 EPM System 组件的 HTTP(S) 请求，并确保身份管理产品在将请求转发到 EPM System 组件之前对用户进行身份验证。
- 在将请求转发到 EPM System 组件时，网关应用程序通过 HTTP 头请求，将经过身份验证的用户的身份传播到 EPM System 组件。

要支持此身份验证方案，应将 EPM System 配置为使用通过 HTTP(S) 头请求传播的经过身份验证的用户身份。

从 Oracle Access Manager 单点登录

Oracle Enterprise Performance Management System 通过接受包含登录属性值的自定义 HTTP 头（默认名称为 `HYPLOGIN`）与 Oracle Access Manager 集成。登录属性是您在 Oracle Hyperion Shared Services 中配置外部用户目录时设置的。有关登录属性的简短说明，请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。

您可以使用任何向 EPM System 提供登录属性值的头名称。在通过 Oracle Access Manager 为 Shared Services 配置 SSO 时需使用头名称。

EPM System 使用登录属性值根据配置的用户目录（在本例中，是 Oracle Access Manager 对用户进行身份验证时所根据的用户目录）对用户进行身份验证，然后生成在整个 EPM System 中启用 SSO 的 EPM SSO 令牌。系统将在 Native Directory 中检查用户的设置信息，以授权用户使用 EPM System 资源。

 注:

Oracle Essbase Administration Services 控制台（胖客户端）不支持从 Oracle Access Manager 进行 SSO。

有关配置 Oracle Access Manager 以及执行诸如设置 HTTP 头和策略域等任务的信息，请参阅 Oracle Access Manager 文档。本指南假定您已在正常运行的 Oracle Access Manager 部署中完成以下任务：

- 为 EPM System 组件设置必需的策略域
- 配置 HTTP 头，以将登录属性值传递给 EPM System
- 保护“[要保护的资源](#)”中列出的 EPM System 资源。Oracle Access Manager 将会质询访问受保护资源的请求。
- 取消保护“[取消保护的资源](#)”中列出的 EPM System 资源。Oracle Access Manager 将不质询访问取消保护资源的请求。

要通过 Oracle Access Manager 为 EPM System 配置 SSO：

1. 将 Oracle Access Manager 用来对用户进行身份验证的用户目录添加为 EPM System 中的外部用户目录。请参阅《*Oracle Enterprise Performance Management System 用户安全管理指南*》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。

 注:

确保选中“连接信息”屏幕中的受信任复选框，以指示该用户目录是受信任的 SSO 源。

2. 为 EPM System 配置 SSO。请参阅“[针对 SSO 配置 EPM System](#)”。

从 **SSO** 提供程序或代理列表中选择 Oracle Access Manager。如果 Oracle Access Manager 中的 HTTP 头使用的名称不是 HYPLOGIN，请在 **SSO** 机制列表旁边的文本框中输入自定义头的名称。

3. 仅限 Oracle Data Relationship Management：

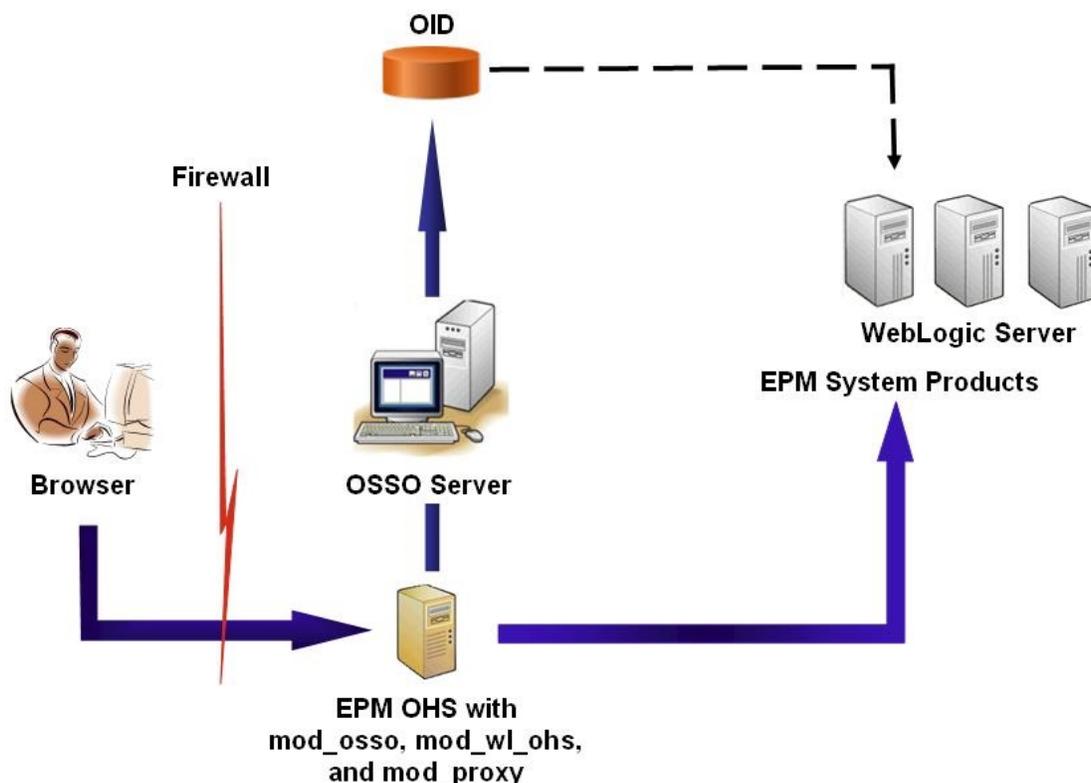
- a. 配置 Data Relationship Management 使用 Shared Services 身份验证。
- b. 在 Data Relationship Management 控制台中启用 SSO。

有关详细信息，请参阅 Data Relationship Management 文档。

OracleAS Single Sign-on

OracleAS Single Sign-on (OSSO) 解决方案提供对使用 Oracle Internet Directory (OID) 作为用户目录的 Web 应用程序的 SSO 访问。用户可使用 OID 中定义的用户名和密码登录到 Oracle Enterprise Performance Management System 产品。

流程流



OSSO 流程:

1. 通过使用 EPM System URL (例如 `http://OSSO_OHS_Server_NAME:OSSO_OHS_Server_PORT/interop/index.jsp`)，用户可访问定义为受 OSSO 保护的应用程序的 EPM System 组件。
2. 由于 URL 受 OSSO 保护，因此 Oracle HTTP Server 上部署的 `mod_osso` 将拦截请求。通过使用 `mod_osso`，Oracle HTTP Server 检查有效的 cookie。如果请求中没有有效的 cookie，Oracle HTTP Server 将用户重定向到 OSSO 服务器，而此服务器会向用户质询凭据，以便根据 OID 对用户进行身份验证。
3. OSSO 服务器创建 `obSSOCookie`，并将控制返回给 Oracle HTTP Server 上的 `mod_osso` 模块，该模块在浏览器中设置 `obSSOCookie`。此外，它还通过 `mod_wl_ohs` (Oracle WebLogic Server) 将请求重定向到 EPM System 资源。将请求转发给 EPM System 资源之前，Oracle HTTP Server 会设置 `Proxy-Remote-User` 头，EPM System 安全性将使用此头启用 SSO。
4. EPM System 组件将验证从 `Proxy-Remote-User` 中检索出标识的用户是否存在于 OID 中。为了使此进程正常运行，使用 OSSO 服务器配置的 OID 应当配置为 Oracle Hyperion Shared Services 中的外部用户目录。

先决条件

1. 功能完整的 Oracle Application Server Infrastructure。

要建立 Oracle Application Server Infrastructure，请安装和配置 Oracle Identity Management Infrastructure 10.1.4。确保已启用 OSSO。Oracle Identity Management Infrastructure 10.1.4 安装包括以下支持 OSSO 的组件。

- Oracle 10g OSSO 服务器。
- OID (OSSO 服务器使用它验证凭据)。请参阅以下指南：

- 《Oracle Fusion Middleware Installation Guide for Oracle Identity Management》
- 《Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory》
- Oracle HTTP Server (用作 OSSO 服务器的前端)。此安装包括 `mod_ossso`，它支持您为 OSSO 定义合作伙伴应用程序。

 注:

此 Oracle HTTP Server 实例是 OSSO 基础结构的一部分；它不直接用于为 EPM System 组件配置 OSSO。

在安装过程中，确保 `mod_ossso` 已作为合作伙伴应用程序注册到 OSSO 服务器。

2. 功能完整的 EPM System 部署。
为 EPM System 组件配置 Web 服务器之后，EPM System Configurator 将在 Oracle HTTP Server 上配置 `mod_wl_ohs.conf`，以通过此代理将请求转发给 WebLogic Server:

测试部署

完成 SSL 部署之后，验证一切是否都能正常运行。

要测试部署：

1. 使用浏览器访问安全 Oracle Hyperion Enterprise Performance Management Workspace URL：
如果使用 `epm.myCompany.com` 作为外部通信的服务器别名，并使用 4443 作为 SSL 端口，则 EPM Workspace URL 为

`https://epm.myCompany.com:4443/workspace/index.jsp`
2. 在“登录”屏幕上，输入用户名和密码。
3. 单击登录。
4. 验证您能否安全访问部署的 Oracle Enterprise Performance Management System 组件。

为 EPM System 启用 OSSO

本节假定您已完全配置 OSSO 基础结构。请参阅《Oracle Application Server 管理员指南》。

将 EPM System Web 服务器注册为合作伙伴应用程序

您可使用 Oracle Identity Manager SSO 注册工具 (`ssoreg.sh` 或 `ssoreg.bat`) 将 Oracle Enterprise Performance Management System Web 服务器注册为作为 OSSO 服务器前端的 Oracle HTTP Server 上的合作伙伴应用程序。

在托管作为 OSSO 服务器前端的 Oracle HTTP Server 的服务器上执行以下过程。此过程将生成经模糊化处理的 `osso.conf` 并将其存储在您选定的位置。

要将 EPM System Web 服务器注册为合作伙伴应用程序：

1. 在托管作为 OSSO 服务器前端的 Oracle HTTP Server 的服务器上，打开控制台并导航到 Oracle HTTP Server 的 `ORACLE_HOME/sso/bin` 目录，例如，`C:/OraHome_1/sso/bin` (Windows)。
2. 执行类似如下带有 `-remote_midtier` 选项的命令：

```
ssoreg.bat -site_name epm.myCompany.com
-mod_osso_url http://epm.myCompany.com:19400
-config_mod_osso TRUE
-update_mode CREATE
-remote_midtier
-config_file C:\OraHome_1\myFiles\osso.conf
```

下面解释了此命令中使用的参数。在此说明中，合作伙伴应用程序是指用作 EPM System Web 服务器的 Oracle HTTP Server。

- `-site_name` 标识合作伙伴的网站；例如，`epm.myCompany.com`。
- `-mod_osso_url` 指示合作伙伴应用程序 URL，格式为 `PROTOCOL://HOST_NAME:PORT`。这是 EPM System Web 服务器用于接受传入客户端请求的 URL；例如，`http://epm.myCompany.com:19000`。
- `-config_mod_osso` 标识合作伙伴应用程序使用 `mod_osso`。您必须包括 `config_mod_osso` 参数以生成 `osso.conf`。
- `-update_mode` 指示更新模式。默认情况下，使用 `CREATE` 以生成新记录。
- `-remote_midtier` 指定 `mod_osso` 合作伙伴应用程序位于远程中端。当合作伙伴应用程序位于与 OSSO 服务器不同的 `ORACLE_HOME` 时，请使用此选项。
- `-virtualhost` 指示合作伙伴应用程序 URL 是虚拟主机。如果您使用的不是虚拟主机，请勿使用此参数。如果您要注册与虚拟主机关联的合作伙伴应用程序 URL，则必须在 `httpd.conf` 中定义该虚拟主机。请参阅“[可选：定义虚拟主机](#)”。
- `-config_file` 指示要在其中生成 `osso.conf` 文件的路径。

可选：定义虚拟主机

如果您在注册合作伙伴应用程序时使用的是虚拟主机 URL，则必须更新用作 EPM System Web 服务器的 Oracle HTTP Server 上的 `httpd.conf` 以定义该虚拟主机。

要定义虚拟主机：

1. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/httpd.conf`。
2. 添加类似下面的命令。此定义假定 Web 服务器在虚拟服务器 `epm.myCompany.com` 的 `epm.myCompany.com:19400` 端口上运行。请根据您的特定需求修改以下设置。

```
NameVirtualHost epm.myCompany.com:19400
Listen 19400
<VirtualHost epm.myCompany.com:19400>
DocumentRoot "C:/Oracle/Middleware/user_projects/epmsystem1/httpConfig/ohs
```

```

        /config/OHS/ohs_component/private-docs"
        include "${ORACLE_INSTANCE}/config/${COMPONENT_TYPE}
        /${COMPONENT_NAME}/mod_osso.conf"
    </VirtualHost>

```

创建 mod_osso.conf

在作为 EPM System Web 服务器前端的 Oracle HTTP Server 上创建 mod_osso.conf。

要创建 mod_osso.conf:

1. 使用文本编辑器新建一个文件。
2. 将以下内容复制到文件中并根据您的环境进行修改。

```

LoadModule osso_module C:/Oracle/Middleware/ohs/ohs/modules/
mod_osso.so
<IfModule mod_osso.c>
    OsoIpCheck off
    OsoIdleTimeout off
    OsoSecureCookies off
    OsoConfigFile C:/Oracle/Middleware/user_projects/epmsystem1/
httpConfig/
        ohs/config/OHS/ohs_component/osso/osso.conf

```

3. 在 <IfModule mod_osso.c 定义中, 包括与以下类似的位置定义, 以标识您计划使用 OSSO 保护的每个资源。

```

        <Location /interop/>
            require valid user
            AuthType Oso
        </Location>
    </IfModule>

```

4. 将文件另存为 mod_osso.conf。

重新放置 osso.conf

在将 EPM System Web 服务器注册为合作伙伴应用程序 (请参阅[“将 EPM System Web 服务器注册为合作伙伴应用程序”](#)) 的过程中, 会在 -config_file 指令标识的位置创建一个经模糊化处理的 osso.conf 文件。

要重新放置 osso.conf:

1. 找到您在将 EPM System Web 服务器注册为合作伙伴应用程序 (请参阅[“将 EPM System Web 服务器注册为合作伙伴应用程序”](#)) 时创建的 osso.conf。
2. 将 osso.conf 复制到 mod_osso.conf 中定义的 OsoConfigFile 属性所标识的目录中 (位于作为 OSSO 服务器前端的 Oracle HTTP Server 上) (请参阅[“创建 mod_osso.conf”](#))。

为 EPM System 配置 OSSO

将与 OSSO 解决方案集成的 OID 配置为 EPM System 中的外部用户目录, 然后启用 SSO。

要为 EPM System 配置 OSSO:

1. 将 OSSO 解决方案使用的 OID 配置为外部用户目录。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。
2. 在 EPM System 中启用 SSO。请参阅[“针对 SSO 配置 EPM System”](#)。

 **注：**

要将 OSSO 配置为标识管理解决方案，您必须在 **SSO** 提供程序或代理中选择其他，在 **SSO** 机制中选择自定义 HTTP 头，并输入 Proxy-Remote-User 作为自定义 HTTP 头的名称。

3. 至少将一个 OID 用户设置为 Oracle Hyperion Shared Services 管理员。
4. 重启 EPM System 产品和使用 Shared Services 安全 API 的自定义应用程序。

 **注：**

在启动 EPM System 产品前，确保随 Shared Services 配置的 OID 正在运行。

可选：在 OSSO 服务器上启用调试消息

要在 OSSO 服务器上记录调试消息，请修改 `policy.properties`。调试消息会写入 `ORACLE_HOME/sso/log/ssoServer.log`。

要记录调试消息：

1. 使用文本编辑器打开 OSSO 服务器上的 `ORACLE_HOME/sso/conf/policy.properties`；例如，`C:\OraHome_1\sso\conf\policy.properties`。
2. 将 `debugLevel` 属性的值设置为 `DEBUG`。

```
debugLevel = DEBUG
```

3. 保存并关闭 `policy.properties`。

可选：对受保护的资源启用调试消息

要记录使用 `mod_osso.conf` 进行保护的资源的 OSSO 调试消息，请修改 EPM System Web 服务器上的 `httpd.conf`。调试消息将写入 `EPM_ORACLE_INSTANCE/httpConfig/ohs/diagnostics/logs/OHS/ohs_component/ohs_component.log`。

要记录受保护资源的调试消息：

1. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/httpd.conf`。
2. 将 `OraLogSeverity` 属性的值设置为 `TRACE`。

```
OraLogSeverity TRACE:32
```

3. 保存并关闭 `httpd.conf`。

针对 SSO 保护 EPM System 产品

您必须保护 Oracle Enterprise Performance Management System 资源，以便将来自用户的 SSO 请求重定向到安全代理（OAM、OSSO 或 SiteMinder）。

Oracle HTTP Server 使用 `mod_ossso` 将用户重定向到 OSSO 服务器。只有在要保护的 `mod_ossso` 中配置了用户请求的 URL 时，才会重定向用户。请参阅《Oracle HTTP Server Administrator's Guide》中的 "[Managing Security](#)"。

有关针对 SiteMinder SSO 保护资源的信息，请参阅 SiteMinder 文档。

仅 OAM：防止将默认头添加到响应中

默认情况下，OAM 将两个头添加到受保护的 URL；Pragma: no-cache 和 Cache-Control: no-cache。因为这些头与 EPM System 和 Web 应用程序添加的类似缓存指令冲突，浏览器可能无法缓存受保护 URL 的内容，从而导致性能降低。

有关防止将这些 OAM 头添加到响应的详细信息，请参阅《Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service》的 "[Oracle Access Management Performance Tuning](#)" 一节中的 "[Tuning OAM Agents](#)"。

要保护的资源

下表列出了必须保护的上下文。针对 OSSO 保护资源的语法如下所示（以 `interop` 为例）：

```
<Location /interop>
Require valid-user
AuthType Basic
order deny,allow
deny from all
allow from myServer.myCompany.com
satisfy any
</Location>
```

`allow from` 参数指定可从中绕过上下文保护的服务器。

对于 Oracle Hyperion Enterprise Performance Management Workspace 和 Oracle Hyperion Financial Reporting，您仅需设置以下示例中指示的参数：

```
<Location /workspace>
Require valid-user
AuthType Basic
</Location>
```

表 3-1 要保护的 EPM System 资源

EPM System 产品	要保护的上下文
Oracle Hyperion Shared Services	<ul style="list-style-type: none">/interop/interop/.../*

表 3-1 (续) 要保护的 EPM System 资源

EPM System 产品	要保护的上下文
EPM Workspace	<ul style="list-style-type: none"> • /workspace • /workspace/.../*
Financial Reporting	<ul style="list-style-type: none"> • /hr • /hr/.../*
Oracle Hyperion Planning	<ul style="list-style-type: none"> • /HyperionPlanning • /HyperionPlanning/.../*
Oracle Integrated Operational Planning	<ul style="list-style-type: none"> • /interlace • /interlace/.../*
Oracle Hyperion Financial Management	<ul style="list-style-type: none"> • /hfmadf • /hfmadfe/.../* • /hfmoofficeprovider • /hfmoofficeprovider/.../* • /hfmsmartviewprovider • /hfmsmartviewprovider/.../*
Oracle Hyperion Financial Reporting Web Studio	/frdesigner/**
Oracle Data Relationship Management	<ul style="list-style-type: none"> • /drm-web-client • /drm-web-client/.../*
Oracle Essbase Administration Services	<ul style="list-style-type: none"> • /hbrlauncher • /hbrlauncher/.../*
Oracle Hyperion Financial Data Quality Management	<ul style="list-style-type: none"> • /HyperionFDM • /HyperionFDM/.../*
Oracle Hyperion Calculation Manager	<ul style="list-style-type: none"> • /calcmgr • /calcmgr/.../*
Oracle Hyperion Provider Services	<ul style="list-style-type: none"> • /aps • /aps/.../*
Oracle Hyperion Profitability and Cost Management	<ul style="list-style-type: none"> • /profitability • /profitability/.../*
Account Reconciliation Manager	<ul style="list-style-type: none"> • /arm • /arm/.../*
Oracle Hyperion Financial Close Management	<ul style="list-style-type: none"> • /fcc • /fcc/.../*
Oracle Hyperion Financial Data Quality Management, Enterprise Edition	<ul style="list-style-type: none"> • /aif • /aif/.../*
Oracle Hyperion Tax Governance Tax Operations	/tss /taxop
Oracle Hyperion Tax Provision Supplemental Data Manager	/taxprov <ul style="list-style-type: none"> • /sdm* • /sdm/** • /sdm/./** • /SDM-Datamodel-context-root/**

表 3-1 (续) 要保护的 EPM System 资源

EPM System 产品	要保护的上下文
Oracle Essbase	<ul style="list-style-type: none"> • /essbase/.../* • /essbase/** • /essbase*

取消保护的资源

下表列出了必须取消保护的上下文。用于针对 OSSO 取消保护资源（使用 /interop/framework(.*) 作为范例）的语法如下所示：

```
<LocationMatch /interop/framework(.*)>
  Require valid-user
  AuthType Basic
  allow from all
  satisfy any
</LocationMatch>
```

表 3-2 要取消保护的 EPM System 资源

EPM System 产品	取消保护的上下文
Shared Services	<ul style="list-style-type: none"> • /interop/framework • /interop/framework* • /interop/framework.* • /interop/framework/.../* • /interop/Audit • /interop/Audit* • /interop/Audit.* • /interop/Audit/.../* • /interop/taskflow • /interop/taskflow* • /interop/taskflow/.../* • /interop/WorkflowEngine • /interop/WorkflowEngine/* • /interop/WorkflowEngine/.../* • /interop/TaskReceiver • /framework/lcm/HSSMigration
EPM Workspace	<ul style="list-style-type: none"> • /epmstatic/.../* • /workspace/bpmstatic/.../* • /workspace/static/.../* • /workspace/cache/.../*

表 3-2 (续) 要取消保护的 EPM System 资源

EPM System 产品	取消保护的上下文
Planning	<ul style="list-style-type: none"> • /HyperionPlanning/Smartview • /HyperionPlanning/faces/PlanningCentral • /HyperionPlanning/servlet/HspDataTransfer • /HyperionPlanning/servlet/HspLCMServlet • /HyperionPlanning/servlet/HspADMServlet/.../* • /HyperionPlanning/servlet/HspADMServlet/** • /HyperionPlanning/servlet/HspADMServlet* • /HyperionPlanning/servlet/HspAppManagerServlet/.../* • /HyperionPlanning/servlet/HspAppManagerServlet/** • /HyperionPlanning/servlet/HspAppManagerServlet*
Financial Reporting	<ul style="list-style-type: none"> • /hr/common/HRLogon.jsp • /hr/services • /hr/services/* • /hr/services/.../* • /hr/modules/com/hyperion/reporting/web/reportViewer/HRStaticReport.jsp • /hr/modules/com/hyperion/reporting/web/repository/HRObjectListXML.jsp • /hr/modules/com/hyperion/reporting/web/reportViewer/HRHtmlReport.jsp • /hr/modules/com/hyperion/reporting/web/bookViewer/HRBookTOCFns.jsp • /hr/modules/com/hyperion/reporting/web/bookViewer/HRBookPdf.jsp
Data Relationship Management	/drm-migration-client
Calculation Manager	<ul style="list-style-type: none"> • /calcmgr/importexport.postExport.do • /calcmgr/common.performAction.do • /calcmgr/lcm.performAction.do* • /calcmgr/lcm.performAction.do/*
Administration Services	<ul style="list-style-type: none"> • /eas • /easconsole • /easdocs
Financial Management	<ul style="list-style-type: none"> • /hfm/EIE/EIListener.asp • /hfmapplicationsservice • /oracle-epm-fm-webservices • /hfm lcm service

表 3-2 (续) 要取消保护的 EPM System 资源

EPM System 产品	取消保护的上下文
Financial Close Management	<ul style="list-style-type: none"> • /FCC-DataModel-context-root • /oracle-epm-erpi-webservices/* • /ARM-DataModel-context-root • /oracle-epm-erpi-webservices/** • /arm/batch/armbatchexecutionservlet • /ARM-DataModel-context-root
Integrated Operational Planning	<ul style="list-style-type: none"> • /interlace/services/ • /interlace/services/* • /interlace/services/* • /interlace/services/.../* • /interlace/anteros • /interlace/anteros/* • /interlace/anteros/* • /interlace/anteros/.../* • /interlace/interlace • /interlace/interlace/* • /interlace/interlace/* • /interlace/interlace/.../* • /interlace/WebHelp • /interlace/WebHelp/* • /interlace/WebHelp/* • /interlace/WebHelp/.../* • /interlace/html • /interlace/html/* • /interlace/html/* • /interlace/html/.../* • /interlace/email-book • /interlace/email-book/* • /interlace/email-book/* • /interlace/email-book/.../*
Profitability and Cost Management	<ul style="list-style-type: none"> • /profitability/cesagent • /profitability/lcm • /profitability/control • /profitability/ApplicationListener • /profitability/HPMApplicationListener
Oracle Essbase	<ul style="list-style-type: none"> • /essbase/agent/.../* • /essbase/jet/logout.html • /essbase/jet/.+\. (js css gif jpe?g png)\$
FDMEE	<ul style="list-style-type: none"> • /aif/services/FDMRuleService • /aif/services/RuleService • /aif/LCMServlet

使用身份管理产品配置基于头的 SSO

先决条件

- 完全配置的 Oracle Enterprise Performance Management System。身份管理产品的目录服务器必须在 EPM System 中配置为用户目录，才能对用户进行授权。
- 完全配置的身份管理产品（Microsoft Azure AD、Okta 等），支持基于头的身份验证。

使用兼容的身份管理产品为 EPM System 配置基于头的 SSO 时涉及以下常规过程。由于所涉及的具体步骤取决于您使用的产品，因此请查阅您的身份管理产品手册以了解详细步骤。

有关使用 Oracle Identity Cloud Service 配置基于头的身份验证的详细步骤，请参阅“[使用 Oracle Identity Cloud Service 为 EPM System 配置基于头的 SSO](#)”。

1. 将 EPM System 注册为身份管理产品中的企业应用程序。此步骤允许身份管理管理员在企业应用程序上配置身份验证，包括支持的功能，如多因素身份验证。
使用附加了 workspace/index.jsp 的网关完全限定域名 (Fully-Qualified Domain Name, FQDN) (例如，https://gateway.server.example.com:443/workspace/index.jsp) 作为 EPM System 的企业应用程序 URL。

配置 EPM System 企业应用程序以传播 HTTP 头。

您可以选择将任何未保留的头名称作为 HTTP 头的名称。头的值应该是唯一标识 EPM System 用户的属性。

2. 安装、配置和注册应用程序网关以确保企业应用程序仅将经过身份验证的请求转发到 EPM System。

使用以下配置设置：

- 网关服务器的 FQDN (例如，gateway.server.example.com:443) 作为接入点。
- EPM System 的 FQDN (例如，epm.server.example.com:443) 作为经过身份验证的 HTTP(S) 请求应转发到的资源。

3. 在 EPM System 中启用 SSO，以支持应用程序网关传播的 HTTP(S) 头。有关详细信息，请参阅“[设置安全选项](#)”。

要启用 SSO：

- a. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。
- b. 选择管理，然后选择配置用户目录。
- c. 单击安全选项。
- d. 在单点登录配置部分中：
 - i. 选中启用 SSO 复选框。
 - ii. 从 SSO 提供程序或安全代理下拉列表中，选择其他。
 - iii. 从 SSO 机制下拉列表中，选择自定义 HTTP 头，然后指定安全代理传递到 EPM System 的头名称。

- e. 单击确定。

4. 将 Oracle Hyperion Enterprise Performance Management Workspace 的“注销后的 URL”设置更新为您希望用户在注销 EPM System 时看到的网页 URL。

要更新 EPM Workspace 中的“注销后的 URL”设置：

- a. 以系统管理员身份访问 EPM Workspace。请参阅“[访问 EPM Workspace](#)”。

- b. 依次选择导航、**Workspace** 设置和服务器设置。
 - c. 在 **Workspace** 服务器设置中，将注销后的 **URL** 更改为您希望用户在注销 EPM System 时看到的网页 URL。
 - d. 单击确定。
5. 重新启动 Oracle Hyperion Foundation Services 和所有 EPM System 受管服务器。

使用 Oracle Identity Cloud Service 为 EPM System 配置基于头的 SSO

在此方案中，Oracle Identity Cloud Service 对 Oracle Enterprise Performance Management System 用户进行身份验证并传播必需的 HTTP 头以启用 SSO。

本节讨论设置和配置 EPM System 以通过 Oracle Identity Cloud Service 支持 SSO 所涉及的步骤。您可以外推这些步骤，以通过支持基于头的身份验证的任何身份管理系统（例如，Azure AD）或基础结构即服务 (Infrastructure as a Service, IaaS) 提供程序来支持 EPM System 的基于头的身份验证。

概念工作流程如下所示：

- 充当反向代理的网关应用程序通过限制未经身份验证的网络访问来保护 EPM System 组件。
- 网关应用程序拦截对 EPM System 组件的 HTTP(S) 请求，并确保身份管理产品在将请求转发到 EPM System 组件之前对用户进行身份验证。
- 在将请求转发到 EPM System 组件时，网关应用程序通过 HTTP 头请求，将经过身份验证的用户的身份传播到 EPM System 组件。

先决条件和示例 URL

要使用 Oracle Identity Cloud Service 建立基于头的 SSO：

- 完全配置的 Oracle Enterprise Performance Management System。
- 具有完全配置的 Oracle App Gateway 的主机或容器，充当反向代理，通过限制未经授权的访问来保护 EPM System。
Oracle App Gateway 应配置为拦截对 EPM System 组件的 HTTP 请求，并确保在将请求转发到 EPM System 之前，Oracle Identity Cloud Service 已对用户进行身份验证。在将请求转发到 EPM System 组件时，Oracle App Gateway 应通过 HTTP 头请求传播经过身份验证的用户的身份。
- 域管理员对 Oracle Identity Cloud Service 的访问权限。

此讨论中使用以下示例 URL：

- Oracle Identity Cloud Service 服务器（身份提供程序）的完全限定域名 (Fully Qualified Domain Name, FQDN) 基本 URL：
`https://identity.server.example.com:443/`
- Oracle App Gateway 服务器（托管网关应用程序）的 FQDN：
`https://gateway.server.example.com:443/`
- EPM System 的企业应用程序 URL。这是附加了 `workspace/index.jsp` 的 Oracle App Gateway 服务器的 FQDN：
`https://gateway.server.example.com:443/workspace/index.jsp`

 **Note:**

为 Oracle Identity Cloud Service 和 Oracle App Gateway 配置了 HTTPS 支持。
EPM System 的 HTTPS 支持是可选的。
此讨论假设已为 EPM System 配置了 HTTPS 支持。

为 EPM System 启用基于头的身份验证

为 Oracle Enterprise Performance Management System 启用基于头的身份验证涉及以下步骤：

- 将 EPM System 应用程序和网关添加到 Oracle Identity Cloud Service
- 配置应用程序网关
- 配置用户目录以进行授权
- 在 EPM System 中启用 SSO
- 更新 EPM Workspace 设置

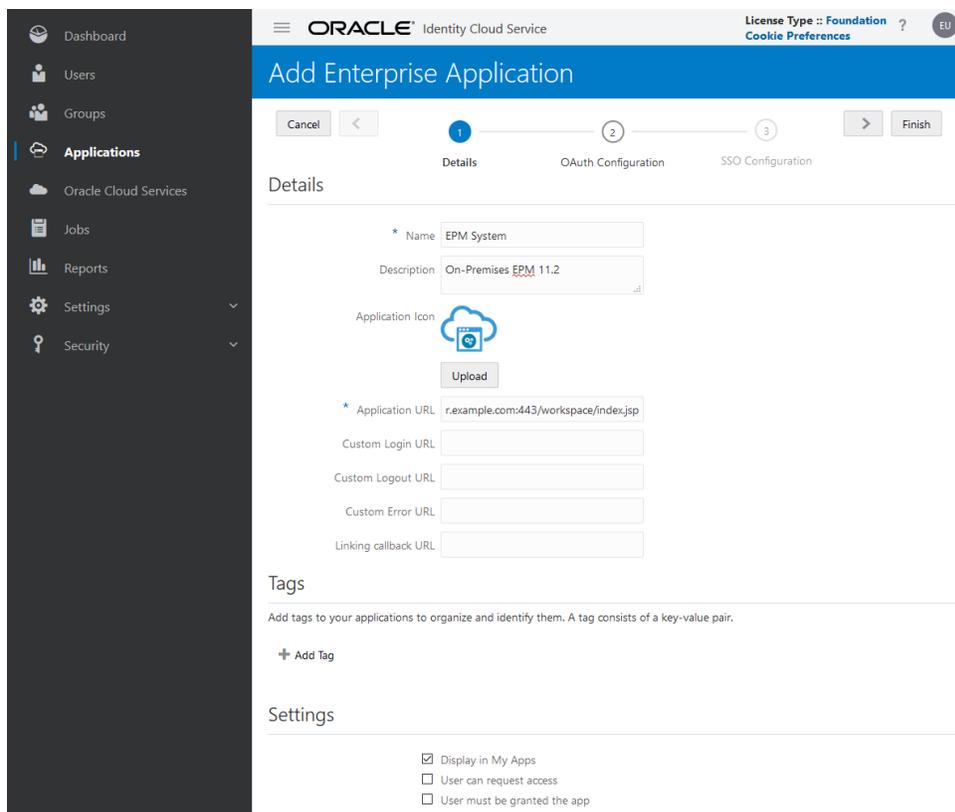
将 EPM System 应用程序和网关添加到 Oracle Identity Cloud Service

要设置基于头的身份验证，需要将 Oracle Enterprise Performance Management System 创建为企业应用程序。

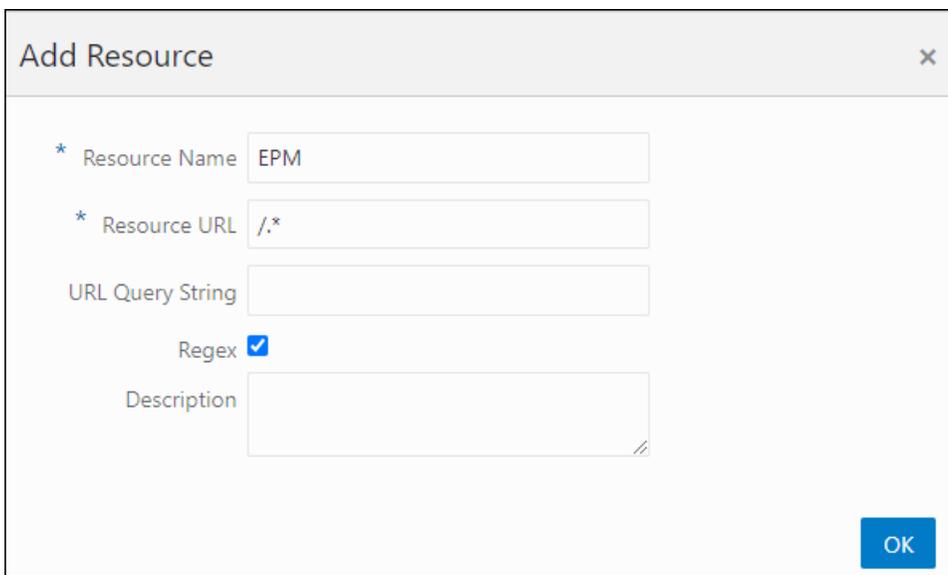
在 Oracle Cloud Identity Console 中将 EPM System 添加为企业应用程序

要将 EPM System 添加为企业应用程序：

1. 以域管理员身份访问 Oracle Cloud Identity Console。
 - a. 使用浏览器，转至 <https://www.oracle.com/cloud/sign-in.html>。
 - b. 输入 Oracle Fusion Cloud EPM 帐户名称。
 - c. 在 Oracle Fusion Cloud EPM 帐户登录页面中，输入您的用户名和密码，然后单击登录。
 - d. 在导航抽屉中，依次单击用户和身份 (主要)。
 - e. 单击身份控制台。
2. 将 EPM System 添加为企业应用程序。
 - a. 在导航抽屉中，单击应用程序。
 - b. 单击添加，然后单击企业应用程序。



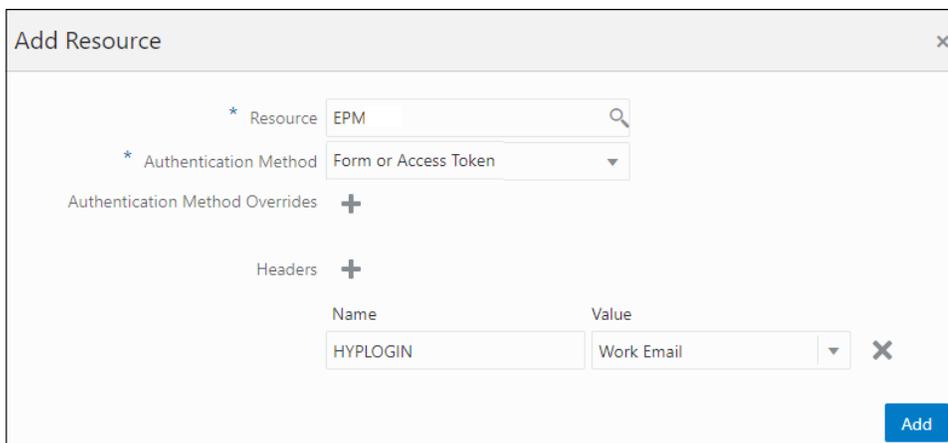
3. 添加应用程序详细信息：
 - a. 在名称中，输入用于标识 EPM System 企业应用程序的唯一名称。
 - b. 根据需要输入说明。
 - c. （可选）上传 EPM System 的应用程序图标。单击上传以选择和上传该图标。
 - d. 在应用程序 URL 中，输入网关应将用户重定向到的启动 URL。此 URL 是 Oracle App Gateway 的 FQDN，附加了 workspace/index.jsp，它是 EPM System 应用程序上下文。
 - e. 在设置下，选择在“我的应用程序”中显示以将 EPM System 企业应用程序显示在 Oracle Cloud Identity Console 中我的应用程序页面的 SSO 配置选项卡上。
 - f. 单击下一步。
4. 指定 SSO 配置详细信息。
 - a. 单击 SSO 配置。
 - b. 为企业应用程序添加资源。
在 SSO 配置中，展开资源。
 - i. 单击添加。



The screenshot shows a dialog box titled "Add Resource" with a close button (X) in the top right corner. It contains the following fields and controls:

- * Resource Name:** A text input field containing "EPM".
- * Resource URL:** A text input field containing "/.*".
- URL Query String:** An empty text input field.
- Regex:** A checked checkbox.
- Description:** A text area with a small icon in the bottom right corner.
- OK:** A blue button at the bottom right.

- ii. 指定唯一的资源名称。
 - iii. 在资源 **URL** 中，输入 `/.*`。
 - iv. 选中正则表达式复选框。
 - v. 单击确定。
 - vi. 在 **SSO** 配置中，展开资源。
- c. 添加身份验证策略。
在 **SSO** 配置中，展开身份验证策略。
- i. 选中允许 **CORS** 和需要安全 **Cookie** 复选框。
 - ii. 单击托管资源下的添加，并将表单或访问令牌定义为 SSO 资源的身份验证方法。



The screenshot shows the "Add Resource" dialog box with the following configuration:

- * Resource:** A dropdown menu showing "EPM".
- * Authentication Method:** A dropdown menu showing "Form or Access Token".
- Authentication Method Overrides:** A plus sign (+) to expand the section.
- Headers:** A plus sign (+) to expand the section.
- Header Configuration Table:**

Name	Value
HYPLOGIN	Work Email
- Add:** A blue button at the bottom right.

- iii. 在资源中，选择在前面步骤中添加的 SSO 资源。
- iv. 展开标头。
- v. 输入将传播到 EPM System 的 HTTP 头的名称。
默认的身份验证头名称为 `HYPLOGIN`。您可以使用自己选择的任何名称。

- vi. 在值中，选择唯一标识 EPM System 用户的属性。
此字段的值应与 EPM System 中的用户标识匹配。例如，如果 EPM System 中的用户标识为电子邮件 ID，则选择“工作电子邮件”作为值。
 - vii. 单击保存。
5. 单击完成以创建企业应用程序。
 6. 单击激活以启用应用程序。
 7. 注册应用程序网关，并为 EPM System 设置主机和应用程序。
 - a. 在导航抽屉中，依次单击安全性和应用程序网关。
 - b. 单击添加。
 - c. 在详细信息中，输入网关的唯一名称和可选说明。
 - d. 单击下一步以打开“主机”屏幕。
 - e. 为 EPM System 添加应用程序网关主机。
 - i. 在“主机”屏幕中，单击添加。

The screenshot shows a dialog box titled "Add Host" with a close button (X) in the top right corner. It contains the following fields and options:

- Host Identifier:** A text input field containing "EPMAppGateway".
- Host:** A text input field containing "gateway.server.example.com".
- Port:** A text input field containing "443".
- SSL Enabled:** A checkbox that is checked.
- Additional Properties:** A text area containing the following text:

```
ssl_certificate /usr/local/gateway.server.example.com.crt;  
ssl_certificate_key /usr/local/gateway.server.example.com.key;  
ssl_password_file /usr/local/gateway.server.example.com.password.txt;
```
- Save:** A green button at the bottom right.

- ii. 在主机标识符中，输入 EPMAppGateway。
- iii. 在主机中，输入用于托管应用程序网关服务器的计算机的完全限定域名，例如 gateway.server.example.com。
- iv. 在端口中，输入应用程序网关服务器响应 HTTPS 请求的端口。
- v. 选中已启用 **SSL** 复选框。
- vi. 在其他属性选项卡中，输入以下信息：
 - SSL 证书位置
 - SSL 证书密钥
 - SSL 密码文件（如果需要）有关详细信息，请参阅《Administering Oracle Identity Cloud Service》中 "Setup an App Gateway" 中的 "Register an App Gateway"。
- vii. 单击保存。
- viii. 单击下一步以打开“应用程序”屏幕。

- f. 将 EPM System 企业应用程序添加到应用程序网关。
 - i. 在应用程序上，单击添加。
 - ii. 在应用程序中，选择之前添加到 Oracle Cloud Identity Console 的 EPM System 企业应用程序。

The screenshot shows a dialog box titled "Assign an App to gate". It contains the following fields and values:

- * Application: EPM System
- * Select a Host: EPMAAppGateway
- Policy: default
- * Resource Prefix: /
- * Origin Server: https://epm.server.example.com:443
- Additional Properties: ssl_certificate /usr/local/epm.server.example.com.crt; ssl_certificate_key /usr/local/epm.server.example.com.key; ssl_password_file /usr/local/epm.server.example.com.password.txt;

A green "Save" button is located at the bottom right of the dialog.

- iii. 在选择主机中，选择 EPMAAppGateway（已添加到应用程序网关的 EPM System 主机）。
 - iv. 在资源前缀中，输入 / 以将所有请求转发到 EPM System 主机。
 - v. 在源服务器中，输入用于托管 Oracle Hyperion Enterprise Performance Management Workspace 的计算机的完全限定域名，以及 EPM Workspace 使用的端口号。
 - vi. 单击保存。
8. 记录应用程序网关的“客户端 ID”和“客户端密钥”。这些值是设置应用程序网关所必需的。
 - a. 在导航抽屉中，依次单击安全性和应用程序网关。
 - b. 单击您为 EPM System 企业应用程序添加的网关名称。
 - c. 将客户端 ID（字母数字字符串）复制到文本编辑器。
 - d. 单击显示密钥以显示客户端密钥代码。
 - e. 将客户端密钥（字母数字字符串）复制到文本编辑器。
 - f. 保存文本文件。

 **Note:**

每次对 Oracle Identity Cloud Service 进行配置更新时，必须重新启动应用程序网关服务器。要启动和停止应用程序网关服务器，请参阅[启动和停止应用程序网关](#)。

配置应用程序网关

有关详细信息，请参阅《*Administering Oracle Identity Cloud Service*》中的“[Set Up an App Gateway](#)”。

配置应用程序网关服务器时，需要使用在上一节中记录的客户端 ID 和客户端密钥。

配置用户目录以进行授权

某些身份管理产品（例如，Oracle Identity Cloud Service 和 Microsoft Azure）无法直接配置为 Oracle Enterprise Performance Management System 中的用户目录。您可以使用 Oracle Unified Directory 或 Oracle Virtual Directory 配置此类产品，然后将 Oracle Unified Directory 或 Oracle Virtual Directory 配置为 EPM System 中的用户目录。有关配置用户目录的详细步骤，请参阅“[配置用户目录](#)”。

在 EPM System 中启用 SSO

在 Oracle Enterprise Performance Management System 中配置“安全选项”以启用 SSO。有关详细说明，请参阅“[设置安全选项](#)”。

要启用 SSO：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。
2. 选择管理，然后选择配置用户目录。
3. 单击安全选项。
4. 在单点登录配置部分中：
 - a. 选中启用 **SSO** 复选框。
 - b. 从 **SSO** 提供程序或安全代理下拉列表中，选择其他。
 - c. 从 **SSO** 机制下拉列表中，选择自定义 **HTTP** 头，然后指定安全代理传递到 EPM System 的头名称（`HYPLOGIN`，或者您在 Oracle Cloud Identity Console 中为企业应用程序添加资源时指定的自定义名称）。
5. 单击确定。

Note:

确保在进行任何 SSO 配置更改后重新启动所有 EPM System 服务。

更新 EPM Workspace 设置

1. 以系统管理员身份访问 Oracle Hyperion Enterprise Performance Management Workspace。请参阅“[访问 EPM Workspace](#)”。
2. 依次选择导航、**Workspace** 设置和服务器设置。
3. 在 **Workspace** 服务器设置中，将注销后的 **URL** 更改为您希望用户在注销 Oracle Enterprise Performance Management System 时看到的网页 URL。
4. 单击确定。

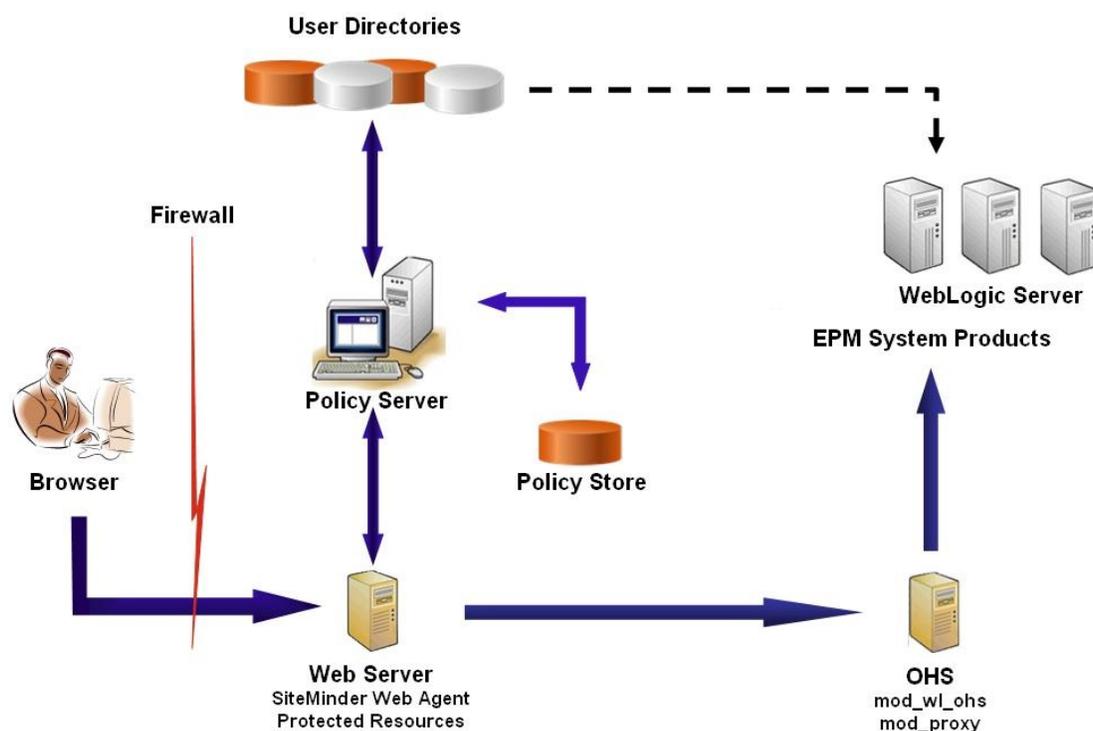
- 重新启动 Oracle Hyperion Foundation Services 和所有 EPM System 组件。

SiteMinder SSO

SiteMinder 是一个纯 Web 解决方案。桌面应用程序及其插件（例如，Microsoft Excel 和 Report Designer）无法通过 SiteMinder 使用身份验证。但是，Oracle Smart View for Office 可以使用 SiteMinder 身份验证。

流程流

以下图例概述了已启用 SiteMinder 的 SSO：



SiteMinder SSO 进程：

1. 用户尝试访问受 SiteMinder 保护的 Oracle Enterprise Performance Management System 资源。他们通过 URL 连接到作为 SiteMinder 策略服务器前端的 Web 服务器；例如，http://WebAgent_Web_Server_Name:WebAgent_Web_ServerPort/interop/index.jsp
2. Web 服务器将用户重定向到策略服务器，后者将向用户质询凭据。在根据配置的用户目录验证凭据之后，策略服务器将凭据传递给托管 SiteMinder Web 代理的 Web 服务器。
3. 托管 SiteMinder Web 代理的 Web 服务器将请求重定向到作为 EPM System 前端的 Oracle HTTP Server。Oracle HTTP Server 将用户重定向到所请求的部署在 Oracle WebLogic Server 上的应用程序。
4. EPM System 组件检查设置信息并提供内容。为了使此过程正常运行，SiteMinder 用于验证用户身份的用户目录必须配置为 EPM System 中的外部用户目录。这些目录还必须配置为受信任的目录。

特殊注意事项

SiteMinder 是一个纯 Web 解决方案。桌面应用程序及其插件（例如，Microsoft Excel 和 Report Designer）无法通过 SiteMinder 使用身份验证。但是，Smart View 可以使用 SiteMinder 身份验证。

先决条件

1. 功能完整的 SiteMinder 安装，其中包含以下组件：
 - SiteMinder 策略服务器，您可以在其上定义策略和代理对象
 - SiteMinder Web 代理，安装在作为 SiteMinder 策略服务器前端的 Web 服务器上
2. 功能完整的 EPM System 部署。
当您为 EPM System 组件配置 Web 服务器时，EPM System Configurator 会将 `mod_wl_ohs.conf` 配置为通过代理将请求转发给 WebLogic Server。

启用 SiteMinder Web 代理

Web 代理安装在 Web 服务器上，它将拦截 EPM System 资源请求。未经身份验证的用户尝试访问受保护的 EPM System 资源时，会强制 Web 代理向用户质询 SSO 凭据。当用户经过身份验证时，策略服务器将添加经过身份验证的用户的登录名称，该名称由标题传递。随后，HTTP 请求将传递给 EPM System Web 服务器，该服务器将重定向请求。EPM System 组件从头中提取经身份验证的用户凭据。

SiteMinder 支持在各种异构 Web 服务器平台上运行的 EPM System 产品之间使用 SSO。如果 EPM System 产品使用不同的 Web 服务器，您必须确保可以在同一域内的 Web 服务器之间传递 SiteMinder Cookie。为此，您需要在每个 Web 服务器的 `WebAgent.conf` 文件中，将 `Cookiedomain` 属性的值指定为相应的 EPM System 应用程序域。

请参阅《*Netegrity SiteMinder Agent Guide*》中的 "Configuring Web Agents"。

注：

由于 Oracle Hyperion Shared Services 使用基本身份验证来保护其内容，因此拦截 Shared Services 请求的 Web 服务器应启用基本身份验证以支持 SiteMinder SSO。

要配置 Web 代理，请执行 `WEBAGENT_HOME/install_config_info/nete-wa-config` 以运行 SiteMinder Web 代理配置向导；例如，在 Windows 上，可执行 `C:\netegrity\webagent\install_config_info\nete-wa-config.exe`。配置过程中将为 SiteMinder Web 服务器创建 `WebAgent.conf`。

要启用 SiteMinder Web 代理：

1. 使用文本编辑器打开 `WebAgent.conf`。此文件的位置取决于您正在使用的 Web 服务器。
2. 将 `enableWebAgent` 属性的值设置为 `Yes`。
`enableWebAgent="YES"`
3. 保存并关闭 Web 代理配置文件。

示例 3-1 配置 SiteMinder 策略服务器

SiteMinder 管理员必须配置策略服务器以便启用 EPM System 产品 SSO。

配置过程包括：

- 创建 SiteMinder Web 代理并添加适用于 SiteMinder Web 服务器的配置对象
- 为每个应受保护的 EPM System 资源创建一个领域，并将 Web 代理添加到该领域。请参阅[“要保护的资源”](#)
- 在为受保护的 EPM System 资源创建的领域内，为取消保护的资源创建领域。请参阅[“取消保护的资源”](#)
- 创建 HTTP 头引用。该头应向 EPM System 应用程序提供登录属性的值。有关登录属性的简短说明，请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。
- 在领域内创建使用 Get、Post 和 Put 作为 Web 代理操作的规则
- 创建值为 `hyplogin=<%userattr="SM_USERLOGINNAME"%>` 的响应属性
- 创建策略，分配用户目录访问权并将为 EPM System 创建的规则添加到当前成员列表
- 为您针对 EPM System 组件创建的规则设置响应

示例 3-2 配置 SiteMinder Web 服务器以将请求转发给 EPM System Web 服务器

配置托管 SiteMinder Web 代理的 Web 服务器，以便将经身份验证的用户（包含标识用户的头）发出的请求转发给 EPM System Web 服务器。

对于基于 Apache 的 Web 服务器，使用类似如下的指令转发经身份验证的请求：

```
ProxyPass / http://EPM_WEB_SERVER:EPM_WEB_SERVER_PORT/  
ProxyPassReverse / http://EPM_WEB_SERVER:EPM_WEB_SERVER_PORT/  
ProxyPreserveHost On  
#If SiteMinder Web Server is using HTTPS but EPM Web Server is using HTTP  
RequestHeader set WL-Proxy-SSL true
```

在此指令中，将 `EPM_WEB_SERVER` 和 `EPM_WEB_SERVER_PORT` 替换为环境的实际值。

示例 3-3 在 EPM System 中启用 SiteMinder

与 SiteMinder 的集成要求您为 EPM System 产品启用 SiteMinder 身份验证。请参阅[“针对 SSO 配置 EPM System”](#)。

Kerberos 单点登录

概览

如果为托管 EPM System 产品的应用程序服务器设置了 Kerberos 身份验证，Oracle Enterprise Performance Management System 产品将支持 Kerberos SSO。

Kerberos 是一个受信任的身份验证服务，其中每个 Kerberos 客户端信任其他 Kerberos 客户端（用户、网络服务等）的身份。

用户访问 EPM System 产品时，会发生以下情况：

1. 用户通过 Windows 计算机登录到 Windows 域，即 Kerberos 领域。

2. 用户通过配置为使用集成 Windows 身份验证的浏览器尝试登录到运行于应用程序服务器上的 EPM System 产品。
3. 应用程序服务器 (Negotiate Identity Asserter) 会拦截请求，并从浏览器的授权头中获取包含 Kerberos 票证的简单和受保护通用安全服务 API (GSSAPI) 协商机制 (SPNEGO) 令牌。
4. 该声明器依据其身份存储库来验证令牌中包含的用户标识，以将有关用户的信息传递给 EPM System 产品。EPM System 产品依据 Active Directory 对用户名进行验证。EPM System 产品发出 SSO 令牌，该令牌支持跨所有 EPM System 产品使用 SSO。

支持限制

支持为所有 EPM System 产品使用 Kerberos SSO，但有以下例外：

- Oracle Smart View for Office 以外的胖客户端不支持 Kerberos SSO。
- Smart View 仅支持 Kerberos 与 Oracle Essbase、Oracle Hyperion Planning 及 Oracle Hyperion Financial Management 提供程序之间的集成

假设

本文档包含应用程序级别的 Kerberos 配置步骤，且假定您已掌握系统级别的 Kerberos 配置知识。在开始这些过程之前，请确认已满足这些任务的先决条件。

本文档假定您在使用完全功能且已启用 Kerberos 的网络环境中工作，该环境中已配置用于 Kerberos 身份验证的 Windows 客户端。

- 已为公司的 Active Directory 配置 Kerberos 身份验证。请参阅“[Microsoft Windows Server 文档](#)”。
- 已配置用于访问 EPM System 产品的浏览器使用 Kerberos 票证进行协商。
- KDC 和客户端计算机之间的时间同步偏差不超过五分钟。请参阅 "Authentication Errors are Caused by Unsynchronized Clocks": [http://technet.microsoft.com/en-us/library/cc780011\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc780011(WS.10).aspx)。

针对 WebLogic Server 的 Kerberos SSO

Oracle WebLogic Server Kerberos SSO 使用 Negotiate Identity Asserter 来协商和解码 SPNEGO 令牌，以便针对 Microsoft 客户端启用 SSO。WebLogic Server 解码 SPNEGO 令牌以获得 Kerberos 票证，并在验证票证后将其映射到 WebLogic Server 用户。您可以将 WebLogic Server 的 Active Directory 身份验证器与 Negotiate Identity Asserter 一起使用，以将 Active Directory 配置为 WebLogic Server 用户的用户目录。

当浏览器请求访问 EPM System 产品时，KDC 将向浏览器发出 Kerberos 票证，该票证将创建包含支持的 GSS 令牌类型的 SPNEGO 令牌。Negotiate Identity Asserter 解码 SPNEGO 令牌，并使用 GSSAPI 来接受安全上下文。发起请求的用户的标识将映射为用户名，并回传到 WebLogic Server。此外，WebLogic Server 将确定用户所属的组。在此阶段，将向用户提供所请求的 EPM System 产品。

注：

用户必须使用支持 SPNEGO 的浏览器（例如，Internet Explorer 或 Firefox）来访问在 WebLogic Server 上运行的 EPM System 产品。

EPM System 产品授权过程使用派生自身份验证过程的用户 ID 检查设置数据。将基于设置数据对 EPM System 产品访问进行限制。

支持 Kerberos 身份验证的 WebLogic Server 步骤

管理员应完成以下任务以支持 Kerberos 身份验证：

- 为 EPM System 创建 WebLogic 域。请参阅“[为 EPM System 创建 WebLogic 域](#)”。
- 创建身份验证提供程序。请参阅“[在 WebLogic Server 中创建 LDAP 身份验证提供程序](#)”。
- 创建 Negotiate Identity Asserter。请参阅“[创建 Negotiate Identity Asserter](#)”。
- 创建 Kerberos 标识。请参阅“[为 WebLogic Server 创建 Kerberos 标识](#)”。
- 更新 Kerberos 的 JVM 选项。请参阅“[更新 Kerberos 的 JVM 选项](#)”。
- 配置授权策略。请参阅“[配置授权策略](#)”。
- 部署 SSODiag 并使用它验证 WebLogic Server 是否已准备好支持 EPM System 使用 Kerberos SSO。请参阅“[使用 SSODiag 测试 Kerberos 环境](#)”。

为 EPM System 创建 WebLogic 域

通常，可以将 EPM System 组件部署到 EPMSystem WebLogic 域中（默认位置是 `MIDDLEWARE_HOME/user_projects/domains/EPMSystem`）。

要配置 EPM System WebLogic 域使用 Kerberos 身份验证：

1. 安装 EPM System 组件。
2. 仅部署 Oracle Hyperion Foundation Services。
Foundation Services 部署会创建默认的 EPM System WebLogic 域。
3. 登录到 Oracle Hyperion Shared Services Console，验证 Foundation Services 部署是否成功。请参阅“[启动 Shared Services Console](#)”。

在 WebLogic Server 中创建 LDAP 身份验证提供程序

WebLogic Server 管理员可创建 LDAP 身份验证提供程序，该程序将用户和组信息存储在外部 LDAP 服务器中。遵循 LDAP v2 或 v3 标准的 LDAP 服务器可与 WebLogic Server 结合使用。请参阅以下参考资料：

- 《*Oracle Fusion Middleware Securing Oracle WebLogic Server*》指南中的“[Configuring LDAP Authentication Providers](#)”。
- 《*Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*》中的“[Configure Authentication and Identity Assertion Providers](#)”。

创建 Negotiate Identity Asserter

Negotiate Identity Assertion 提供程序可针对 Microsoft 客户端启用 SSO。它会解码 SPNEGO 令牌以获取 Kerberos 令牌，并在验证 Kerberos 令牌后将其映射到 WebLogic 用户。Negotiate Identity Assertion 提供程序是根据 WebLogic Security Framework 的定义进行的安全服务提供程序接口 (SSPI) 实施，提供了根据客户端的 SPNEGO 令牌对客户端进行身份验证的必要逻辑。

- 《*Oracle Fusion Middleware Securing Oracle WebLogic Server*》指南中的“[Configuring a Negotiate Identity Assertion Provider](#)”。
- 《*Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*》中的“[Configure Authentication and Identity Assertion Providers](#)”。

在创建 Negotiate Identity Assertion 提供程序时，将所有身份验证器的“JAAS 控制标志”选项设置为 SUFFICIENT。请参阅《Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help》中的“Set the JAAS control flag”。

为 WebLogic Server 创建 Kerberos 标识

在 Active Directory 域控制器计算机上，创建表示 WebLogic Server 和 EPM System Web 服务器的用户对象，然后将它们映射到表示 Kerberos 领域中 WebLogic Server 和 Web 服务器的服务主体名称 (SPN)。客户端无法定位到没有 SPN 的服务。您可将 SPN 存储在密码表文件中，系统会将这些文件复制到登录过程中要使用的 WebLogic Server 域中。

有关详细过程，请参阅《Oracle Fusion Middleware Securing Oracle WebLogic Server》指南中的“Creating identification for WebLogic Server”。

要为 WebLogic Server 创建 Kerberos 标识：

1. 在 Active Directory 域控制器计算机上，为托管 WebLogic Server 域的计算机创建用户帐户，例如 `epmHost`。

注：

将标识创建为用户对象（而不是计算机）。
使用计算机的简单名称；例如，如果主机名称为 `epmHost.example.com`，
则使用 `epmHost`。

记录创建用户对象时使用的密码。创建 SPN 时会需要它。

请勿选择任何密码选项，尤其是用户必须在下次登录时更改密码选项。

2. 修改用户对象以遵循 Kerberos 协议。帐户必须要求 Kerberos 预身份验证。
 - 在帐户选项卡中，选择要使用的加密。
 - 确保没有选中其他帐户选项（尤其是不要求 Kerberos 预身份验证）。
 - 由于设置加密类型可能会损坏对象的密码，因此请将此密码重置为创建对象时所设置的密码。
3. 在托管 Active Directory 域控制器的计算机上，打开命令提示窗口，然后导航到 Active Directory 支持工具的安装目录。
4. 创建并配置所需的 SPN。
 - a. 使用类似如下的命令验证 SPN 是否与在此过程的步骤 1 中创建的用户对象 (`epmHost`) 相关联。

```
setspn -L epmHost
```

- b. 使用类似如下的命令，为 Active Directory 域服务 (AD DS) 中的 WebLogic Server 配置 SPN，然后生成包含共享密钥的密码表文件。

```
ktpass -princ HTTP/epmHost.example.com@EXAMPLE.COM -pass  
password -mapuser epmHost -out c:\epmHost.keytab
```

5. 在托管 WebLogic Server 的计算机上创建密码表文件。
 - a. 打开命令提示窗口。

- b. 导航到 `MIDDLEWARE_HOME/jdk/bin`。
- c. 执行类似下面的命令：

```
ktab -k keytab_filename -a epmHost@example.com
```

- d. 当提示输入密码时，输入在此过程的步骤 1 中创建用户时所设置的密码。
6. 将密码表文件复制到 WebLogic 域的启动目录中；例如，复制到 `C:\Oracle\Middleware\user_projects\domains\EPMSystem` 中。
7. 确认 Kerberos 身份验证是否正常运行。

```
kinit -k -t keytab-file account-name
```

在此命令中，`account-name` 指示 Kerberos 主体；例如，`HTTP/epmHost.example.com@EXAMPLE.COM`。此命令的输出应类似如下所示：

```
New ticket is stored in cache file C:\Documents and Settings\Username\krb5cc_MachineB
```

更新 Kerberos 的 JVM 选项

请参阅《Oracle Fusion Middleware Securing Oracle WebLogic Server 11g Release 1 (10.3.1)》中的 ["Using Startup Arguments for Kerberos Authentication with WebLogic Server"](#) 和 ["Creating a JAAS Login File"](#)。

如果 EPM System 受管服务器作为 Windows 服务运行，则更新 Windows 注册表以设置 JVM 启动选项。

要在 Windows 注册表中更新 JVM 启动选项：

1. 打开 Windows 注册表编辑器。
2. 依次选择我的电脑、**HKEY_LOCAL_MACHINE**、**Software**、**Hyperion Solutions**、**Foundationservices0** 以及 **HyS9EPMServer_epmsystem1**。
3. 创建以下字符串值：



注：

下表中列出的名称是一些示例。

表 3-3 Kerberos 身份验证的 JVM 启动选项

名称	类型	数据
JVMOption44	REG_SZ	<code>-Djava.security.krb5.realm=Active Directory Realm Name</code>
JVMOption45	REG_SZ	<code>-Djava.security.krb5.kdc=Active Directory host name or IP address</code>
JVMOption46	REG_SZ	<code>-Djava.security.auth.login.config=location of Kerberos login configuration file</code>

表 3-3 (续) Kerberos 身份验证的 JVM 启动选项

名称	类型	数据
JVMOption47	REG_SZ	- Djavax.security.auth.useSubjectCredsOnly=false

- 更新 JVMOptionCount DWord 的值以反映添加的 JVMOptions (当前的小数值加 4)。

配置授权策略

有关为访问 EPM System 的 Active Directory 用户配置授权策略的信息, 请参阅《Oracle Fusion Middleware Securing Resources Using Roles and Policies for Oracle WebLogic Server》指南中的“Options for Securing Web Application and EJB Resources”。

有关示例策略配置步骤, 请参阅“创建 SSODiag 策略”。

使用 SSODiag 测试 Kerberos 环境

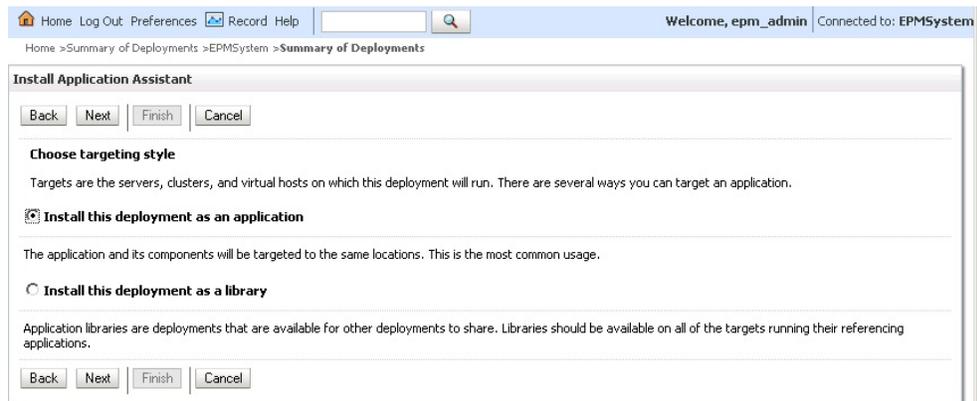
SSODiag 是一个诊断性的 Web 应用程序, 可以测试 Kerberos 环境中的 WebLogic Server 是否已准备好支持 EPM System。

部署 SSODiag

使用部署 Foundation Services 时指定的 WebLogic Server 管理员凭据 (默认用户名为 epm_admin) 来部署 SSODiag。

要部署和配置 SSODiag:

- 登录到 EPM System 域的 WebLogic Server 管理控制台。
- 在“更改中心”中, 单击锁定并编辑
- 从域结构中的 **EPMSystem**, 单击部署。
- 在部署摘要中, 单击安装。
- 在路径中, 选择 `EPM_ORACLE_HOME/products/Foundation/AppServer/InstallableApps/common/SSODiag.war`。
- 单击下一步。
- 在选择定位样式中, 确保已选中将此部署安装为应用程序, 然后单击下一步。



8. 在选择部署目标中，选择以下目标，然后单击下一步。

- EPMServer
- 群集中的所有服务器

The screenshot shows the 'Install Application Assistant' wizard in the 'Summary of Deployments' section. The current step is 'Select deployment targets'. The user is prompted to select servers and/or clusters for deployment. Under 'Available targets for SSODiag', the 'Clusters' section is expanded, and 'EPMServer' is selected with the radio button 'All servers in the cluster' chosen. Navigation buttons 'Back', 'Next', 'Finish', and 'Cancel' are visible at the top and bottom of the wizard.

9. 在可选设置中的安全模型中，选择定制角色和策略：仅使用在管理控制台中定义的角色和策略。

The screenshot shows the 'Install Application Assistant' wizard in the 'Optional Settings' step. The user is prompted to modify settings or accept defaults. Under the 'General' section, the deployment name is set to 'SSODiag'. Under the 'Security' section, the radio button 'Custom Roles and Policies: Use only roles and policies that are defined in the Administration Console.' is selected. Navigation buttons 'Back', 'Next', 'Finish', and 'Cancel' are visible at the top of the wizard.

10. 单击下一步。

11. 在“预览”屏幕上，选择否，稍后复查配置。

12. 单击完成。

13. 在“更改中心”中，选择激活更改。

配置 Oracle HTTP Server 处理 SSODiag

更新 `mod_wl_ohs.conf` 以配置 Oracle HTTP Server，将 SSODiag URL 请求转发到 WebLogic Server。

要配置 Oracle HTTP Server 中的 URL 转发：

1. 使用文本编辑器打开 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/fmwconfig/components/OHS/ohs_component/mod_wl_ohs.conf`。
2. 为 SSODiag 添加 `LocationMatch` 定义：

```
<LocationMatch /SSODiag/>
    SetHandler weblogic-handler
    WeblogicCluster myServer:28080
</LocationMatch>
```

在先前的示例中，`myServer` 表示 Foundation Services 主机，28080 表示 Oracle Hyperion Shared Services 在其上侦听请求的端口。

3. 保存并关闭 `mod_wl_ohs.conf`。
4. 重新启动 Oracle HTTP Server。

创建 SSODiag 策略

在 WebLogic Server 管理控制台中创建策略以保护以下 SSODiag URL。

```
http://OHS_HOST_NAME:PORT/SSODiag/krbssodiag
```

在本示例中，`OHS_HOST_NAME` 表示托管 Oracle HTTP Server 的服务器名称，`PORT` 表示 Oracle HTTP Server 在其上侦听请求的端口。

要创建策略以保护 SSODiag：

1. 在 EPM System 域 WebLogic Server 管理控制台的“更改中心”内，选择锁定并编辑。
2. 依次选择部署、SSODiag、安全性、URLPatterns 和策略。
3. 创建以下 URL 模式：
 - /
 - /index.jsp
4. 修改您创建的每个 URL 模式：
 - a. 从独立 Web 应用程序 URL 模式中的 URL 模式列表中，单击您创建的模式 (/) 打开它。
 - b. 选择添加条件。
 - c. 在谓词列表中，选择用户。
 - d. 选择下一步。
 - e. 在用户参数名中，输入访问已配置为使用 Kerberos 身份验证的客户端桌面时使用的帐户对应的 Active Directory 用户（例如 `krbuser1`），然后选择添加。
`krbuser1` 是 Active Directory 或 Windows 桌面用户。

- f. 选择完成。
5. 选择保存。

使用 SSODiag 测试用于 Kerberos 身份验证的 WebLogic Server 配置

如果用于 Kerberos 身份验证的 WebLogic Server 配置正常运行，则 *Oracle Hyperion Kerberos SSO diagnostic Utility V 1.0* 页会显示以下信息：

```
Retrieving Kerberos User principal name... Success.  
Kerberos principal name retrieved... SOME_USER_NAME
```

▲ 注意：

如果 SSODiag 无法检索 Kerberos 主体名称，请勿配置 EPM System 组件使用 Kerberos 身份验证。

要测试用于 Kerberos 身份验证的 WebLogic Server 配置：

1. 启动 Foundation Services 和 Oracle HTTP Server。
2. 使用 WebLogic Server 管理控制台，启动 SSODiag Web 应用程序，让其为所有请求提供服务。
3. 使用有效的 Active Directory 凭据，登录到配置为使用 Kerberos 身份验证的客户端计算机。
4. 使用浏览器，连接到以下 SSODiag URL：

```
http://OHS_HOST_NAME:PORT/SSODiag/krbssodiag
```

在本示例中，*OHS_HOST_NAME* 表示托管 Oracle HTTP Server 的服务器名称，*PORT* 表示 Oracle HTTP Server 在其上侦听请求的端口。

如果 Kerberos 身份验证正常运行，SSODiag 将显示以下信息：

```
Retrieving Kerberos User principal name... Success.  
Kerberos principal name retrieved... SOME_USER_NAME
```

如果 Kerberos 身份验证不能正常运行，SSODiag 将显示以下信息：

```
Retrieving Kerberos User principal name... failed.
```

更改安全模型

由安全领域保护的 Web 应用程序的安全模型是 `DOnly`。您必须将安全模型更改为 `CustomRolesAndPolicies`。

要更改安全模型：

1. 使用文本编辑器打开 `MIDDLEWARE_HOME/user_projects/domains/EPMSystem/config/config.xml`。

2. 在每个 Foundation Services 组件的应用程序部署描述符中查找以下元素：

```
<security-dd-model>DDOnly</security-dd-model>
```

3. 按如下所示更改每个组件的安全模型：

```
<security-dd-model>CustomRolesAndPolicies</security-dd-model>
```

4. 保存并关闭 config.xml。

更新 EPM System 安全配置

更改 EPM System 安全配置以启用 Kerberos SSO。

要配置 EPM System 使用 Kerberos 身份验证：

1. 以管理员身份登录到 Shared Services Console。
2. 将已配置使用 Kerberos 身份验证的 Active Directory 域添加为 Shared Services 中的外部用户目录。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”。
3. 启用 SSO。请参阅“[配置 OID、Active Directory 和其他基于 LDAP 的用户目录](#)”。在安全选项中，选择下表中的设置以启用 Kerberos SSO。

表 3-4 启用 Kerberos SSO 的设置

字段	所需的设置
启用 SSO	已选中
SSO 提供程序或代理	其他
SSO 机制	从 HTTP 请求获取远程用户

4. 重新启动 Shared Services。

测试 Kerberos SSO

登录到 Foundation Services 以验证 Kerberos SSO 是否在正常运行。

要测试 Kerberos SSO：

1. 验证 Foundation Services 和 Oracle HTTP Server 是否正在运行。
2. 使用有效的 Active Directory 凭据登录已配置使用 Kerberos 身份验证的客户端计算机。
3. 使用浏览器连接到 Foundation Services URL。

配置 EPM System 组件

使用 EPM System Configurator 配置其他 EPM System 组件，并将其部署到已部署 Foundation Services 的 WebLogic 域。

配置用于 Kerberos 身份验证的 EPM System 受管服务器

在 Microsoft Windows 环境中，EPM System 受管服务器作为 Windows 服务运行。您必须修改每个 WebLogic 受管服务器的启动 JVM 选项。非紧凑部署模式下的受管服务器综合列表：

- AnalyticProviderServices0
- CalcMgr0
- ErpIntegrator0
- EssbaseAdminServer0
- FinancialReporting0
- HFMWeb0
- FoundationServices0
- HpsAlerter0
- HpsWebReports0
- Planning0
- Profitability0

如果 EPM System Web 应用程序在紧凑部署模式下部署，则仅需要更新 EPMSystem0 受管服务器的启动 JVM 选项。如果有多个紧凑的受管服务器，则必须更新所有受管服务器的启动 JVM 选项。

请参阅《Oracle Fusion Middleware Securing Oracle WebLogic Server》指南中的 "[Using Startup Arguments for Kerberos Authentication with WebLogic Server](#)"。

 注：

以下过程介绍如何设置 FoundationServices 受管服务器的启动 JVM 选项。您必须为部署中的每个 WebLogic 受管服务器执行此任务。

有关在 WebLogic Server 启动脚本中配置 JVM 选项的详细信息，请参阅“[更新 Kerberos 的 JVM 选项](#)”。

要在 WebLogic Server 启动脚本中配置 JVM 选项

配置授权策略

为将访问 EPM System 组件（Foundation Services 除外）的 Active Directory 用户配置授权策略。有关通过 WebLogic 管理控制台配置安全策略的信息，请参阅“[配置授权策略](#)”。

更改 EPM System 组件的默认安全模型

可以编辑 EPM System 配置文件以更改默认的安全模型。对于非紧凑 EPM System 部署，必须更改 config.xml 中记录的每个 EPM System Web 应用程序的默认安全模型。EPM System Web 应用程序列表：

- AIF
- APS
- CALC
- EAS
- FINANCIALREPORTING
- PLANNING

- PROFITABILITY
- SHAREDSEVICES
- WORKSPACE

要更改安全模型：

1. 使用文本编辑器打开 `MIDDLEWARE_HOME/user_projects/domains/EPMSysstem/config/config.xml`
2. 在每个 EPM System 组件的应用程序部署定义中，按以下示例所示将 `<security-dd-model>` 的值设置为 `CustomRolesAndPolicies`：

```
<app-deployment>
  <name>SHAREDSEVICES#11.1.2.0</name>
  <target>EPMServer</target>
  <module-type>ear</module-type>
  <source-path>C:\Oracle\Middleware\EPMSysstem11R1/products/
Foundation/AppServer/InstallableApps/common/interop.ear</source-
path>
  <security-dd-model>CustomRolesAndPolicies</security-dd-model>
  <staging-mode>nostage</staging-mode>
</app-deployment>
```

3. 保存并关闭 `config.xml`。
4. 重新启动 WebLogic Server。

为 EPM System 组件创建 URL 保护策略

在 WebLogic Server 管理控制台中，创建 URL 保护策略以保护每个 EPM System 组件的 URL。有关详细信息，请参阅《*Oracle Fusion Middleware Securing Resources Using Roles and Policies for Oracle WebLogic Server*》指南中的“[Options for Securing Web Applications and EJB Resources](#)”。

要创建 URL 保护策略：

1. 在 EPM System 域的 WebLogic Server 管理控制台的“更改中心”内，单击锁定并编辑。
2. 单击部署。
3. 展开部署中的 EPM System 企业应用程序（例如 `PLANNING`），然后单击其 Web 服务器（例如 `HyperionPlanning`）。有关 EPM System 组件的列表，请参阅“[更改 EPM System 组件的默认安全模型](#)”。

注：

一些企业应用程序（例如 Oracle Essbase Administration Services），包含多个 Web 应用程序，必须对它们的 URL 模式进行定义。

4. 为 Web 应用程序创建 URL 模式作用域策略。
 - AIF
 - APS
 - CALC

- EAS
 - FINANCIALREPORTING
 - PLANNING
 - PROFITABILITY
 - SHAREDSEVICES
 - WORKSPACE
- a. 依次单击安全性、策略和新建。
 - b. 在 **URL 模式** 中，输入 EPM System 产品的受保护和取消保护的 URL。有关更多详细信息，请参阅“[保护和取消保护 EPM System 资源](#)”。
 - c. 单击确定。
 - d. 单击您创建的 URL 模式。
 - e. 单击添加条件。
 - f. 在谓词列表中，选择策略条件，然后单击下一步。
Oracle 建议使用组条件，这会将安全策略授予指定组中的所有成员。
 - g. 指定与您所选谓词相关的参数。例如，如果在上一步中选择了组，则应完成以下步骤：
 - 单击添加。
 - 重复先前的步骤以添加更多组。
 - i. 单击完成。
如果 WebLogic Server 无法在 Active Directory 中找到组，则会显示一条错误消息。您必须先解决此错误，然后再继续。
 - j. 选择保存。
5. 对于部署中的其他 EPM System 组件，重复此过程的步骤 3 和 步骤 4。
 6. 在“更改中心”内，单击版本配置。
 7. 重新启动 WebLogic Server。

在 Web 应用程序中启用基于客户端证书的身份验证

在 `EPM_ORACLE_HOME/products/` 中以下应用程序存档的配置文件中，插入 `login-config` 定义。

- `Essbase/eas/server/AppServer/InstallableApps/Common/eas.ear`
- `FinancialDataQuality/AppServer/InstallableApps/aif.ear`
- `financialreporting/InstallableApps/HReports.ear`
- `Profitability/AppServer/InstallableApps/common/profitability.ear`

要启用基于客户端证书的身份验证：

1. 停止 EPM System 组件和进程。
2. 使用 7 Zip，展开企业存档中包含的 Web 存档，例如 `EPM_ORACLE_HOME/products/Essbase/eas/server/AppServer/InstallableApps/Common/eas.ear/eas.war`

3. 导航到 WEB-INF。
4. 修改 web.xml，在紧挨 </webapp> 元素的前面添加以下 login_config 定义：

```
<login-config>  
    <auth-method>CLIENT-CERT</auth-method>  
</login-config>
```

5. 保存 web.xml。
6. 当 7-Zip 询问是否更新存档时，单击是。

更新 EPM System 安全配置

配置 EPM System 安全以使用 SSO。请参阅[“针对 SSO 配置 EPM System”](#)。

针对 SSO 配置 EPM System

Oracle Enterprise Performance Management System 产品必须配置为支持安全代理以实现 SSO。对于所有 EPM System 产品，Oracle Hyperion Shared Services 中指定的配置确定以下各项：

- 是否接受来自安全代理的 SSO
- 要针对 SSO 接受的身份验证机制

在启用 SSO 的环境中，用户首先访问的 EPM System 产品将解析 SSO 机制，以检索其中包含的通过验证的用户 ID。EPM System 产品将依据 Shared Services 中配置的用户目录检查用户 ID，以确定用户是否为有效的 EPM System 用户。它还发出一个令牌，该令牌可跨 EPM System 产品启用 SSO。

Shared Services 中指定的配置将启用 SSO，并确定要为所有 EPM System 产品的 SSO 接受的身份验证机制。

要从 Web 标识管理解决方案中启用 SSO：

1. 以 Shared Services 管理员身份启动 Oracle Hyperion Shared Services Console。请参阅[“启动 Shared Services Console”](#)。
2. 选择管理，然后选择配置用户目录。
3. 验证 Web 标识管理解决方案使用的用户目录是否已配置为 Shared Services 中的外部用户目录。

例如，要启用 Kerberos SSO，您必须配置 Active Directory，将其配置为用于 Kerberos 身份验证的外部用户目录。

有关说明，请参阅[“配置用户目录”](#)。

4. 选择安全选项。
5. 选择显示高级选项。
6. 在“定义的用户目录”屏幕的单点登录配置中，执行以下步骤：
 - a. 选择启用 SSO。
 - b. 从 SSO 提供程序或代理中，选择 Web 标识管理解决方案。如果要使用 Kerberos 配置 SSO，请选择其他。

建议的 SSO 机制将自动处于选定状态。请参阅下表。另请参阅[“支持的 SSO 方法”](#)。

 注:

如果未使用建议的 SSO 机制，您必须在 **SSO** 提供程序或代理中选择其他。例如，要使用除用于 SiteMinder 的 HTTP 头之外的其他机制，请在 **SSO** 提供程序或代理中选择其他，然后在 **SSO** 机制中选择要使用的 SSO 机制。

表 3-5 Web 标识管理解决方案的首选 SSO 机制

Web 标识管理解决方案	建议的 SSO 机制
Oracle Access Manager	自定义 HTTP 头 ¹
OSSO	自定义 HTTP 头
SiteMinder	自定义 HTTP 头
Kerberos	从 HTTP 请求获取远程用户

¹ 默认的 HTTP 头名称为 HYPLOGIN。如果在使用自定义 HTTP 头，请替换此名称。

7. 单击确定。

Smart View 的单点登录选项

虽然 Oracle Smart View for Office 是一个胖客户端而不是浏览器，但它使用 HTTP 连接到服务器组件，从系统角度来看，其行为非常类似于浏览器。Smart View 支持浏览器界面支持的所有基于 Web 的标准集成方法。但是，也存在一些局限性：

- 如果从连接到 Oracle Enterprise Performance Management System 组件的现有浏览器会话中启动 Smart View，则用户必须再次登录到 Smart View，因为它不共享现有会话中的 cookie。
- 如果使用基于 HTML 的自定义登录表单（而不是默认的 Oracle Access Manager 登录表单），请确保自定义表单的源包括字符串 `loginform`。这是 Smart View 与 Oracle Access Manager 集成所必需的。

4

配置用户目录

另请参阅：

- [用户目录和 EPM System 安全](#)
- [与用户目录配置相关的操作](#)
- [Oracle Identity Manager 和 EPM System](#)
- [Active Directory 信息](#)
- [配置 OID、Active Directory 和其他基于 LDAP 的用户目录](#)
- [将关系数据库配置为用户目录](#)
- [测试用户目录连接](#)
- [编辑用户目录设置](#)
- [删除用户目录配置](#)
- [管理用户目录搜索顺序](#)
- [设置安全选项](#)
- [重新生成加密密钥](#)
- [使用特殊字符](#)

用户目录和 EPM System 安全

很多用户和身份管理系统（统称为用户目录）都支持 Oracle Enterprise Performance Management System 产品。其中包括启用了轻量级目录访问协议 (LDAP) 的用户目录，如 Sun Java System Directory Server（以前称为 SunONE Directory Server）和 Active Directory。EPM System 还支持将关系数据库用作外部用户目录。

通常，EPM System 产品在设置过程中使用 Native Directory 和外部用户目录。有关支持的用户目录的列表，请参阅 "[Oracle Enterprise Performance Management System Certification Matrix](#)"。

EPM System 产品要求为访问这些产品的每个用户设立一个用户目录帐户。可将这些用户分配到各个组，以便简化设置。可以为用户和组设置 EPM System 角色和对象 ACL。考虑到管理开销，Oracle 不建议分别设置各个用户。可以在 Oracle Hyperion Shared Services Console 中看到所配置的所有用户目录中的用户和组。

默认情况下，EPM System Configurator 将 Shared Services 存储库配置为 Native Directory 以支持 EPM System 产品。目录管理员使用 Shared Services Console 访问和管理 Native Directory。

与用户目录配置相关的操作

要支持 SSO 和授权，系统管理员必须配置外部用户目录。系统管理员可以从 Oracle Hyperion Shared Services Console 中执行与配置和管理用户目录相关的若干任务。以下主题提供了相关说明：

- 配置用户目录：
 - [配置 OID、Active Directory 和其他基于 LDAP 的用户目录](#)
 - [将关系数据库配置为用户目录](#)
- [测试用户目录连接](#)
- [编辑用户目录设置](#)
- [删除用户目录配置](#)
- [管理用户目录搜索顺序](#)
- [设置安全选项](#)

Oracle Identity Manager 和 EPM System

Oracle Identity Manager 是一个角色与用户管理解决方案，可以自动化在企业资源中添加、更新以及删除用户帐户和属性级权利的整个过程。Oracle Identity Manager 可作为单独的产品或作为 Oracle Identity and Access Management Suite Plus 的组件提供。

Oracle Enterprise Performance Management System 通过使用属于 LDAP 组的企业角色与 Oracle Identity Manager 集成。EPM System 组件的角色可分配给企业角色。添加到 Oracle Identity Manager 企业角色的用户或组会自动继承所分配的 EPM System 角色。

例如，假定您有名为 *Budget Planning* 的一个 Oracle Hyperion Planning 应用程序。为了支持该应用程序，您可以在 Oracle Identity Manager 中创建三个企业角色：Budget Planning Interactive User、Budget Planning End User 和 Budget Planning Admin。设置 EPM System 角色时，请确保设置管理员为 Oracle Identity Manager 中的企业角色设置在 *Budget Planning* 和其他 EPM System 组件（包括 Shared Services）中必需的角色。在 Oracle Identity Manager 中分配给企业角色的所有用户和组均继承 EPM System 角色。有关部署和管理 Oracle Identity Manager 的信息，请参见 Oracle Identity Manager 文档。

要将 Oracle Identity Manager 与 EPM System 进行集成，管理员必须执行以下步骤：

- 确保在某个启用了 LDAP 的用户目录（例如 OID 或 Active Directory）中定义将用于 EPM System 设置的 Oracle Identity Manager 企业角色的成员（用户和组）。
- 配置启用了 LDAP 的用户目录，其中的企业角色成员被定义为 EPM System 中的外部用户目录。请参阅[“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”](#)。

Active Directory 信息

本节介绍此文档中使用的 Microsoft Active Directory 概念。

DNS 查找和主机名查找

系统管理员可以配置 Active Directory，以便 Oracle Hyperion Shared Services 能够执行静态主机名查找或 DNS 查找来识别 Active Directory。静态主机名查找不支持 Active Directory 故障转移。

在多个域控制器上配置了 Active Directory 的情况下，使用 DNS 查找可确保 Active Directory 的高可用性。如果配置为执行 DNS 查找，Shared Services 将查询 DNS 服务器来确定注册的域控制器，并连接到权重最高的域控制器。如果 Shared Services 所连接的域控制器出现故障，Shared Services 将动态切换到权重最高的下一个可用域控制器。

注：

只有在支持故障转移的冗余 Active Directory 设置可用的情况下，才能配置 DNS 查找。有关信息，请参阅 Microsoft 文档。

全局目录

全局目录是存储林中所有 Active Directory 对象的副本的域控制器。全局目录为其主机域存储目录中所有对象的完整副本，而为林中的所有其他域（这些域用于典型的用户搜索操作）存储所有对象的部分副本。有关设置全局目录的信息，请参阅 Microsoft 文档。

如果您的组织在使用全局目录，请使用以下方法之一来配置 Active Directory：

- 将全局目录服务器配置为外部用户目录（推荐方法）。
- 将每个 Active Directory 域配置为独立的外部用户目录。

通过配置全局目录（而不是单个 Active Directory 域），将允许 Oracle Enterprise Performance Management System 产品访问林内的本地和通用组。

配置 OID、Active Directory 和其他基于 LDAP 的用户目录

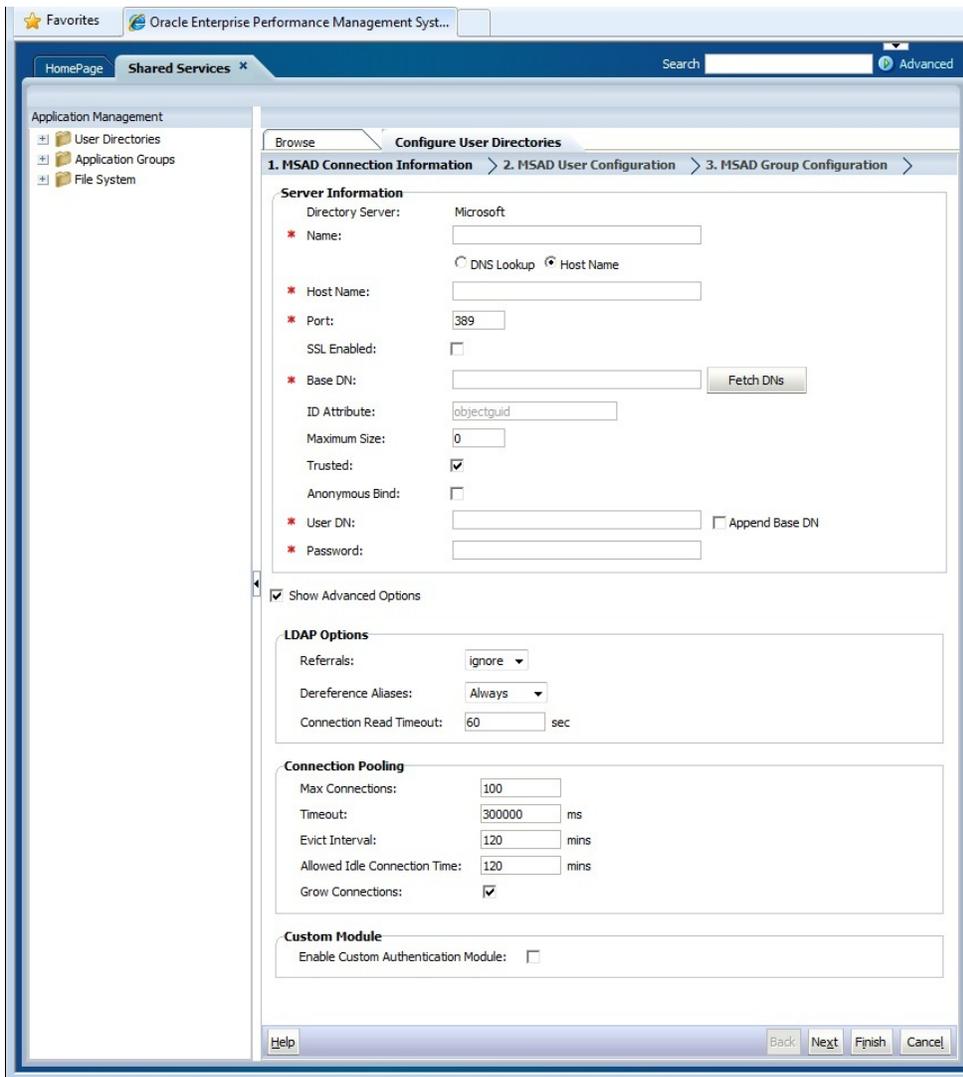
系统管理员使用本节所述的过程来配置基于 LDAP 的企业用户目录，例如 OID、Sun Java System Directory Server、Oracle Virtual Directory、Active Directory、IBM Tivoli Directory Server 或配置屏幕上未列出的其他基于 LDAP 的用户目录。

要配置 OID、Active Directory 和其他基于 LDAP 的用户目录：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。
2. 选择管理，然后选择配置用户目录。
“提供程序配置”选项卡随即打开。该屏幕中列出了所有已配置的用户目录，包括 Native Directory。
3. 单击新建。
4. 在目录类型下，选择一个选项：
 - 选择轻量级目录访问协议 (LDAP) 以配置除 Active Directory 之外的基于 LDAP 的用户目录。选择此选项来配置 Oracle Virtual Directory。
 - 选择 **Microsoft Active Directory (MSAD)** 以配置 Active Directory。

仅限 **Active Directory** 和 **Active Directory 应用程序模式 (ADAM)**：如果您要为 Active Directory 或 ADAM 使用自定义 ID 属性（除 `ObjectGUID` 以外的属性，例如 `sAMAccountName`），请选择轻量级目录访问协议 (**LDAP**)，并将其配置为目录类型其他。

5. 单击下一步。



6. 输入必需的参数。

表 4-1 “连接信息”屏幕

标签	说明
目录服务器	<p>选择一个用户目录。ID 属性值将更改为选定产品的建议恒定唯一标识属性。</p> <p>如果在步骤 4 中选择了 Active Directory，将会自动选中该属性。</p> <p>在下列情况中请选择 Other：</p> <ul style="list-style-type: none"> 您配置的是未列出的用户目录类型，例如 Oracle Virtual Directory 您配置的是已列出的某个支持 LDAP 的用户目录（例如 OID），但您希望使用自定义 ID 属性。 您要将 Active Directory 或 ADAM 配置为使用自定义 ID 属性。
	<p> 注：</p> <p>由于 Oracle Virtual Directory 在一个目录视图中提供 LDAP 目录和 RDMBS 数据存储库的虚拟抽象化形式，因此无论 Oracle Virtual Directory 支持的用户目录的数量和类型如何，Oracle Enterprise Performance Management System 都会将其视为单个外部用户目录。</p>
名称	<p>示例： Oracle Internet Directory</p> <p>用户目录的描述性名称。在配置了多个用户目录的情况下，用于标识特定用户目录。名称不得包含空格和下划线以外的字符。</p> <p>示例： Corporate_OID</p>
DNS 查找	<p>仅限 Active Directory：选择此选项将启用 DNS 查找。请参阅“DNS 查找和主机名查找”。Oracle 建议在生产环境中配置 DNS 查找作为连接到 Active Directory 的方法以避免连接失败。</p>
	<p> 注：</p> <p>如果在配置全局目录，请不要选择此选项。</p>
	<p>选择此选项时，将显示以下字段：</p> <ul style="list-style-type: none"> 域： Active Directory 林的域名。 示例： example.com 或 us.example.com AD 站点： Active Directory 站点名称，通常为存储在 Active Directory 配置容器中的站点对象的相对可分辨名称。一般情况下，AD 站点标识地理位置，如城市、州/省、地区或国家。 示例： Santa Clara 或 US_West_region DNS 服务器： 支持对域控制器进行 DNS 服务器查找的服务器的 DNS 名称。

表 4-1 (续) “连接信息”屏幕

标签	说明
主机名	<p>仅限 Active Directory：选择此选项将启用静态主机名查找。请参阅“DNS 查找和主机名查找”。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> 注： 如果在配置 Active Directory 全局目录，请选择此选项。</p> </div>
主机名	<p>用户目录服务器的 DNS 名称。如果用户目录要用来支持通过 SiteMinder 进行的 SSO，请使用完全限定域名。Oracle 建议仅为测试目的使用主机名建立 Active Directory 连接。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> 注： 如果要配置 Active Directory 全局目录，请指定全局目录服务器主机名。请参阅“全局目录”。</p> </div> <p>示例：MyServer</p>
端口	<p>用户目录在其中运行的端口号。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> 注： 如果要配置 Active Directory 全局目录，请指定全局目录服务器使用的端口（默认值为 3268）。请参阅“全局目录”。</p> </div> <p>示例：389</p>
已启用 SSL	<p>如果选中该复选框，则与此用户目录的通信将启用安全通信。该用户目录必须配置为支持安全通信。</p>
基本 DN	<p>节点的可分辨名称 (DN)，针对用户和组的搜索应从该节点中开始。您也可以使用提取 DN 按钮列出可用的基本 DN，然后从列表中选择适当的基本 DN。</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> 注： 如果要配置全局目录，请指定林的基本 DN。</p> </div> <p>有关特殊字符的使用限制，请参阅“使用特殊字符”。</p> <p>Oracle 建议您选择包含所有 EPM System 产品用户和组的最低 DN。</p> <p>示例：dc=example,dc=com</p>

表 4-1 (续) “连接信息”屏幕

标签	说明
ID 属性	<p>仅当在目录类型中选择了 Other 时才可以修改该属性值。此属性必须是存在于目录服务器上的用户和组对象中的公共属性。</p> <p>系统会自动为该属性设置建议的值：OID orclguid、SunONE (nsuniqueid)、IBM Directory Server (Ibm-entryUuid)、Novell eDirectory (GUID) 和 Active Directory (ObjectGUID)。</p> <p>示例：orclguid</p> <p>如果在目录服务器中选择其他之后手动设置该值（例如，要配置 Oracle Virtual Directory），则 ID 属性值应该：</p> <ul style="list-style-type: none"> 指向一个唯一的属性 不特定于位置 不随时间变化
大小上限	<p>搜索可返回的最大结果数。如果此值大于用户目录设置支持的值，用户目录值将覆盖此值。</p> <p>对于除 Active Directory 之外的其他用户目录，将此字段留空可检索符合搜索标准的所有用户和组。</p> <p>对于 Active Directory，将此值设置为 0 可检索符合搜索标准的所有用户和组。</p> <p>如果在授权管理模式配置 Oracle Hyperion Shared Services，请将此值设置为 0。</p>
受信任	<p>如果选中该复选框，则指明此提供程序是受信任的 SSO 源。来自受信任源的 SSO 令牌不包含用户的密码。</p>
匿名绑定	<p>如果选中该复选框，则指明 Shared Services 可通过匿名方式绑定到用户目录以搜索用户和组。只能在用户目录允许匿名绑定时使用。如果未选择此选项，您必须在“用户 DN”中指定具有足够访问权限的帐户，以便搜索存储用户信息的目录。</p> <p>Oracle 建议您不要使用匿名绑定。</p>
<div style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;">  注： OID 不支持匿名绑定。 </div>	
用户 DN	<p>如果选择了匿名绑定，则此选项处于禁用状态。</p> <p>用户的可分辨名称，Shared Services 应使用此名称与用户目录进行绑定。此用户必须对 DN 内的 RDN 属性具有搜索权限。例如，在 dn: cn=John Doe, ou=people, dc=myCompany, dc=com 中，绑定用户应具有对 cn 属性的搜索访问权限。</p> <p>用户 DN 中的特殊字符必须用转义字符指定。有关限制，请参阅“使用特殊字符”。</p> <p>示例：cn=admin,dc=myCompany,dc=com</p>
附加基本 DN	<p>用于将基本 DN 附加到用户 DN 的复选框。如果要使用目录管理员帐户作为用户 DN，请不要附加基本 DN。</p> <p>如果选择了“匿名绑定”选项，则此复选框处于禁用状态。</p>
密码	<p>用户 DN 密码</p> <p>如果选择了“匿名绑定”选项，则此框处于禁用状态。</p> <p>示例：UserDNpassword</p>

表 4-1 (续) “连接信息”屏幕

标签	说明
显示高级选项	用于显示高级选项的复选框。
参照	仅限 Active Directory: 如果将 Active Directory 配置为跟随参照, 请选择跟随以自动跟随 LDAP 参照。选择忽略以不使用参照。
取消引用别名	选择一种方法, Shared Services 搜索应使用该方法在用户目录中取消引用别名, 以便搜索检索别名的 DN 指向的对象。请选择: <ul style="list-style-type: none"> 总是: 始终取消引用别名。 绝不: 绝不取消引用别名。 正在查找: 仅在名称解析过程中取消引用别名。 正在搜索: 仅在名称解析后取消引用别名。
连接读取超时	LDAP 提供程序因未获得响应而中止 LDAP 读取操作的间隔 (秒)。 默认值: 60 秒
最大连接数	连接池中的最大连接数。对于基于 LDAP 的目录 (包括 Active Directory), 默认值为 100。 默认值: 100
超时	从池中获取连接时的超时。此期间过后将引发异常。 默认值: 300000 毫秒 (5 分钟)
退出间隔	可选: 运行退出进程以清理池的间隔。退出进程将删除超过允许的闲置连接时间的闲置连接。 默认值: 120 分钟
允许的闲置连接时间	可选: 退出进程删除池中的闲置连接之前等待的时间。 默认值: 120 分钟
增多连接	此选项指示连接池中的连接数是否可超过最大连接数。默认情况下处于选定状态。如果不允许连接池增大, 那么, 在为超时设置的时间内没有连接时, 系统将返回错误。
启用自定义身份验证模块	如果选中该复选框, 则可以使用自定义身份验证模块对此用户目录中定义的用户进行身份验证。但必须在“安全选项”屏幕中输入身份验证模块的完全限定 Java 类名称。请参阅“ 设置安全选项 ”。 自定义身份验证模块的身份验证对瘦客户端和胖客户端都是透明的, 不要更改客户端部署。请参阅《Oracle Enterprise Performance Management System 安全配置指南》中的“使用自定义身份验证模块”。

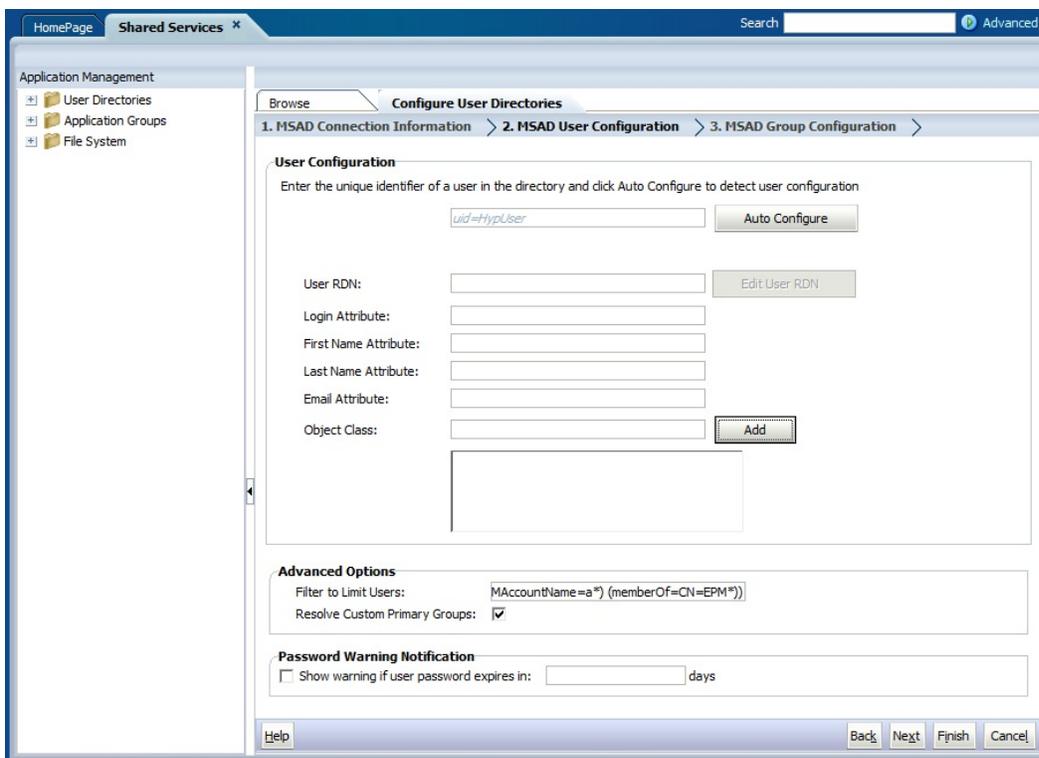
7. 单击下一步。

Shared Services 使用“用户配置”屏幕中设置的属性来创建用户 URL, 该用户 URL 用于确定针对用户的搜索的起始节点。使用此 URL 可加快搜索速度。

▲ 注意:

用户 URL 不应指向别名。EPM System 安全设置要求用户 URL 指向实际用户。

Oracle 建议您使用屏幕的“自动配置”区域来检索所需的信息。



注：

有关可在用户配置中使用的特殊字符的列表，请参阅“[使用特殊字符](#)”。

- 在自动配置中，使用以下格式输入唯一的用户标识符：`attribute=identifier`；例如，`uid=jdoe`。

用户的属性显示在“用户配置”区域中。

如果在配置 OID，您将无法自动配置用户筛选器，因为 OID 的根 DSE 不包含“命名上下文”属性中的条目。请参阅《*Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*》中的“[Managing Naming Contexts](#)”。

注：

您可以手动将必需的用户属性输入“用户配置”区域中的文本框。

表 4-2 “用户配置”屏幕

标签	说明 ¹
用户 RDN	用户的相对 DN。DN 的每个组成部分都称为一个 RDN，并表示目录树中的一个分支。用户的 RDN 通常相当于 <code>uid</code> 或 <code>cn</code> 。 有关限制，请参阅“ 使用特殊字符 ”。 示例： <code>ou=People</code>

表 4-2 (续) “用户配置”屏幕

标签	说明 ¹
登录属性	<p>用于存储用户登录名的唯一属性（也可以是自定义属性）。在登录到 EPM System 产品时，用户使用此属性的值作为用户名。用户 ID（登录属性的值）在所有用户目录中必须唯一。例如，您可以分别使用 uid 和 sAMAccountName 作为 SunONE 和 Active Directory 配置的登录属性。这些属性的值在所有用户目录（包括 Native Directory）中必须唯一。</p> <div style="border: 1px solid #0070c0; padding: 10px; margin: 10px 0;"> <p> 注： 用户 ID 不区分大小写。</p> </div> <div style="border: 1px solid #0070c0; padding: 10px; margin: 10px 0;"> <p> 注： 如果针对 Kerberos 环境中 Oracle Application Server 上部署的 EPM System 产品将 OID 配置为外部用户目录，那么您必须将此属性设置为 userPrincipalName。</p> </div> <p>默认值</p> <ul style="list-style-type: none"> • Active Directory: : cn • 除 Active Directory 外的 LDAP 目录: uid
名字属性	<p>用于存储用户的名字的属性 默认值: givenName</p>
姓氏属性	<p>用于存储用户的姓氏的属性 默认值: sn</p>
电子邮件属性	<p>可选: 用于存储用户电子邮件地址的属性 默认值: mail</p>

表 4-2 (续) “用户配置”屏幕

标签	说明 ¹
对象类	<p>用户的对象类（可与用户关联的必需和可选属性）。Shared Services 在搜索筛选器中使用此屏幕中列出的对象类。使用这些对象类，Shared Services 应可找到应加以设置的所有用户。</p> <div style="border: 1px solid #0070c0; padding: 10px; margin: 10px 0;"> <p> 注： 如果您要将 Active Directory 或 ADAM 配置为用户目录类型其他以使用自定义 ID 属性，则必须将该值设置为 user。</p> </div> <p>如果需要，您可以手动添加对象类。要添加对象类，请在对象类框中输入对象类名，然后单击添加。 要删除对象类，请选择对象类并单击删除。</p> <p>默认值</p> <ul style="list-style-type: none"> • Active Directory: user • 除 Active Directory 外的 LDAP 目录: person、organizationalPerson、inetorgperson
用于限制用户的筛选器	<p>一个 LDAP 查询，该查询仅检索要设置为具有 EPM System 产品角色的用户。例如，LDAP 查询 (uid=Hyp*) 仅检索其名称以 Hyp 开头的用户。</p> <p>“用户配置”屏幕验证用户 RDN，并建议使用用户筛选器（如果需要）。</p> <p>用户筛选器用于限制查询过程中返回的用户数量。如果用户 RND 标识的节点包含大量无需进行设置的用户，则用户筛选器特别有用。用户筛选器可用于排除不需要进行设置的用户，从而提高性能。</p>
多属性 RDN 的用户搜索属性	<p>仅限 Active Directory 以外的启用了 LDAP 的用户目录：仅当目录服务器配置为使用多属性 RDN 时才应设置此值。您设置的值必须是 RDN 属性之一。您指定的属性值必须唯一并且该属性可以搜索。例如，假定 SunONE 目录服务器配置为合并 cn (cn=John Doe) 和 uid (uid=jDoe12345) 属性来创建类似下面的多属性 RDN：</p> <pre>cn=John Doe+uid=jDoe12345, ou=people, dc=myCompany, dc=com</pre> <p>在这种情况下，如果这些属性满足以下条件，则可以使用 cn 或 uid：</p> <ul style="list-style-type: none"> • “连接信息”选项卡上的“用户 DN”字段中标识的用户可以搜索该属性 • 该属性要求在用户目录中设置唯一值
解析自定义主要组	<p>仅限 Active Directory：此复选框表示是否标识主要用户组，以便确定有效角色。此复选框在默认情况下处于选中状态。Oracle 建议不要更改这一设置。</p>
如果用户密码在以下天数内失效，则显示警告：	<p>仅限 Active Directory：此复选框表示 Active Directory 用户密码在指定天数内过期时是否显示警告消息。</p>

¹ EPM System 安全性可能会在配置值为可选的一些字段中使用默认值。如果未在这类字段中输入值，则在运行时期将使用默认值。

9. 单击下一步。

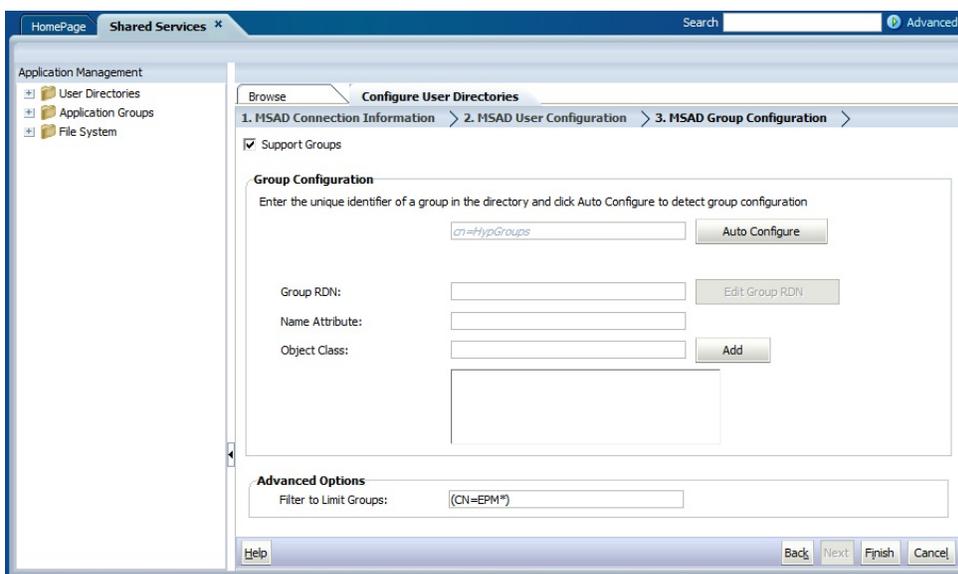
此时将打开“组配置”屏幕。Shared Services 将使用此屏幕中设置的属性来创建组 URL，该 URL 用于确定针对组的搜索的起始节点。使用此 URL 可加快搜索速度。

注意：

组 URL 不应指向别名。EPM System 安全设置要求组 URL 指向实际组。如果要配置使用组别名的 Novell eDirectory，组 URL 内必须有可用的组别名和组帐户。

注：

可以选择是否在“组配置”屏幕中输入数据。如果不输入组 URL 设置，Shared Services 将在基本 DN 内搜索以查找组，从而可能会对性能产生负面影响，特别是在用户目录包含多个组的情况下尤为如此。



10. 如果您的组织不打算设置组，或者未在用户目录中将用户归类到组中，请清除支持组。如果清除此选项，将禁用此屏幕上的字段。

如果要支持组，Oracle 建议您使用自动配置功能来检索所需的信息。

如果将 OID 配置为用户目录，您将无法使用自动配置功能，因为 OID 的根 DSE 不包含“命名上下文”属性中的条目。请参阅《Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory》中的“[Managing Naming Contexts](#)”。

11. 在自动配置文本框中，输入唯一的组标识符，然后单击启动。

必须采用以下格式表示组标识符：属性=标识符；例如，cn=western_region。

组的属性显示在“组配置”区域中。

 **注：**

您可以在“组配置”文本框中输入所需的组属性。

 **注意：**

如果没有为节点名称中包含 / (斜杠) 或 \ (反斜杠) 的用户目录设置组 URL，针对用户和组的搜索将会失败。例如，如果没有为节点中用户和组所在的用户目录（例如，OU=child\ou,OU=parent/ou 或 OU=child/ou,OU=parent \ ou）指定组 URL，任何列举用户或组的操作都将失败。

表 4-3 “组配置”屏幕

标签	说明 ¹
组 RDN	<p>组的“相对 DN”。该值为相对于基本 DN 的路径，用作组 URL。指定用于标识最低用户目录节点的组 RDN，该节点中包含您打算设置的所有组。</p> <p>如果使用 Active Directory 主要组进行设置，请确保主要组位于“组 RDN”中。如果它在组 URL 的范围之外，Shared Services 将不会检索主要组。</p> <p>组 RDN 对登录和搜索性能的影响很大。由于它是所有组搜索的起始点，因此，您必须确定其中包含 EPM System 产品所有组的最低可能的节点。要保证最佳性能，组 RDN 中存在的组的数量不应超过 10,000。如果存在更多组，请使用组筛选器，以便只检索您想要设置的组。</p>
名称属性	<p> 注：</p> <p>如果组 URL 内可用组的数量超过 10,000，Shared Services 将显示警告。</p> <p>有关限制，请参阅“使用特殊字符”。</p> <p>示例：ou=Groups</p> <p>用于存储组的名称的属性 默认值</p> <ul style="list-style-type: none"> • 包括 Active Directory 在内的 LDAP 目录： cn • Native Directory： cssDisplayNameDefault

表 4-3 (续) “组配置”屏幕

标签	说明 ¹
对象类	<p>组的对象类。Shared Services 在搜索筛选器中使用此屏幕中列出的对象类。使用这些对象类，Shared Services 应可找到与用户关联的所有组。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin: 10px 0;"> <p> 注:</p> <p>如果您要将 Active Directory 或 ADAM 配置为用户目录类型其他以使用自定义 ID 属性，则必须将该值设置为 group?member。</p> </div> <p>如果需要，您可以手动添加对象类。要添加对象类，请在“对象类”文本框中输入对象类名，然后单击添加。 要删除对象类，请选择对象类，然后单击删除。</p> <p>默认值</p> <ul style="list-style-type: none"> • Active Directory: group?member • 除 Active Directory 外的 LDAP 目录: groupofuniquenames?uniquemember, groupOfNames?member • Native Directory: groupofuniquenames?uniquemember, cssGroupExtend?cssIsActive
用于限制组的筛选器	<p>一个 LDAP 查询，该查询仅检索要设置为具有 EPM System 产品角色的组。例如，LDAP 查询 ((cn=Hyp*)(cn=Admin*)) 仅检索其名称以 Hyp 或 Admin 开头的组。</p> <p>如果组 RND 标识的节点包含大量无需进行设置的组，则用于限制查询返回的组数量的组筛选器特别重要。筛选器可以用于排除不需要进行设置的组，从而提高性能。</p> <p>如果使用 Active Directory 主要组进行设置，请确保所设置的任何组筛选器都可检索组 URL 范围内的主要组。例如，筛选器 ((cn=Hyp*)(cn=Domain Users)) 将检索名称以 Hyp 开头的组以及名称为 Domain Users 的主要组。</p>

¹ EPM System 安全性可能会在配置值为可选的一些字段中使用默认值。如果未在这类字段中输入值，则在运行时期间将使用默认值。

12. 单击完成。

Shared Services 将保存配置并返回到“定义的用户目录”屏幕，该屏幕现在将列出您配置的用户目录。

13. 测试配置。 请参阅“[测试用户目录连接](#)”。

14. 如果需要，更改分配的搜索顺序。 有关详细信息，请参阅“[管理用户目录搜索顺序](#)”。

15. 如果需要，请指定安全选项。 有关详细信息，请参阅“[设置安全选项](#)”。

16. 重新启动 Oracle Hyperion Foundation Services 和其他 EPM System 组件。

将关系数据库配置为用户目录

可以使用 Oracle、SQL Server 和 IBM DB2 关系数据库的系统表中的用户和组信息来支持设置。如果无法从数据库的系统架构中派生组信息，则 Oracle Hyperion Shared Services 不支持对该数据库提供程序中的组进行设置。例如，Shared Services 无法从旧版本的 IBM DB2 中提取组信息，原因是数据库使用操作系统中定义的组。不过，设置管理员可以将这些用户添加到 Native Directory 内的组中并设置这些组。有关支持的平台信息，请参阅 Oracle 技术网 (OTN) 上 "[Oracle Fusion Middleware Supported System Configurations](#)" 页面上发布的 "Oracle Enterprise Performance Management System Certification Matrix"。

注：

如果您使用 DB2 数据库，则用户名必须至少包含 8 个字符。对于 Oracle 和 SQL Server 数据库而言，用户名不能超过 256 个字符；对于 DB2 而言，用户名不能超过 1000 个字符。

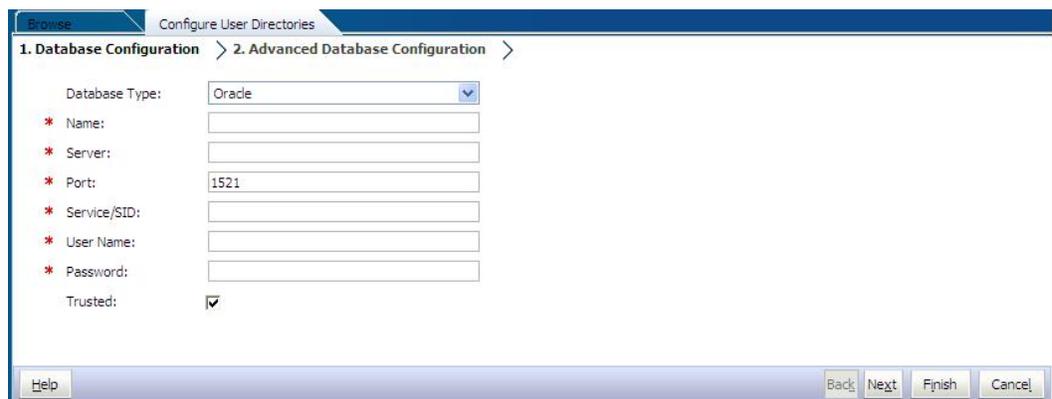
配置 Shared Services 以便以数据库管理员身份（例如，Oracle SYSTEM 用户）连接到数据库来检索用户和组的列表。

注：

Shared Services 只检索用于设置的活动数据库用户。将忽略不活动和锁定的数据库用户帐户。

要配置数据库提供程序：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。
2. 选择管理，然后选择配置用户目录。
3. 单击新建。
4. 在目录类型屏幕中，选择关系数据库 (Oracle、DB2、SQL Server)。
5. 单击下一步。



The screenshot shows a configuration window titled "Configure User Directories" with a sub-tab "2. Advanced Database Configuration". The "Database Type" is set to "Oracle". The following fields are present:

- Name: (empty text box)
- Server: (empty text box)
- Port: 1521
- Service/SID: (empty text box)
- User Name: (empty text box)
- Password: (empty text box)
- Trusted:

Buttons at the bottom include "Help", "Back", "Next", "Finish", and "Cancel".

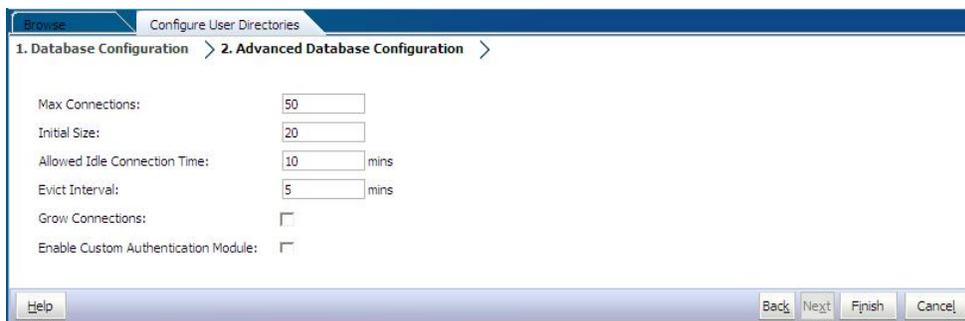
6. 在“数据库配置”选项卡上，输入配置参数。

表 4-4 “数据库配置”选项卡

标签	说明
数据库类型	关系数据库提供程序。Shared Services 只支持 Oracle 和 SQL Server 数据库作为数据库提供程序。 示例：Oracle
名称	数据库提供程序的唯一配置名称。 示例：Oracle_DB_FINANCE
服务器	运行数据库服务器的计算机的 DNS 名称。 示例：myserver
端口	数据库服务器端口号 示例：1521
服务/SID (仅限 Oracle)	系统标识符 (默认值为 orcl) 示例：orcl
数据库 (仅限 SQL Server 和 DB2)	Shared Services 应连接的数据库 示例：master
用户名	Shared Services 访问数据库时应当使用的用户名。此数据库用户必须拥有对数据库系统表的访问权限。Oracle 建议您对 Oracle 数据库使用 system 帐户，对 SQL Server 数据库使用数据库管理员的用户名。 示例：SYSTEM
密码	用户名中所标识用户的密码。 示例：system_password
受信任	用于指定此提供程序是受信任 SSO 源的复选框。来自受信任源的 SSO 令牌不包含用户的密码。

7. 可选：单击下一步以配置连接池。

“高级数据库配置”选项卡将打开。



8. 在“高级数据库配置”中，输入连接池参数。

表 4-5 “高级数据库配置”选项卡

标签	说明
最大连接数	池中的最大连接数。默认值为 50。
初始大小	初始化池时的可用连接数。默认值为 20。
允许的闲置连接时间	可选：退出进程删除池中的闲置连接之前等待的时间。默认值为 10 分钟。
退出间隔	可选：运行退出进程以清理池的时间间隔。退出进程将删除超过允许的闲置连接时间的闲置连接。默认值为 5 分钟。
增多连接	指示连接池中的连接数是否可超过最大连接数量。默认情况下，此选项处于清除状态，表示池不能增大。如果不允许连接池增大，那么，在为超时设置的时间内没有连接时，系统将返回错误。
启用自定义身份验证模块	如果选中该复选框，则可以使用自定义身份验证模块对此用户目录中定义的用户进行身份验证。但必须在“安全选项”屏幕中输入身份验证模块的完全限定 Java 类名称。请参阅“ 设置安全选项 ”。自定义身份验证模块的身份验证对瘦客户端和胖客户端都是透明的。请参阅《Oracle Enterprise Performance Management System 安全配置指南》中的“使用自定义身份验证模块”。

9. 单击完成。
10. 单击确定返回到“定义的用户目录”屏幕。
11. 测试数据库提供程序的配置。请参阅“[测试用户目录连接](#)”。
12. 如果需要，请更改分配的搜索顺序。有关详细信息，请参阅“[管理用户目录搜索顺序](#)”。
13. 如果需要，指定安全性设置。请参阅“[设置安全选项](#)”。
14. 重新启动 Oracle Hyperion Foundation Services 和其他 Oracle Enterprise Performance Management System 组件。

测试用户目录连接

配置用户目录之后，测试连接以确保 Oracle Hyperion Shared Services 可使用当前设置连接到用户目录。

要测试用户目录连接：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。
2. 选择管理，然后选择配置用户目录。
3. 从用户目录列表中，选择要测试的外部用户目录配置。
4. 依次单击测试和确定。

编辑用户目录设置

管理员可以修改除名称外的任何用户目录配置参数。Oracle 不建议编辑用于设置的用户目录的配置数据。

 **注意：**

编辑用户目录配置中的某些设置（例如 ID Attribute）会使设置数据失效。在修改已设置的用户目录的设置时，请务必谨慎。

要编辑用户目录配置：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅[“启动 Shared Services Console”](#)。
2. 选择管理，然后选择配置用户目录。
3. 选择要编辑的用户目录。
4. 单击编辑。
5. 修改配置设置。

 **注：**

您不能修改配置名称。如果在修改 LDAP 用户目录配置，您可以从“目录服务器”列表中选择其他目录服务器或 Other（用于自定义 LDAP 目录）。无法编辑 Native Directory 参数。

有关可编辑的参数的说明，请参阅以下各表：

- Active Directory 和其他基于 LDAP 的用户目录：请参阅[“配置 OID、Active Directory 和其他基于 LDAP 的用户目录”](#)中的表。
 - 数据库：请参阅[“将关系数据库配置为用户目录”](#)中的表。
6. 单击确定以保存更改。

删除用户目录配置

系统管理员可以随时删除外部用户目录配置。删除目录配置将使派生自该用户目录的用户和组的所有设置信息失效，并从搜索顺序中删除该目录。

 **提示：**

如果不想使用已用于设置的已配置用户目录，请将其从搜索顺序中删除，以便在查找用户和组时不对其进行搜索。此操作将保持设置信息的完整性，并使您以后还能使用该用户目录。

要删除用户目录配置：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅[“启动 Shared Services Console”](#)。
2. 选择管理，然后选择配置用户目录。
3. 选择一个目录。

4. 单击删除。
5. 单击确定。
6. 再次单击确定。
7. 重新启动 Oracle Hyperion Foundation Services 和其他 Oracle Enterprise Performance Management System 组件。

管理用户目录搜索顺序

在系统管理员配置外部用户目录时，Oracle Hyperion Shared Services 会自动将用户目录添加到搜索顺序中，并向其分配在 Native Directory 之前的下一个可用搜索序号。当 Oracle Enterprise Performance Management System 搜索用户和组时，将根据该搜索顺序在已配置的用户目录中循环搜索。

系统管理员可以将用户目录从搜索顺序中移除，在这种情况下，Shared Services 会自动重新分配剩余目录的搜索顺序。搜索顺序中未包含的用户目录将不用于支持身份验证和设置。

注：

当 Shared Services 遇到指定帐户时会终止对用户或组的搜索。Oracle 建议将包含大多数 EPM System 用户的企业目录放在搜索顺序的顶部。

默认情况下，Native Directory 设置为搜索顺序中的最后一个目录。管理员可以执行以下任务来管理搜索顺序：

- [向搜索顺序中添加用户目录](#)
- [更改搜索顺序](#)
- [删除分配的搜索顺序](#)

向搜索顺序中添加用户目录

新配置的用户目录将自动添加到搜索顺序。如果从搜索顺序中删除了某个目录，您可以将其添加到搜索顺序的末尾。

要向搜索顺序中添加用户目录：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。
2. 选择管理，然后选择配置用户目录。
3. 选择一个禁用的用户目录添加到搜索顺序中。
4. 单击包括。
只有当选择了不在搜索顺序中的用户目录时，此按钮才可用。
5. 单击确定返回到“定义的用户目录”屏幕。
6. 重新启动 Oracle Hyperion Foundation Services 和其他 EPM System 组件。

删除分配的搜索顺序

从搜索顺序中删除用户目录不会使目录配置失效；该操作会从为验证用户身份而搜索的目录的列表中删除该目录。未包括在搜索顺序中的目录将设置为已禁用状态。当管理员从搜索顺序中删除用户目录时，分配给其他用户目录的搜索序列将自动更新。



注：

无法从搜索顺序中删除 Native Directory。

要从搜索顺序中删除用户目录：

1. 以系统管理员身份访问 Shared Services Console。请参阅[“启动 Shared Services Console”](#)。
2. 选择管理，然后选择配置用户目录。
3. 选择要从搜索顺序中删除的目录。
4. 单击排除。
5. 单击确定。
6. 在“目录配置结果”屏幕上，单击确定。
7. 重新启动 Foundation Services 和其他 EPM System 组件。

更改搜索顺序

分配给每个用户目录的默认搜索顺序取决于配置目录时所采用的顺序。默认情况下，Native Directory 设置为搜索顺序中的最后一个目录。

要更改搜索顺序：

1. 以系统管理员身份访问 Shared Services Console。请参阅[“启动 Shared Services Console”](#)。
2. 选择管理，然后选择配置用户目录。
3. 选择要更改其搜索顺序的目录。
4. 单击上移或下移。
5. 单击确定。
6. 重新启动 Foundation Services、其他 EPM System 组件和使用 Shared Services 安全 API 的自定义应用程序。

设置安全选项

安全选项由适用于搜索顺序中包括的所有用户目录的全局参数组成。

要设置安全选项：

1. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅[“启动 Shared Services Console”](#)。
2. 选择管理，然后选择配置用户目录。

3. 选择安全选项。
4. 在安全选项中，设置全局参数。

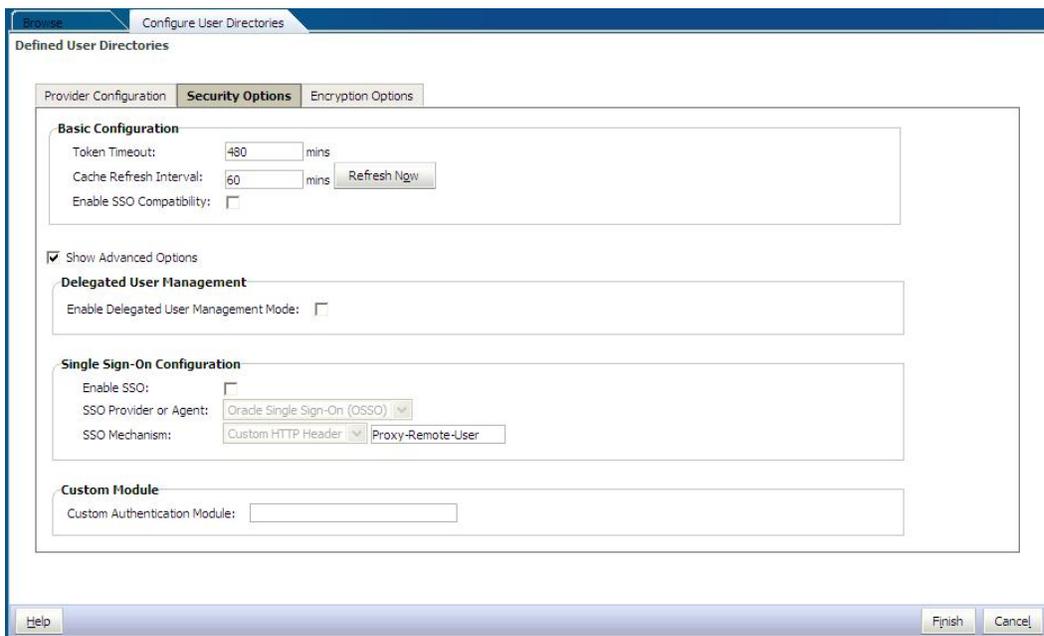


表 4-6 适用于用户目录的安全选项

参数	说明
令牌超时	Oracle Enterprise Performance Management System 产品或 Web 标识管理解决方案发出的 SSO 令牌过期的时间（分钟）。此期间过后，用户必须再次登录。令牌超时基于服务器的系统时钟设置。默认值为 480 分钟。
	<p> 注： 令牌超时与会话超时不同。</p>
高速缓存刷新间隔	刷新包含组与用户关系数据的 Oracle Hyperion Shared Services 高速缓存的间隔（分钟）。默认值为 60 分钟。 Shared Services 仅在下一次高速缓存刷新之后才缓存有关新外部用户目录组和添加到现有组的新用户的信息。只有在高速缓存刷新后，通过新建的外部用户目录组设置的用户才能获取其设置的角色。
立即刷新	单击此按钮可手动启动 Shared Services 高速缓存刷新，该高速缓存包含组与用户关系数据。您可能希望在外用户目录中创建新组并对其设置之后或向现有组添加新用户之后启动高速缓存刷新。仅在 Shared Services 执行使用高速缓存中数据的调用操作之后，才刷新高速缓存。
启用 SSO 兼容性	如果您的部署已与 Oracle Business Intelligence Enterprise Edition 11.1.1.5 或更早版本集成，请选择该选项。

表 4-6 (续) 适用于用户目录的安全选项

参数	说明
启用授权用户管理模式	使 EPM System 产品的授权用户管理能够支持分布式设置活动管理的选项。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“授权用户管理”。
启用 SSO	实现对来自安全代理（比如 Oracle Access Manager）的 SSO 的支持的选项
SSO 提供程序或代理	<p>选择 EPM System 产品应从中接受 SSO 的 Web 标识管理解决方案。如果未列出您的 Web 标识管理解决方案（例如 Kerberos），请选择其他。</p> <p>当您选择 SSO 提供程序时，首选的 SSO 机制和名称会自动处于选定状态。如果需要，您可以更改 SSO 机制（HTTP 头或自定义登录类）的名称。</p> <p>如果选择 Other 作为 SSO 提供程序或代理，您必须确保它支持受 EPM System 支持的某种 SSO 机制。请参阅《Oracle Enterprise Performance Management System 安全配置指南》中的“支持的 SSO 方法”。</p>
SSO 机制	<p>选择的 Web 标识管理解决方案用来向 EPM System 产品提供用户登录名的方法。有关可接受的 SSO 方法的说明，请参阅《Oracle Enterprise Performance Management System 安全配置指南》中的“支持的 SSO 方法”。</p> <ul style="list-style-type: none"> Custom HTTP Header: 设置安全代理传递给 EPM System 的 HTTP 头名称。 Custom Login Class: 指定处理 HTTP 身份验证请求的自定义 Java 类。请参阅《Oracle Enterprise Performance Management System 安全配置指南》中的“自定义登录类”。 <div style="border: 1px solid #0070C0; padding: 5px; margin: 10px 0;"> <p> 注: 自定义登录类不同于自定义身份验证。</p> </div> <ul style="list-style-type: none"> HTTP Authorization Header: 标准的 HTTP 机制。 Get Remote User from HTTP Request: 安全代理在 HTTP 请求中填充远程用户时，选择此选项。

表 4-6 (续) 适用于用户目录的安全选项

参数	说明
自定义身份验证模块	<p>自定义身份验证模块的完全限定的 Java 类名称 (例如 <code>com.mycompany.epm.CustomAuthenticationImpl</code>)。对于选择了自定义身份验证模块的所有用户目录, 将使用该名称对其中的用户进行身份验证。</p> <p>只有当目录配置启用了身份验证模块时 (默认启用), 才会对用户目录使用身份验证模块。</p> <p>Oracle Hyperion Foundation Services 要求将自定义身份验证 JAR 文件命名为 <code>CustomAuth.jar</code>。<code>CustomAuth.jar</code> 必须包含在 <code>MIDDLEWARE_HOME\user_projects\domains\WEBLOGIC_DOMAIN\lib</code> (通常为 <code>C:\Oracle\Middleware\user_projects\domains\EPMSysstem\lib</code>) 中。</p> <p>在所有客户端安装中, <code>CustomAuth.jar</code> 必须位于 <code>EPM_ORACLE_HOME\common\jlib\11.1.2.0</code> (通常为 <code>C:\Oracle\Middleware\EPMSysstem11R1\common\jlib\11.1.2.0</code>) 中。</p> <p>可以在 JAR 文件中使用任何包结构和类名称。</p> <p>有关详细信息, 请参阅《Oracle Enterprise Performance Management System 安全配置指南》中的“使用自定义身份验证模块”。</p>

- 单击确定。
- 重新启动 Foundation Services 和其他 EPM System 组件。

重新生成加密密钥

Oracle Enterprise Performance Management System 使用以下密钥确保安全性:

- 单点登录令牌加密密钥, 用于加密和解密 EPM System SSO 令牌。该密钥存储在 Oracle Hyperion Shared Services Registry 中
- 可信服务密钥, EPM System 组件使用它来验证请求 SSO 令牌的服务的可靠性
- 提供程序配置加密密钥, 用于加密 EPM System 安全性与配置的外部用户目录进行绑定所使用的密码 (启用 LADP 的用户目录的用户 DN 密码)。该密码是在配置外部用户目录时设置的。

应定期更改这些密钥以增强 EPM System 的安全性。Oracle Hyperion Shared Services 和 EPM System 的安全子系统使用 128 位密钥强度的 AES 加密。

▲ 注意:

当您重新生成单点加密密钥时, Oracle Hyperion Financial Management 和 Oracle Hyperion Profitability and Cost Management 使用的任务流将会失效。重新生成密钥后, 请打开并保存任务流, 以使它们重新生效。

要重新生成单点登录加密密钥、提供程序配置密钥或可信服务密钥:

- 以系统管理员身份访问 Oracle Hyperion Shared Services Console。请参阅“[启动 Shared Services Console](#)”。

2. 选择管理，然后选择配置用户目录。
3. 选择加密选项。
4. 在加密选项中，选择要重新生成的密钥。

表 4-7 EPM System 加密选项

选项	说明
单点登录令牌	<p>选择此选项可重新生成用于加密和解密 EPM System SSO 令牌的加密密钥。</p> <p>如果在安全选项上选择了启用 SSO 兼容性，请选择以下按钮之一：</p> <ul style="list-style-type: none"> • 生成新密钥，创建新的 SSO 令牌加密密钥 • 重置为默认值，恢复默认的 SSO 令牌加密密钥
可信服务密钥	<p>选择此选项可重新生成可信身份验证密钥，EPM System 组件使用它验证请求 SSO 令牌的服务的可靠性。</p>
提供程序配置密钥	<p>选择此选项可重新生成用于加密 EPM System 安全性与配置的外部用户目录进行绑定所使用的密码（启用 LDAP 的用户目录的用户 DN 密码）的密钥。该密码是在配置外部用户目录时设置的。</p>

 注：

如果恢复为默认加密密钥，必须从所有 EPM System 主机中删除现有密钥存储文件 (`EPM_ORACLE_HOME/common/CSS/ssHandlerTK`)。

5. 单击确定。
6. 如果您选择生成新的 SSO 加密密钥，请完成此步骤。
 - a. 单击下载。
 - b. 单击确定，将支持新 SSO 加密密钥的密钥库文件 `ssHandlerTK` 保存到 Oracle Hyperion Foundation Services 所在服务器上的文件夹中。
 - c. 在所有的 EPM System 主机上将 `ssHandlerTK` 复制到 `EPM_ORACLE_HOME/common/CSS` 中。
7. 重新启动 Foundation Services 和其他 EPM System 组件。

使用特殊字符

Active Directory 和其他基于 LDAP 的用户目录允许在诸如 DN、用户名、角色和组名等实体中使用特殊字符。可能需要进行特殊处理才能使 Oracle Hyperion Shared Services 识别此类字符。

通常情况下，在用户目录设置（例如，基本 DN 以及用户 URL 和组 URL）中指定特殊字符时必须使用转义字符。下表列出了可在用户名、组名、用户 URL、组 URL 和用户 DN 的 OU 值中使用的特殊字符。

表 4-8 支持的特殊字符

字符	名称或含义	字符	名称或含义
(左括号	\$	美元
)	右括号	+	加号
"	双引号	&	& 符号
'	单引号	\	反斜杠
,	逗号	^	三角符号
=	等于号	;	分号
<	小于号	#	井号
>	大于号	@	at

 注:

不得在基本 DN 内的组织单位名称中使用 / (斜杠)

- 不允许在“登录用户”属性的值中使用特殊字符。
- 不支持在用户名、组名称、用户和组 URL 以及用户 DN 中 OU 的名称中使用星号 (*)。
- 不支持含有特殊字符组合的属性值。
- 可以不用转义字符而直接使用 & 符号。对于 Active Directory 设置，必须将 & 指定为 &#38;。
- 用户和组名称不能同时包含反斜杠 (\) 和斜杠 (/)。例如，不支持诸如 test/\user 和 new\test/user 之类的名称。

表 4-9 不需要转义的字符

字符	名称或含义	字符	名称或含义
(左括号	'	单引号
)	右括号	^	三角符号
\$	美元	@	at
&	& 符号		

 注:

& 必须指定为 &#38;。

如果在用户目录设置（用户名、组名称、用户 URL、组 URL 和用户 DN）中使用这些字符，则必须对字符进行转义。

表 4-10 用户目录配置设置中特殊字符的转义

特殊字符	转义	示例设置	转义示例
逗号 (,)	反斜杠 (\)	ou=test,ou	ou=test\,ou
加号 (+)	反斜杠 (\)	ou=test+ou	ou=test\+ou
等于号 (=)	反斜杠 (\)	ou=test=ou	ou=test\=ou
井号 (#)	反斜杠 (\)	ou=test#ou	ou=test\#ou
分号 (;)	反斜杠 (\)	ou=test;ou	ou=test\;ou
小于号 (<)	反斜杠 (\)	ou=test<ou	ou=test\<>ou
大于号 (>)	反斜杠 (\)	ou=test>ou	ou=test\>ou
双引号 (")	两个反斜杠 (\)	ou=test"ou	ou=test\\"ou
反斜杠 (\)	三个反斜杠 (\)	ou=test\ou	ou=test\\ou

 注:

- 在用户 DN 中，必须用一个反斜杠对双引号 (") 进行转义。例如，必须将 ou=test"ou 指定为 ou=test\"ou。
- 在用户 DN 中，必须用一个反斜杠对反斜杠 (\) 进行转义。例如，必须将 ou=test\ou 指定为 ou=test\\ou。

 注意:

如果未指定用户 URL，则在 RDN 根内创建的用户不得包含 / (斜杠) 或 \ (反斜杠)。同样，如果未指定组 URL，则不得在 RDN 根内创建的组的名称中使用这些字符。例如，不支持诸如 OU=child\ou,OU=parent/ou 或 OU=child/ou,OU=parent\ou 等组名称。如果使用唯一属性作为用户目录配置中的 ID 属性，则不存在此问题。

Native Directory 中的特殊字符

支持在 Native Directory 用户名和组名中使用特殊字符。

表 4-11 支持的特殊字符：Native Directory

字符	名称或含义	字符	名称或含义
@	at	,	逗号
#	井号	=	等于号
\$	美元	+	加号
^	三角符号	;	分号
(左括号	!	叹号
)	右括号	%	百分号

表 4-11 (续) 支持的特殊字符: Native Directory

字符	名称或含义	字符	名称或含义
'	单引号		

5

使用自定义身份验证模块

另请参阅：

- [概览](#)
- [用例示例和限制](#)
- [先决条件](#)
- [设计和编码注意事项](#)
- [部署自定义身份验证模块](#)

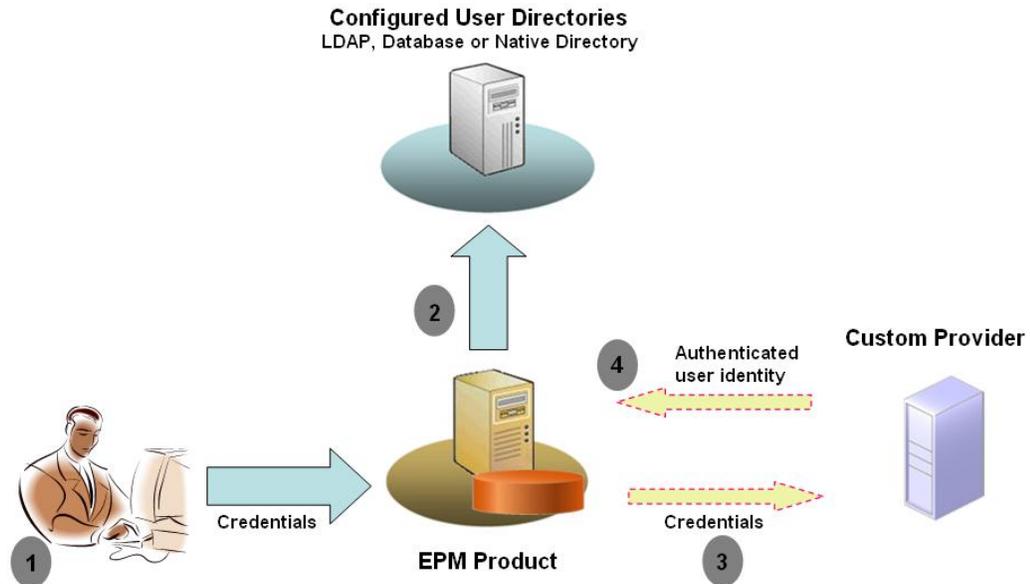
概览

自定义身份验证模块是客户开发和实施以对 Oracle Enterprise Performance Management System 用户进行身份验证的 Java 模块。通常，EPM System 产品使用登录屏幕来捕获用于验证用户身份的用户名和密码。不必使用 EPM System 身份验证，您可以使用自定义身份验证模块来验证用户身份，并将经验证的用户凭据传递给 EPM System 以进一步处理。实施自定义身份验证模块不涉及修改 EPM System 产品。

您可以将自定义身份验证模块用于胖客户端（例如 Oracle Smart View for Office 和 Oracle Essbase Studio）和瘦客户端（例如 Oracle Hyperion Enterprise Performance Management Workspace）。

自定义身份验证模块使用用户在登录到 EPM System 产品时输入的信息。如果为用户目录启用了自定义身份验证模块，则通过它对用户进行身份验证。成功对用户进行身份验证之后，自定义身份验证模块会将用户名返回给 EPM System。

以下图例显示了一个自定义身份验证方案示例：



例如，您可以使用 RSA SecurID 基础结构作为自定义提供程序，以确保对 EPM System 执行透明的强身份验证。概述：

1. 用户输入凭据（通常是用户名和密码）以访问 EPM System 产品。这些凭据应当向自定义身份验证模块使用的提供程序唯一标识用户。例如，如果您使用 RSA SecurID 基础结构对用户进行身份验证，则用户需输入 RSA 用户 ID 和 PIN（而非 EPM System 用户 ID 和密码）。
2. 使用搜索顺序（请参阅“[搜索顺序](#)”），EPM System 可循环搜索配置的用户目录来定位用户。
 - 如果没有为当前用户目录配置自定义身份验证，EPM System 将尝试通过 EPM System 身份验证来定位用户并对其进行身份验证。
 - 如果为用户目录配置了自定义身份验证，则 EPM System 将把身份验证过程授权给自定义模块。
3. 如果 EPM System 将身份验证授权给了自定义模块，则自定义身份验证模块将接受凭据并使用自己的逻辑指示根据自定义提供程序（例如 RSA SecurID 基础结构）对用户进行身份验证。
4. 自定义身份验证模块根据提供程序对用户进行身份验证后，会将用户名返回给 EPM System 或者返回 Java 异常。

自定义身份验证模块返回的用户名必须与启用了自定义身份验证的用户目录中的用户名完全相同。

- 如果自定义身份验证模块返回用户名，说明 EPM System 在已启用自定义身份验证的用户目录中找到了用户。在此阶段，EPM System 不会搜索未配置自定义身份验证的用户目录。
- 如果自定义身份验证模块抛出异常或返回空用户，则 EPM System 将继续按搜索顺序在未启用自定义身份验证的剩余用户目录中搜索用户。如果找不到凭据匹配的用户，EPM System 将显示错误。

用例示例和限制

自定义身份验证实施方案包括以下内容：

- 添加一次性密码支持
- 针对资源访问控制工具 (RACF) 执行身份验证
- 向已启用 LDAP 的用户目录添加简单身份验证和安全层 (SASL) 绑定（而非简单的 LDAP 绑定）

如果实施自定义身份验证模块，则使用质询/响应机制的身份验证可能无法正常运行。自定义身份验证模块抛出的自定义消息不会传播到客户端。由于客户端（例如 Oracle Hyperion Enterprise Performance Management Workspace）会覆盖错误消息而显示常规消息，因此以下方案无效：

- 两个连续的 RSA SecurID PIN
- 采用质询方式的密码变体，例如，输入密码的第一个、最后一个和第三个字符

先决条件

- 名为 CustomAuth.jar 的经全面测试的 Java 存档，其中应包含自定义身份验证模块库。CustomAuth.jar 必须实施公共接口 CSSCustomAuthenticationIF，此接口在 com.hyperion.css 包中被定义为 Oracle Hyperion Shared Services API 标准的一部分。请参阅“http://download.oracle.com/docs/cd/E12825_01/epm.111/epm_security_api_11111/client/com/hyperion/css/CSSCustomAuthenticationIF.html”。
- 以 Shared Services 管理员身份访问 Shared Services

设计和编码注意事项

搜索顺序

除 Native Directory 外，还可以在 Oracle Hyperion Shared Services 中配置多个用户目录。系统会为配置的所有用户目录分配默认的搜索顺序位置。您可以从 Oracle Hyperion Shared Services Console 中修改搜索顺序。除 Native Directory 外，您可以将已配置的用户目录从搜索顺序中删除。Oracle Enterprise Performance Management System 不会使用不在搜索顺序中的用户目录。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》。

搜索顺序确定了 EPM System 循环搜索用户目录以对用户进行身份验证时遵照的顺序。如果用户在某用户目录中经过了身份验证，则 EPM System 将停止搜索并返回该用户。如果用户在搜索顺序中的所有用户目录中都无法通过身份验证，则 EPM System 将拒绝身份验证并返回错误。

自定义身份验证对搜索顺序的影响

自定义身份验证将影响 EPM System 安全性解释搜索顺序的方式。

如果自定义身份验证模块返回用户名，则 EPM System 仅在已启用自定义身份验证的用户目录中查找用户。在此阶段中，EPM System 将忽略未配置为启用自定义身份验证的用户目录。

了解自定义身份验证流

以下用例方案用于浏览自定义身份验证流：

- [用例方案 1](#)
- [用例方案 2](#)
- [用例方案 3](#)

用例方案 1

下表详细列出了此方案中使用的 EPM System 用户目录配置和搜索顺序。此方案假定自定义身份验证模块使用 RSA 基础结构对用户进行身份验证。

表 5-1 方案 1 的设置

用户目录类型和名称	搜索顺序	自定义身份验证	示例用户名称	密码 ¹
Native Directory	1	禁用	test_user_1 test_user_2 test_user_3	password
已启用 LDAP SunONE_West	2	禁用	test_ldap1 test_ldap_2 test_user_3 test_ldap_4	ldappassword
已启用 LDAP SunONE_East	3	已启用	test_ldap1 test_ldap_2 test_user_3	Ldappassword (对于 SunONE) 和 RSA PIN (对于自定义 模块)

¹ 为方便起见，假定所有用户均使用相同的用户目录密码。

要启动身份验证过程，用户可在 EPM System 产品登录屏幕中输入用户名和密码。在此方案中，自定义身份验证模块执行以下操作：

- 接受用户名和 RSA PIN 作为用户凭据
- 以 `username@providername` 格式返回用户名；例如，将 `test_ldap_2@SunONE_East` 返回给 EPM System 安全性

表 5-2 用户交互和结果

用户名和密码	身份验证结果	登录用户目录
test_user_1/password	成功	Native Directory
test_user_3/password	成功	Native Directory
test_user_3/ ldappassword	成功	SunONE_West (搜索顺序 2) ¹
test_user_3/RSA PIN	成功	SunONE_Eas (搜索顺序 3) ²
test_ldap_2/ ldappassword	成功	SunONE_West (搜索顺序 2)
test_ldap_4/RSA PIN	失败 EPM System 将显示身份验证错误。 ³	

¹ 由于用户已输入 EPM System 凭据，因此自定义身份验证无法对此用户进行身份验证。EPM System 仅可在未启用自定义身份验证的用户目录中标识此用户。用户不在 Native Directory (搜索顺序编号 1) 中，但确认在 SunONE West (搜索顺序编号 2) 中。

² EPM System 在 Native Directory (搜索顺序编号 1) 或 SunONE West (搜索顺序编号 2) 中未找到此用户。自定义身份验证模块根据 RSA Server 对用户进行验证，并将 `test_user_3@SunONE_EAST` 返回

- 给 EPM System。EPM System 在启用了自定义身份验证的用户目录 SunONE East（搜索顺序编号 3）中找到用户。
- Oracle 建议，所有使用自定义模块进行身份验证的用户均应包含在搜索顺序所含的已启用自定义身份验证的用户目录中。如果自定义身份验证模块返回的用户不在搜索顺序所含的已启用自定义身份验证的用户目录中，则登录会失败。

用例方案 2

下表详细列出了此方案中使用的 EPM System 用户目录配置和搜索顺序。此方案假定自定义身份验证模块使用 RSA 基础结构对用户进行身份验证。

在此方案中，自定义身份验证模块执行以下操作：

- 接受用户名和 RSA PIN 作为用户凭据
- 将用户名（例如 test_ldap_2）返回至 EPM System 安全性

表 5-3 示例搜索顺序

用户目录	搜索顺序	自定义身份验证	示例用户名称	密码 ¹
Native Directory	1	禁用	test_user_1 test_user_2 test_user_3	password
已启用 LDAP（例如 SunONE）	2	已启用	test_ldap1 test_ldap2 test_user_3	Ldappassword（对于 SunONE）和 RSA PIN（对于自定义模块）

¹ 为方便起见，假定所有用户均使用相同的用户目录密码。

要启动身份验证过程，用户可在 EPM System 产品登录屏幕中输入用户名和密码。

表 5-4 用户交互和结果

用户名和密码	登录结果	登录用户目录
test_user_1/password	成功	Native Directory
test_user_3/password	成功	Native Directory
test_user_3/ldappassword	失败	SunONE ¹
test_user_3/RSA PIN	成功	SunONE ²

¹ 由于密码不匹配，根据 Native Directory 对用户进行身份验证时失败。由于使用的密码不是有效的 RSA PIN，因此使用自定义身份验证模块对用户进行身份验证时失败。由于自定义身份验证设置将覆盖此目录中的 EPM System 身份验证，因此 EPM System 不会尝试对 SunONE（搜索顺序 2）中的该用户进行身份验证。

² 由于密码不匹配，根据 Native Directory 对用户进行身份验证时失败。自定义身份验证模块对用户进行身份验证，并将用户名 test_user_3 返回给 EPM System。

用例方案 3

下表详细列出了此方案中使用的 EPM System 用户目录配置和搜索顺序。此方案假定自定义身份验证模块使用 RSA 基础结构对用户进行身份验证。

为了清楚地展示此方案，Oracle 建议自定义身份验证模块以 username@providername 格式返回用户名；例如，test_ldap_4@SunONE。

表 5-5 示例搜索顺序

用户目录	搜索顺序	自定义身份验证	示例用户名称	密码 ¹
Native Directory	1	已启用	test_user_1 test_user_2 test_user_3	RSA_PIN
已启用 LDAP; 例 如 MSAD	2	禁用	test_ldap1 test_ldap4 test_user_3	ldappassword
已启用 LDAP (例 如 SunONE)	3	已启用	test_ldap1 test_ldap4 test_user_3	Ldappassword (对于 SunONE) 和 RSA PIN (对于自定义模 块)

¹ 为方便起见，假定所有用户均使用相同的用户目录密码。

要启动身份验证过程，用户可在 EPM System 产品登录屏幕中输入用户名和密码。

表 5-6 用户交互和结果

用户名和密码	身份验证结果	登录用户目录
test_user_1/password	成功	Native Directory
test_user_3/RSA_PIN	成功	Native Directory
test_user_3/ldappassword	成功	MSAD (搜索顺序 2)
test_ldap_4/ldappassword	成功	MSAD (搜索顺序 2)
test_ldap_4/RSA PIN	成功	SunONE (搜索顺序 3)

用户目录和自定义身份验证模块

要使用自定义身份验证模块，可以分别配置包含 EPM System 用户和组信息的用户目录，以将身份验证授权给自定义模块。

使用自定义模块进行身份验证的 EPM System 用户必须位于搜索顺序中包括的某个用户目录中（请参阅“[搜索顺序](#)”）。此外，必须配置用户目录以将身份验证授权给自定义模块。

自定义提供程序中的用户标识（例如，在 RSA SecurID 基础结构中为 1357642）可能不同于 Shared Services 中配置的用户目录中的用户名（例如，Oracle Internet Directory 中的 jDoe）。对用户进行身份验证之后，自定义身份验证模块必须将用户名 jDoe 返回给 EPM System。

注：

作为一项最佳做法，Oracle 建议 EPM System 中配置的用户目录中的用户名与自定义身份验证模块使用的用户目录中的用户名完全相同。

CSSCustomAuthenticationIF Java 接口

自定义身份验证模块必须使用 `CSSCustomAuthenticationIF` Java 接口与 EPM System 安全框架集成。如果自定义身份验证成功，则它必须返回用户名字符串。如果身份验证不成功，则返回一条错误消息。要完成身份验证过程，自定义身份验证模块返回的用户名必须位于 Shared Services 搜索顺序中包含的用户目录之一中。EPM System 安全框架支持 `username@providerName` 格式。

注:

由于 EPM System 安全框架在搜索用户时会将用户名解释为通配符，因此请确保自定义身份验证模块返回的用户名不包含 *（星号）。

有关 `CSSCustomAuthenticationIF` 接口签名，请参阅“[样本代码 1](#)”。

您的自定义身份验证模块（可以为类文件）必须包含在 `CustomAuth.jar` 中。包结构不重要。

有关 `CSSCustomAuthenticationIF` 接口的详细信息，请参阅[安全 API 文档](#)。

`CSSCustomAuthenticationIF` 的 `authenticate` 方法支持自定义身份验证。`authenticate` 方法将接受用户在尝试访问 EPM System 时输入的凭据（用户名和密码）作为输入参数。如果自定义身份验证成功，此方法将返回字符串（用户名）。如果身份验证不成功，则抛出 `java.lang.Exception`。此方法返回的用户名应唯一标识 Shared Services 搜索顺序中包含的某个用户目录中的用户。EPM System 安全框架支持 `username@providerName` 格式。

注:

要初始化资源（例如 JDBC 连接池），请使用类构造函数。执行此操作可改进性能，无需为每个身份验证加载资源。

部署自定义身份验证模块

一个 Oracle Enterprise Performance Management System 部署仅支持一个自定义模块。您可以按搜索顺序为一个或多个用户目录启用自定义身份验证。

自定义身份验证模块必须实施 `com.hyperion.css` 包中定义的公共接口

`CSSCustomAuthenticationIF`。该文档假定您具有完全功能的自定义模块，可根据所选的用户提供程序定义用户身份验证的逻辑。开发和测试自定义身份验证模块之后，您必须在 EPM System 环境中实施该模块。

步骤概述

您的自定义身份验证代码不应使用 `log4j` 记录错误。如果您在以前版本中使用的代码使用了 `log4j`，则必须先将其从代码中删除，才能在此版本中使用。

要实施自定义身份验证模块，请完成以下步骤：

- 停止 EPM System 产品，包括 Oracle Hyperion Shared Services 和使用 Shared Services API 的任何系统。

- 将自定义身份验证模块 Java 存档 CustomAuth.jar 复制到部署中：
 - **WebLogic:** 将 CustomAuth.jar 复制到 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` (通常为 `C:/Oracle/Middleware/user_projects/domains/EPMSysstem/lib`)。
- 如果您正在从实施了自定义身份验证模块的版本 11.1.2.0 或 11.1.2.1 升级, 则将 `EPM_ORACLE_HOME/common/jlib/11.1.2.0` 中的 CustomAuth.jar 移至 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib`。
- **所有客户端部署:** 将 CustomAuth.jar 复制到所有 EPM System 客户端部署中的以下位置:
 - `EPM_ORACLE_HOME/common/jlib/11.1.2.0` (通常为 `Oracle/Middleware/common/jlib/11.1.2.0`)。确保 CustomAuth.jar 文件始终放在 `EPM_ORACLE_HOME/common/jlib/11.1.2.0` 目录中。
- 对于使用自定义身份验证的所有服务器和客户端, CustomAuth.jar 文件必须位于以下两个位置:
- * `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib`
 - * `EPM_ORACLE_HOME/common/jlib/11.1.2.0`
- 更新 Shared Services 中的用户目录设置。请参阅[“更新 Shared Services 中的设置”](#)。
 - 启动 Shared Services, 然后启动其他 EPM System 产品。
 - 测试实施。请参阅[“测试部署”](#)。

更新 Shared Services 中的设置

默认情况下, 对所有用户目录禁用自定义身份验证。您可以覆盖默认行为, 为特定的外部用户目录或 Native Directory 启用自定义身份验证。

更新用户目录配置

对于必须启用自定义身份验证的用户目录, 必须更新其配置。

要更新用户目录配置:

1. 启动 Oracle Hyperion Foundation Services。
2. 以系统管理员身份访问 Oracle Hyperion Shared Services Console。
3. 选择管理, 然后选择配置用户目录。
4. 在“定义的用户目录”屏幕上, 选择要更改其自定义身份验证设置的用户目录。



注:
EPM System 只使用搜索顺序中包含的用户目录。

5. 单击编辑。
6. 选择显示高级选项。
7. 在自定义模块中, 选择身份验证模块, 以便为当前用户目录启用自定义模块。
8. 单击完成。

9. 重复此步骤，以更新搜索顺序中其他用户目录的配置。

更新安全选项

在开始以下过程之前，确保 `EPM_ORACLE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` 中存在 `CustomAuth.jar`。

要更新安全选项：

1. 以系统管理员身份访问 Shared Services Console。
2. 选择管理，然后选择配置用户目录。
3. 选择安全选项。
4. 选择显示高级选项。
5. 在身份验证模块中，输入自定义身份验证模块的完全限定类名称（在为其选择了自定义身份验证模块的所有用户目录上，应使用此模块来验证用户的身份）。例如 `com.mycompany.epm.CustomAuthenticationImpl`。
6. 单击确定。

测试部署

如果没有为自定义身份验证配置 Native Directory，请勿使用 Native Directory 用户测试自定义身份验证。

注：

您必须找出并更正与自定义身份验证模块相关的任何问题。Oracle 假定自定义模块已从自定义模块所用用户目录中的用户无缝映射至 EPM System 搜索顺序中已启用自定义身份验证的某个用户目录中的用户。

要测试部署，请使用用户目录中的用户凭据登录到 EPM System（例如，自定义模块使用的 RSA SecurID 基础结构）。这些凭据可能不同于 EPM System 凭据。

如果 EPM System 产品允许您访问其资源，则视为实施成功。如果系统显示错误指示找不到用户，这并非一定代表实施不成功。出现此情况时，请确认自定义用户存储中存在您输入的凭据，而且 EPM System 搜索顺序中的某个已启用自定义身份验证的用户目录中存在匹配的用户。

要测试自定义身份验证：

1. 确保 EPM System 产品正在运行。
2. 访问 EPM System 组件；例如，Oracle Hyperion Enterprise Performance Management Workspace。
3. 以已启用自定义身份验证的用户目录上定义的用户身份登录。
 - a. 在用户名中，输入您的用户凭据，例如 RSA 用户 ID。
 - b. 在密码中，输入密码；例如 RSA PIN。
 - c. 单击登录。
4. 确认您可以访问 EPM System 产品资源。

6

EPM System 安全准则

另请参阅：

- [实施 SSL](#)
- [更改管理员密码](#)
- [重新生成加密密钥](#)
- [更改数据库密码](#)
- [保护 Cookie 的安全](#)
- [缩短 SSO 令牌超时](#)
- [审核安全报表](#)
- [自定义身份验证系统以实现强身份验证](#)
- [禁用 EPM Workspace 调试实用程序](#)
- [更改默认 Web 服务器错误页面](#)
- [对第三方软件的支持](#)

实施 SSL

SSL 使用对数据加密的密码系统。SSL 在客户端和服务器之间创建安全的连接，可以安全地通过该连接发送数据。

为了保护 Oracle Enterprise Performance Management System 环境的安全，您需保护 Web 应用程序使用的所有通信通道以及使用 SSL 的用户目录连接的安全。请参阅“[为 EPM System 组件启用 SSL](#)”。

此外，还需使用防火墙保护所有代理端口，例如，Oracle Hyperion Reporting and Analysis 代理端口 6861。最终用户无需访问 EPM System 代理端口。

更改管理员密码

默认的 Native Directory 管理员用户帐户有权访问所有 Oracle Hyperion Shared Services 功能。此密码是在部署 Oracle Hyperion Foundation Services 时设置的。您必须定期更改此帐户的密码。

编辑 *admin* 用户帐户以更改密码。请参阅《*Oracle Enterprise Performance Management System 用户安全管理指南*》中的“修改用户帐户”。

重新生成加密密钥

使用 Oracle Hyperion Shared Services Console 定期重新生成以下信息：

- 单点登录令牌

 **注意：**

当您生成新的密钥库时，Oracle Hyperion Financial Management 和 Oracle Hyperion Profitability and Cost Management 使用的任务流将会失效。重新生成 keystore 后，请打开并保存任务流，以使它们重新生效。

- 可信服务密钥
- 提供程序配置密钥

请参阅“[重新生成加密密钥](#)”。

 **注：**

Oracle Hyperion Shared Services 和 Oracle Enterprise Performance Management System 的安全子系统采用具有 128 位密钥强度的 AES 加密。

更改数据库密码

定期更改所有 Oracle Enterprise Performance Management System 产品数据库的密码。本节中详细说明了在 Oracle Hyperion Shared Services Registry 中更改数据库密码的过程。

有关更改 EPM System 产品数据库密码的详细过程，请参阅《*Oracle Enterprise Performance Management System 安装与配置指南*》。

在 Shared Services Registry 中更改 EPM System 产品数据库密码：

1. 使用数据库管理控制台，更改用于配置 EPM System 产品数据库的用户帐户的密码。
2. 停止 EPM System 产品（Web 应用程序、服务和进程）。
3. 使用 EPM System Configurator，通过以下过程之一重新配置数据库。

仅限 **Oracle Hyperion Shared Services**：

 **注：**

在 EPM System 产品与 Shared Services 位于不同计算机的分布式环境中，必须在所有服务器上执行此过程。

- a. 在 EPM System Configurator 中的 Foundation 任务中，选择配置数据库。
- b. 在“Shared Services 和注册表数据库配置”页面上，选择连接到之前配置的 **Shared Services** 数据库。
- c. 指定在配置 Shared Services 数据库时使用的用户帐户的新密码。请勿更改任何其他设置。
- d. 继续执行配置，并在完成后单击完成。

Shared Services 之外的 EPM System 产品：

注：

只需针对部署在当前服务器上的 EPM System 产品执行这些步骤。

有关详细说明，请参阅《Oracle Enterprise Performance Management System 安装与配置指南》。

4. 启动 EPM System 产品和服务。

保护 Cookie 的安全

Oracle Enterprise Performance Management System Web 应用程序设置 cookie 来跟踪会话。在设置 cookie（特别是会话 cookie）时，服务器可以设置安全标志，该标志将强制浏览器通过安全的渠道发送 cookie。这种行为可降低会话攻击的风险。

注：

只有当 EPM System 产品部署在已启用 SSL 的环境中时，才需保护 cookie 的安全。

修改 Oracle WebLogic Server 会话描述符以保护 WebLogic Server cookie 的安全。将 session-param 元素中 cookieSecure 属性的值设置为 true。请参阅《Oracle Fusion Middleware Programming Security for Oracle WebLogic Server 11g》中的 "Securing Web Applications"。

缩短 SSO 令牌超时

默认的 SSO 令牌超时时间为 480 分钟。您应缩短 SSO 令牌超时（例如，缩短为 60 分钟），以便在令牌公开时尽可能减少令牌的重用。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“设置安全选项”。

审核安全报表

安全报表包含与配置其审核的安全任务相关的审核信息。定期从 Oracle Hyperion Shared Services Console 生成并审核此报表，特别地，这可确定 Oracle Enterprise Performance Management System 产品中失败的登录尝试次数和设置更改。选择详细信息视图作为报表生成选项，可根据已修改属性和新属性值对报表数据进行分组。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“生成报表”。

自定义身份验证系统以实现强身份验证

您可以使用自定义身份验证模块向 EPM System 中添加强身份验证。例如，您可以在非质询响应模式下使用 RSA SecurID 双重身份验证。该自定义身份验证模块对于瘦客户端和胖客户端都是透明的，不需要更改客户端部署。请参阅“使用自定义身份验证模块”。

禁用 EPM Workspace 调试实用程序

- 出于故障诊断目的，Oracle Hyperion Enterprise Performance Management Workspace 附带有未压缩的 JavaScript 文件。出于安全目的，您应将这些未压缩的 JavaScript 文件从生产环境中删除：
 - 创建 `EPM_ORACLE_HOME/common/epmstatic/wspace/js/` 目录的备份副本。
 - 从 `EPM_ORACLE_HOME/common/epmstatic/wspace/js` 下的每个子目录中，删除除 `DIRECTORY_NAME.js` 之外的其他 `.js` 文件。
每个子目录都包含一个采用该目录名称的 `.js` 文件。例如，`EPM_ORACLE_HOME/common/epmstatic/wspace/js/com/hyperion/bpm/web/common` 包含 `Common.js`。删除除了采用该目录名称的文件（在本例中为 `Common.js`）之外的所有 `.js` 文件。
- EPM Workspace 提供了许多调试实用程序和测试应用程序，在以调试模式部署 EPM Workspace 后可以访问它们。出于安全目的，管理员应在 EPM Workspace 中关闭客户端调试。
要关闭调试模式：
 1. 以管理员身份登录 EPM Workspace。
 2. 依次选择导航、管理和 **Workspace** 服务器设置。
 3. 在 Workspace 服务器设置的 **ClientDebugEnabled** 中，选择否。
 4. 单击确定。

更改默认 Web 服务器错误页面

当应用程序服务器不能用于接受请求时，用于后端应用程序服务器的 Web 服务器插件（例如，用于 Oracle WebLogic Server 的 Oracle HTTP Server 插件）返回显示插件生成信息的默认错误页面。Web 服务器还在其他情况下显示其默认错误页面。攻击者可以使用此信息发现公共网站的已知弱点。

自定义错误页面（Web 应用程序服务器插件和 Web 服务器），以便这些页面不包含有关任何生产系统组件的信息；例如，服务器版本、服务器类型、插件生成日期以及插件类型。有关更多信息，请查询应用程序服务器和 Web 服务器供应商文档。

对第三方软件的支持

Oracle 确认并支持第三方供应商做出的向后兼容性声明。因此，如果供应商声明了向后兼容性，则可以使用后续维护版本和服务程序包。如果确定了不兼容情况，Oracle 将指定应针对其部署产品的补丁版本（并从支持的版本矩阵中删除不兼容的版本），或者提供 Oracle 产品的维护版本或服务修复程序。

服务器端更新：对第三方服务器端组件更新的支持由后续维护版本策略控制。通常，Oracle 支持第三方服务器端组件升级到当前支持版本的下一个维护版本服务包，而不支持升级到下一个主版本。

客户端更新：Oracle 支持客户端组件的自动更新，包括第三方客户端组件的下一个主版本更新。例如，可以将浏览器 JRE 版本更新为当前支持的 JRE 版本。

A

自定义身份验证样本代码

样本代码 1

 注:

您的自定义身份验证代码不应使用 log4j 记录错误。如果您在以前版本中使用的自定义身份验证代码使用了 log4j，则必须先将其从代码中删除，才能在此版本中使用。

以下代码片段是自定义模块的空实施:

```
package com.hyperion.css.custom;

import java.util.Map;
import com.hyperion.css.CSSCustomAuthenticationIF;

public class CustomAuthenticationImpl implements CSSCustomAuthenticationIF {
    public String authenticate(Map context,String userName,
                               String password) throws Exception{
        try{
            //Custom code to find and authenticate the user goes here.
            //The code should do the following:
            //if authentication succeeds:
                //set authenticationSuccessFlag = true
                //return authenticatedUserName
            // if authentication fails:
                //log an authentication failure
                //throw authentication exception
        }
        catch (Exception e){
            //Custom code to handle authentication exception goes here
            //Create a new exception, set the root cause
            //Set any custom error message
            //Return the exception to the caller
        }
        return authenticatedUserName;
    }
}
```

输入参数:

- 上下文: 此映射包含区域设置信息的密钥-值对

- 用户名：此标识符可向使用自定义模块验证用户身份的用户目录唯一标识用户。用户在登录到 Oracle Enterprise Performance Management System 组件时需输入此参数的值。
- 密码：为使用自定义模块验证用户身份的用户目录中的用户设置的密码。用户在登录到 EPM System 组件时需输入此参数的值。

样本代码 2

以下样本代码演示了使用平面文件中包含的用户名和密码对用户进行自定义身份验证的过程。您必须在类构造函数中初始化用户和密码列表，才能使自定义身份验证生效。

```
package com.hyperion.css.security;

import java.util.Map;
import java.util.HashMap;
import com.hyperion.css.CSSCustomAuthenticationIF;
import java.io.*;

public class CSSCustomAuthenticationImpl implements
CSSCustomAuthenticationIF{
    static final String DATA_FILE = "datafile.txt";

    /**
     * authenticate method includes the core implementation of the
     * Custom Authentication Mechanism. If custom authentication is
     * enabled for the provider, authentication operations
     * are delegated to this method. Upon successful authentication,
     * this method returns a valid user name, using which EPM System
     * retrieves the user from a custom authentication enabled provider.
     * User name can be returned in the format username@providerName,
     * where providerName indicates the name of the underlying provider
     * where the user is available. authenticate method can use other
     * private methods to access various core components of the
     * custom authentication module.

     * @param context
     * @param userName
     * @param password
     * @return
     * @throws Exception
     */
}

Map users = null;

public CSSCustomAuthenticationImpl(){
    users = new HashMap();
    InputStream is = null;
    BufferedReader br = null;
    String line;
    String[] userDetails = null;
    String userKey = null;
    try{
        is =
```

```
CSSCustomAuthenticationImpl.class.getResourceAsStream(DATA_FILE);
    br = new BufferedReader(new InputStreamReader(is));
    while(null != (line = br.readLine())){
        userDetails = line.split(":");
        if(userDetails != null && userDetails.length==3){
            userKey = userDetails[0]+ ":" + userDetails[1];
            users.put(userKey, userDetails[2]);
        }
    }
}
catch(Exception e){
    // log a message
}
finally{
    try{
        if(br != null) br.close();
        if(is != null) is.close();
    }
    catch(IOException ioe){
        ioe.printStackTrace();
    }
}
}

/* Use this authenticate method snippet to return username from a flat file
*/

public String authenticate(Map context, String userName, String password)
throws Exception{
    //userName : user input for the userName
    //password : user input for password
    //context : Map, can be used to additional information required by
    //          the custom authentication module.

    String authenticatedUserKey = userName + ":" + password;

    if(users.get(authenticatedUserKey)!=null)
        return (String)users.get(authenticatedUserKey);
    else throw new Exception("Invalid User Credentials");
}

/* Refer to this authenticate method snippet to return username in
   username@providername format */

public String authenticate(Map context, String userName, String password)
throws Exception{

    //userName : user input for userName
    //password : user input for password
    //context : Map can be used to additional information required by
    //          the custom authentication module.

    //Your code should uniquely identify the user in a custom provider and in
    a configured
    //user directory in Shared Services. EPM Security expects you to append
```

```
the provider
    //name to the user name. Provider name must be identical to the name
of a custom
    //authentication-enabled user directory specified in Shared Services.

    //If invalid arguments, return null or throw exception with
appropriate message
    //set authenticationSuccessFlag = false

    String authenticatedUserKey = userName + ":" + password;
    if(users.get(authenticatedUserKey)!=null)
        String userNameStr = (new StringBuffer())
            .append((String)users.get(authenticatedUserKey))
            .append("@").append(PROVIDER_NAME).toString(
);
        return userNameStr;
    else throw new Exception("Invalid User Credentials");
    }
}
```

样本代码 2 的数据文件

确保数据文件已命名为 `datafile.txt`（样本代码中使用的名称），且包括在您创建的 Java 存档中。

使用以下代码作为平面文件的内容，该文件将用作自定义用户目录以支持样本代码 2 实施的自定义身份验证模块（请参阅“[样本代码 2](#)”。）

```
xyz:password:admin
test1:password:test1@LDAP1
test1:password:test1
test1@LDAP1:password:test1@LDAP1
test1@1:password:test1
user1:Password2:user1@SunONE1
user1_1:Password2:user1
user3:Password3:user3
DS_User1:Password123:DS_User1@MSAD1
DS_User1:Password123:DS_User1
DS_User1@1:Password123:DS_User1
```

如果您计划以 `username@providername` 格式返回用户名，请在用作自定义用户目录的平面文件中使用以下内容：

```
xyz:password:admin
test1:password:test1
test1@1:password:test1
user1_1:Password2:user1
user3:Password3:user3
DS1_1G100U_User61_1:Password123:DS1_1G100U_User61
```

```
DS1_1G100U_User61_1@1:Password123:DS1_1G100U_User61  
TUser:password:TUser
```

B

实施自定义登录类

Oracle Enterprise Performance Management System 提供

`com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl`，可从 x509 证书中提取用户标识 (DN)。

如果您必须从除 DN 以外的证书中的属性派生用户标识，则必须开发并实施一个与 `com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl` 类似的自定义登录类，如本附录中所述。

自定义登录类样本代码

此样本代码举例说明了默认

`com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl` 的实施。通常，您应自定义此实施的 `parseCertificate(String sCertificate)` 方法，以便从 DN 以外的证书属性派生用户名：

```
package com.hyperion.css.sso.agent;

import java.io.ByteArrayInputStream;
import java.io.UnsupportedEncodingException;
import java.security.Principal;
import java.security.cert.CertificateException;
import java.security.cert.CertificateFactory;
import java.security.cert.X509Certificate;
import com.hyperion.css.CSSSecurityAgentIF;
import com.hyperion.css.common.configuration.*;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * X509CertificateAuthImpl implements the CSSSecurityAgentIF interface It
 * accepts
 * the X509 certificate of the authenticated user from the Web Server via a
 * header, parses the certificate, extracts the DN of the User and
 * authenticates the user.
 */
public class X509CertificateSecurityAgentImpl implements CSSSecurityAgentIF
{
    static final String IDENTITY_ATTR = "CN";
    String g_userDN = null;
    String g_userName = null;
    String hostAddress= null;
    /**
     * Returns the User name (login name) of the authenticated user,
     * for example demouser. See CSS API documentation for more information
     */
}
```

```
    public String getUsername(HttpServletRequest req,
        HttpServletResponse res)
        throws Exception
    {
        hostAddress = req.getServerName();
        String certStr = getCertificate(req);

        String sCert = prepareCertificate(certStr);

        /* Authenticate with a CN */
        parseCertificate(sCert);

        /* Authenticate if the Login Attribute is a DN */
        if (g_username == null)
        {
            throw new Exception("User name not found");
        }
        return g_username;
    }

    /**
     * Passing null since this is a trusted Security agent
     authentication
     * See Security API documentation for more information on
     CSSSecurityAgentIF
     */
    public String getPassword(HttpServletRequest req,
        HttpServletResponse res)
        throws Exception
    {
        return null;
    }

    /**
     * Get the Certificate sent by the Web Server in the HYPLOGIN
     header.
     * If you pass a different header name from the Web server, change
     the
     * name in the method.
     */
    private String getCertificate(HttpServletRequest request)
    {
        String cStr = (String)request
            .getHeader(CSSConfigurationDefaults.HTTP_HEADER_HYPLOGI
N);
        return cStr;
    }

    /**
     * The certificate sent by the Web server is a String.
     * Put a "\n" in place of whitespace so that the X509Certificate
     * java API can parse the certificate.
     */
    private String prepareCertificate(String gString)
    {

```

```
String str1 = null;
String str2 = null;

str1 = gString.replace("-----BEGIN CERTIFICATE-----", "");
str2 = str1.replace("-----END CERTIFICATE-----", "");
String certStrWithNL = "-----BEGIN CERTIFICATE-----"
    + str2.replace(" ", "\n") + "-----END CERTIFICATE-----";
return certStrWithNL;
}

/**
 * Parse the certificate
 * 1. Create X509Certificate using the certificateFactory
 * 2. Get the Principal object from the certificate
 * 3. Set the g_userDN to a certificate attribute value (DN in this
sample)
 * 4. Parse the attribute (DN in this sample) to get a unique username
 */
private void parseCertificate(String sCertificate) throws Exception
{
    X509Certificate cert = null;
    String userID = null;
    try
    {
        X509Certificate clientCert = (X509Certificate)CertificateFactory
            .getInstance("X.509")
            .generateCertificate(
                new
ByteArrayInputStream(sCertificate
                                                                .getBytes("UTF-8")));

        if (clientCert != null)
        {
            Principal princDN = clientCert.getSubjectDN();
            String dnStr = princDN.getName();
            g_userDN = dnStr;
            int idx = dnStr.indexOf(",");
            userID = dnStr.substring(3, idx);
            g_userName = userID;
        }
    }
    catch (CertificateException ce)
    {
        throw ce;
    }
    catch (UnsupportedEncodingException uee)
    {
        throw uee;
    }
} //end of getUserFromCert
} // end of class
```

部署自定义登录类

要实施自定义登录类，请完成以下步骤：

1. 创建并测试自定义登录类。确保代码中没有对 log4j 的任何引用。请参阅“[自定义登录类样本代码](#)”。

您可以对自定义类使用任何名称。

2. 将自定义登录类打包到 CustomAuth.jar

3. 将 CustomAuth.jar 复制到部署中：

- **WebLogic:** 将 CustomAuth.jar 复制到 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` (通常为 `Oracle/Middleware/user_projects/domains/EPMSysstem/lib`)。

注：

如果您正在从已实施自定义登录类的版本 11.1.2.0 或 11.1.2.1 升级，则将 `EPM_ORACLE_HOME/common/jlib/11.1.2.0` 中的 CustomAuth.jar 移至 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib`。

- **客户端部署:** 将 CustomAuth.jar 复制到所有 Oracle Enterprise Performance Management System 客户端部署中的以下位置：

`EPM_ORACLE_HOME/common/jlib/11.1.2.0` (通常为 `Oracle/Middleware/common/jlib/11.1.2.0`)

如果您使用了自定义登录类，Oracle 建议您启用“客户端证书身份验证”。

C

在用户目录之间迁移用户和组

概览

在许多方案中，已设置的 Oracle Enterprise Performance Management System 用户的用户和组标识可能会过时。如果可用的设置信息变得过时，EPM System 组件将不可访问。可能产生过时设置数据的方案包括：

- 废弃用户目录：组织可能在将用户移到另一个用户目录后废弃了原用户目录。
- 版本升级：用户目录版本升级可能涉及主机名更改或操作系统环境所需的更改。
- 供应商更改：组织可能停用了用户目录，并选用了来自另一个供应商的用户目录。例如，组织可能将 Oracle Internet Directory 替换为了 SunONE Directory Server。

注：

- 在本附录中，您正在逐步淘汰的用户目录称为源用户目录，您将用户帐户移动到用户目录称为目标用户目录。
- 此迁移过程不支持将用户帐户从源用户目录迁移到目标用户目录，只能迁移他们在 EPM 应用程序中的关联。必须手动在目标用户目录中创建用户。此过程适用于任何源用户目录（包括 Native Directory）的用户。

如果为 Hyperion Shared Services 配置的源用户目录具有 Native Directory 以外的组，则还应该在目标用户目录中创建那些组。

先决条件

- 要在用户目录之间迁移其设置数据的 Oracle Enterprise Performance Management System 用户和组必须存在于目标用户目录中。
源用户目录中存在的组关系必须在目标用户目录中保留。
- EPM System 用户的用户名在源和目标用户目录中必须相同。

迁移过程

导出 Native Directory 数据

在源环境中按照以下步骤操作：

使用 Oracle Hyperion Enterprise Performance Management System 生命周期管理从 Native Directory 仅导出以下 Shared Services 对象：

- Native Directory 组

- 分配的角色
- 授权列表

通常，生命周期管理在 `EPM_ORACLE_INSTANCE/import_export/USER_NAME/EXPORT_DIR/resource/Native Directory` 中创建多个导出文件，其中 `USER_NAME` 是用户的标识（例如，执行导出操作的 `admin`），`EXPORT_DIR` 是导出目录的名称。通常，系统将创建以下文件：

- `Groups.csv`
- `Assigned Roles.csv`
- `Delegated Lists.csv`
- `Assigned Roles/PROD_NAME.csv`（针对部署的每个应用程序），其中，`PROD_NAME` 是 Oracle Enterprise Performance Management System 组件的名称；例如，`Shared Services`。

注：

- 有关使用生命周期管理导出数据详细说明，请参阅《Oracle Enterprise Performance Management System 生命周期管理指南》。
- 确保未导出 `Users.csv` 文件。

导出对象后，验证迁移状态报表显示上次导出操作的状态是否为 `Completed`。

要导出 `Native Directory` 数据：

1. 在 Oracle Hyperion Shared Services Console 的“视图”窗格的 **Foundation** 应用程序组中，选择 **Shared Services** 应用程序。
2. 要进行迁移，请仅从以下列表中选择所需对象：
 - `Native Directory` 组
 - 分配的角色
 - 授权列表
3. 单击导出。
4. 输入导出存档的名称。默认为 `admin DATE`；例如，`admin 13-03-18`。
5. 单击导出。

导入 `Native Directory` 数据

在目标环境中按照以下步骤操作：

1. 手动创建：
 - a. 在目标外部用户目录（类似于源用户目录）中创建用户。
 - b. 在目标外部用户目录（类似于源用户目录）中创建组，`Native Directory` 组除外。
2. 配置目标用户目录。
如果将源用户目录中的用户帐户移动到了其他用户目录，请将目标用户目录添加为 EPM System 中的外部用户目录。例如，如果将用户帐户从 Oracle Internet

Directory 移动到 SunONE Directory Server，则将 SunONE Directory Server 添加为外部用户目录。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“第 3 章，配置用户目录”。

 注：

确保目标用户目录包含其数据正从源用户目录迁移的所有 EPM System 用户的用户帐户和组。

如果已将用户移动到已定义为外部用户目录的用户目录，请验证该用户帐户对 Oracle Hyperion Shared Services 是否可见。您可以通过从 Shared Services Console 搜索此用户来验证。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“搜索用户、组、角色和授权列表”。

将目标用户目录配置为外部用户目录时，请验证“登录属性”所指向的属性的值是否原本用作源用户目录中的用户名。请参阅“先决条件”。

3. 将目标用户目录移至搜索顺序顶部。

 注：

如果目标用户目录名称与源目录名称相同，则必须从 EPM System 配置中删除源用户目录。

Shared Services 将为新添加的用户目录分配一个比现有目录的搜索顺序更低的搜索顺序优先级。更改搜索顺序，使目标用户目录具有比源用户目录更高的搜索顺序优先级。使用此指令，Shared Services 可在搜索源之前先在目标用户目录中发现用户。请参阅《Oracle Enterprise Performance Management System 用户安全管理指南》中的“管理用户目录搜索顺序”。

4. 重新启动 Oracle Hyperion Foundation Services 和其他 EPM System 组件，以强制实施您所做的更改。
5. 导入 Native Directory 数据（从源环境导出）：
使用 create/update 选项运行生命周期管理，以导入先前从 Native Directory 中导出的数据（如下面所列）。

- Groups.csv
- Assigned Roles.csv
- Delegated Lists.csv

 注：

- 有关使用生命周期管理导入数据的详细说明，请参阅《Oracle Enterprise Performance Management System 生命周期管理指南》。
- 确保未导入 Users.csv 文件。

导入数据后，验证迁移状态报表显示上次导入操作的状态是否为 Completed。

要导入 Native Directory 数据：

- a. 在 Shared Services Console 的“视图”窗格中，展开文件系统。
- b. 选择导入文件的文件系统位置。
- c. 选择要导入设置信息的对象的类型。
- d. 单击导入。
- e. 单击确定。

特定于产品的更新

▲ 注意：

Oracle 建议在启动特定于产品的更新之前，先备份 Oracle Enterprise Performance Management System 组件使用的存储库中的用户和组数据。更新本地产品存储库中的信息之后，只能从备份中恢复本地产品存储库中的旧用户和组数据。

Planning

Oracle Hyperion Planning 将有关已设置用户和组的信息存储在 Planning 存储库中。如果因在用户目录间迁移用户和组而导致 Native Directory 中的用户标识发生变更，则必须通过选择“迁移用户/组”将 Planning 存储库中的信息与 Native Directory 中的信息同步。在 Planning 中分配数据表单、成员和任务列表的访问权限时会显示此按钮。

Financial Management

Oracle Hyperion Financial Management 将已设置用来访问对象的用户和组的相关信息记录在本地 Financial Management 存储库中。如果因在用户目录间迁移用户和组而导致 Native Directory 中的用户和组信息发生变更，则必须将 Financial Management 存储库中的信息与 Native Directory 中的信息同步。