

Oracle® Enterprise Performance Management System 安全性組態手冊



版本 11.2
F28796-22
2023 年 12 月

ORACLE®

版權所有 © 2005, 2023, Oracle 和 (或) 其關係公司。

主要作者：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>.

目錄

文件協助工具

說明文件意見

1 關於 EPM System 安全性

關於 EPM System	1-1
假設您已熟悉的知識	1-1
安全性基礎建設元件	1-2
使用者驗證	1-2
提供 (角色式授權)	1-5
啟動 Shared Services Console	1-8

2 已啟用 SSL 的 EPM System 元件

假設	2-1
資訊來源	2-1
位置參照	2-2
關於已啟用 SSL 的 EPM System 產品	2-2
受支援的 SSL 案例	2-3
必要的憑證	2-3
在 SSL 卸載器終止 SSL	2-4
EPM System 的完整 SSL 部署	2-7
部署架構	2-7
假設	2-8
針對完整 SSL 設定 EPM System	2-9
重新設定 EPM System 的公用設定	2-10
選用：安裝 WebLogic Server 的根 CA 憑證	2-11
在 WebLogic Server 上安裝憑證	2-12
設定 WebLogic Server	2-13
啟用與已啟用 SSL 之 Oracle 資料庫的 HFM 伺服器連線	2-15

Oracle HTTP Server 程序	2-20
設定部署在 WebLogic Server 上的 EPM System Web 元件	2-23
更新網域組態	2-25
重新啟動伺服器和 EPM System	2-26
測試部署	2-26
設定已啟用 SSL 的外部使用者目錄	2-27
在 Web 伺服器終止 SSL	2-28
適用於 Essbase 11.1.2.4 的 SSL	2-30
安裝及部署 Essbase 元件	2-32
針對 Essbase 使用受信任第三方 CA 憑證	2-32
建立每一階段作業 SSL 連線	2-39
適用於 Essbase 21c 的 SSL	2-39
安裝及部署 Essbase 元件	2-42
針對 Essbase 使用受信任第三方 CA 憑證	2-42
建立每一階段作業 SSL 連線	2-48

3 啟用安全性代理程式的 SSO

受支援的 SSO 方法	3-1
來自 Oracle Access Manager 的單一登入	3-3
OracleAS Single Sign-on	3-5
測試部署	3-6
啟用 EPM System 的 OSSO	3-7
保護 EPM System 產品進行 SSO	3-10
搭配識別管理產品的標頭型 SSO	3-15
針對搭配 Oracle Identity Cloud Services 的標頭型 SSO 設定 EPM System	3-16
先決條件與範例 URL	3-16
啟用 EPM System 的標頭型驗證	3-17
將 EPM System 應用程式與閘道新增至 Oracle Identity Cloud Services	3-17
設定 App Gateway	3-22
設定用於授權的使用者目錄	3-22
在 EPM System 中啟用 SSO	3-22
更新 EPM Workspace 設定	3-22
SiteMinder SSO	3-23
Kerberos 單一登入	3-26
設定 EPM System 進行 SSO	3-39
Smart View 的單一登入選項	3-40

4 設定使用者目錄

使用者目錄和 EPM System 安全性	4-1
使用者目錄組態的相關作業	4-2
Oracle Identity Manager 與 EPM System	4-2
Active Directory 資訊	4-3
設定 OID、Active Directory 及其他 LDAP 型的使用者目錄	4-3
將關聯式資料庫設定為使用者目錄	4-15
測試使用者目錄連線	4-17
編輯使用者目錄設定	4-17
刪除使用者目錄組態	4-18
管理使用者目錄搜尋順序	4-19
設定安全性選項	4-20
重新產生加密金鑰	4-23
使用特殊字元	4-24

5 使用自訂驗證模組

簡介	5-1
使用案例的範例和限制	5-3
先決條件	5-3
設計和編碼的考量	5-3
部署自訂驗證模組	5-8

6 EPM System 的安全準則

實作 SSL	6-1
變更管理密碼	6-1
重新產生加密金鑰	6-1
變更資料庫密碼	6-2
保護 Cookie 的安全	6-3
縮短 SSO 憑證逾時時間	6-3
複查安全性報表	6-3
自訂嚴密驗證的驗證系統	6-4
停用 EPM Workspace 除錯公用程式	6-4
變更預設的 Web 伺服器錯誤頁面	6-4
對於第三方軟體的支援	6-4

A 自訂驗證的程式碼範例

程式碼範例 1	A-1
程式碼範例 2	A-2
程式碼範例 2 的資料檔案	A-4

B 實作自訂登入類別

自訂登入類別的程式碼範例	B-1
部署自訂登入類別	B-4

C 在不同的使用者目錄之間移轉使用者和群組

簡介	C-1
先決條件	C-1
移轉程序	C-1
產品專有的更新	C-4

文件協助工具

如需有關 Oracle 對於協助工具的承諾資訊，請瀏覽 Oracle Accessibility Program 網站，網址為 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>。

取得 Oracle 支援

已購買支援服務的 Oracle 客戶，可從 My Oracle Support 取得網路支援。如需資訊，請瀏覽 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>；如您有聽力障礙，請瀏覽 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>。

說明文件意見

若您對此說明文件有任何意見，請按一下任何「Oracle 說明中心」主題中頁面底部的「意見」按鈕。您也可以將電子郵件傳送至 epmdoc_ww@oracle.com。

1

關於 EPM System 安全性

另請參閱：

- [關於 EPM System](#)
- [假設您已熟悉的知識](#)
- [安全性基礎建設元件](#)
- [使用者驗證](#)
- [提供 \(角色式授權\)](#)
- [啟動 Shared Services Console](#)

關於 EPM System

Oracle Enterprise Performance Management System 產品是一套可在企業內部廣泛使用的全方位系統，它整合各種財務管理和規劃應用程式模組套件與最全面的商業智慧功能，為企業提供報表及分析結果。EPM System 產品的主要元件如下：

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

如需上述每個產品系列中產品和元件的相關資訊，請參閱 *Oracle Enterprise Performance Management System 安裝入門*。

假設您已熟悉的知識

本指南的適用對象，為負責設定、保護及管理 Oracle Enterprise Performance Management System 元件的系統管理員。指南中包含下列知識：

- 非常瞭解您組織的安全性基礎建設，包括：
 - 目錄伺服器，例如 Oracle Internet Directory、Sun Java System Directory Server 及 Microsoft Active Directory
 - 安全通訊端層 (SSL) 的使用，以便保護通訊管道
 - 存取權管理系統，例如 Oracle Access Manager 及 SiteMinder
 - 單一登入 (SSO) 基礎建設，例如 Kerberos
- 熟知與您企業相關的 EPM System 安全性概念

安全性基礎建設元件

Oracle Enterprise Performance Management System 整合多種安全性元件，以確保提供您強大的應用程式安全防護。EPM System 在整合到安全性基礎建設中之後，就會提供一套高安全性的應用程式，確保您的資料與存取都安全無虞。您能用來保護 EPM System 的基礎建設元件包括：

- 選用的存取管理系統，例如 Oracle Access Manager，以提供 EPM System 元件的 SSO 存取
- 整合式 SSO 基礎建設 (例如 Kerberos) 的使用。

您可以搭配存取管理系統 (SiteMinder) 來使用 Kerberos 驗證，以確保 Windows 使用者可以毫無障礙地登入 SiteMinder 和 EPM System 元件。

- 安全通訊端層 (SSL) 的使用，以保護 EPM System 元件與用戶端之間的通訊管道

使用者驗證

使用者驗證可啟用跨 Oracle Enterprise Performance Management System 元件的單一登入 (SSO) 功能，方法是驗證每位使用者的登入資訊來判別通過驗證的使用者。使用者驗證，加上元件專有的驗證功能，會授予使用者 EPM System 元件的存取權。而驗證的授予程序稱為提供。

驗證元件

下列各節說明支援 SSO 的元件：

- [原生目錄](#)
- [外部使用者目錄](#)

原生目錄

原生目錄會參照至 Oracle Hyperion Shared Services 用來支援提供，以及儲存植入資料 (例如預設使用者帳戶) 的關連性資料庫。

原生目錄功能：

- 維護及管理預設的 EPM System 使用者帳戶
- 儲存所有 EPM System 提供資訊 (使用者、群組及角色之間的關係)

您可以使用 Oracle Hyperion Shared Services Console 來存取及管理 原生目錄。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「管理原生目錄」。

外部使用者目錄

使用者目錄會參照至與 EPM System 元件相容的公司使用者和識別管理系統。

支援 EPM System 元件的使用者目錄有許多種，包括 LDAP 型使用者目錄，例如 Oracle Internet Directory、Sun Java System Directory Server (前身為 SunONE Directory Server)，以及 Microsoft Active Directory。也支援使用 Rational 資料庫作為使用者目錄。我們在本文件中，會把所有非原生目錄的使用者目錄稱為外部使用者目錄。

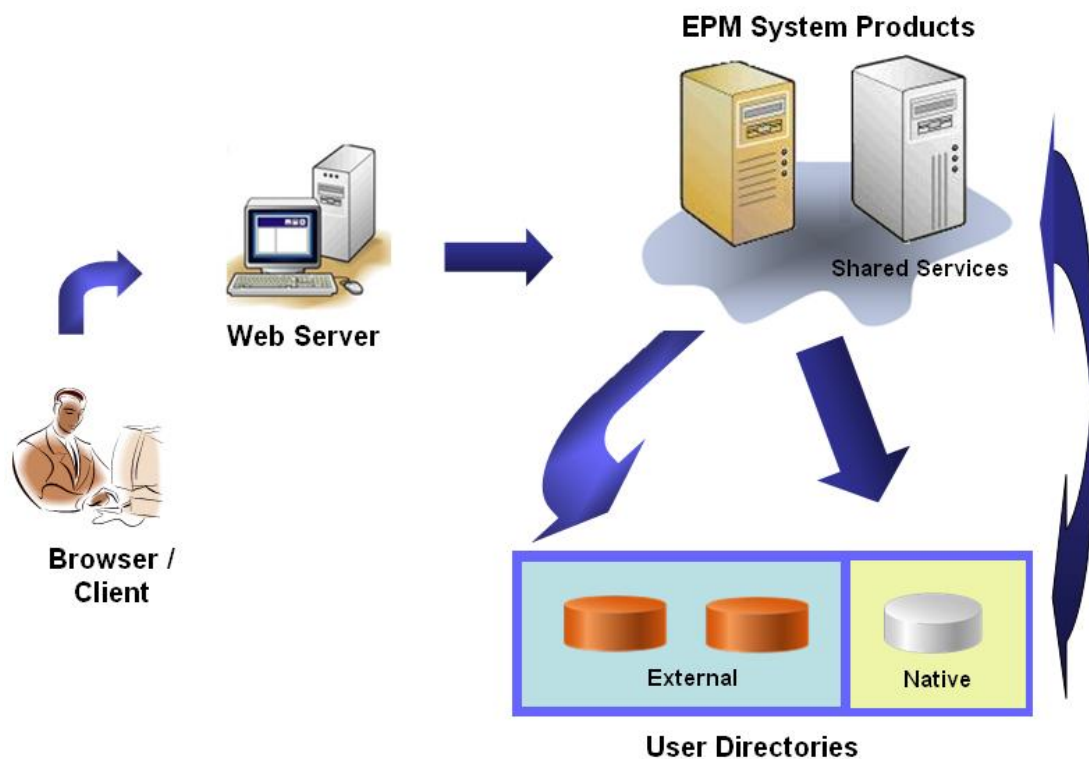
如需受支援的使用者目錄清單，請參閱張貼在 Oracle Technology Network (OTN) 的 [支援 Oracle Fusion Middleware 的系統組態](#) 頁面上的 *Oracle Enterprise Performance Management System Certification Matrix (僅英文版)*。

您可以在 Shared Services Console 中，將許多外部使用者目錄設定成 EPM System 使用者和群組的來源。每個 EPM System 使用者在單一已設定使用者目錄中都必須要有獨特的帳戶。一般來說，EPM System 使用者會受指派成群組，以加快提供的速度。

預設的 EPM System 單一登入

EPM System 支援跨 EPM System Web 應用程式的 SSO 機制，方法是允許某個應用程式的受驗證使用者，在沒有重新輸入認證的情況下輕鬆地導覽至其他應用程式。SSO 的實作方式，就是將某個處理使用者驗證的常見安全性環境，與跨 EPM System 元件的提供功能 (角色型驗證) 整合。

以下圖解說明預設的 SSO 程序。



1. 使用者可透過瀏覽器存取某個 EPM System 元件登入畫面，然後輸入使用者名稱和密碼。
EPM System 元件會向已設定的使用者目錄 (包括原生目錄) 查詢，以便驗證使用者的認證。當系統在使用者目錄中找到相符的使用者帳戶時，就會終止搜尋作業，並將使用者的資訊傳回 EPM System 元件。
如果系統在所有使用者目錄中都找不到該使用者帳戶，就會拒絕存取。
2. EPM System 會利用已擷取的使用者資訊，向原生目錄查詢以取得該使用者的提供詳細資料。
3. EPM System 元件會查看元件中的「存取控制清單」(ACL)，以判斷該使用者能夠存取的應用程式人工因素。

當 EPM System 元件收到原生目錄的提供資訊時，就會讓元件可供該使用者使用。此時，提供給使用者的所有 EPM System 元件都會啟用 SSO 機制。

來自存取管理系統的單一登入

若要進一步保護 EPM System 元件，您可以採用受支援的存取管理系統 (例如 Oracle Access Manager 或 SiteMinder)，它可將通過驗證的使用者認證提供給 EPM System 元件，以及根據預先定義的存取權限來控制存取。

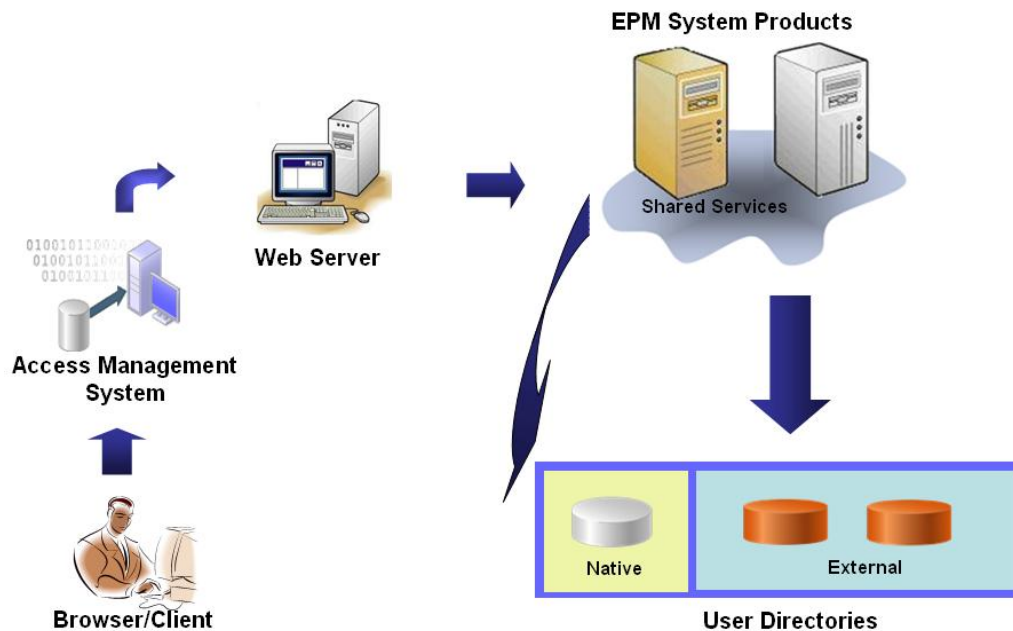
只有 EPM System Web 應用程式才能使用安全性代理程式的 SSO。在此案例中，EPM System 元件會使用安全性代理程式所提供的使用者資訊，來判斷使用者的存取權限。為提高安全性，Oracle 建議您應利用防火牆封鎖對伺服器的直接存取，讓所有要求都必須透過 SSO 入口網站來傳送。

我們支援存取管理系統的 SSO，方法是透過可接受的 SSO 機制來接受通過驗證的使用者認證。請參閱[受支援的 SSO 方法](#)。存取管理系統會對使用者進行驗證，並將登入名稱傳遞給 EPM System。EPM System 會根據已設定使用者目錄來驗證該登入名稱。

請參閱下列主題。

- [來自 Oracle Access Manager 的單一登入](#)
- [OracleAS Single Sign-on](#)
- [SiteMinder SSO](#)
- [Kerberos 單一登入](#)

以下是概念的圖解說明：



1. 使用者利用瀏覽器，要求存取某個受存取管理系統 (例如 Oracle Access Manager 或 SiteMinder) 保護的資源。

 備註：

EPM System 被定義為受存取管理系統保護的資源。

存取管理系統會攔截要求，並顯示登入畫面。使用者會輸入使用者名稱和密碼，然後系統會根據存取管理系統中的已設定使用者目錄，確認該使用者的真實性。EPM System 元件也可設定成能與這些使用者目錄一起運作。

系統會將通過驗證之使用者的相關資訊傳遞至 EPM System 元件，然後元件會把該資訊視為有效資訊來接受。

存取管理系統會利用可接受的 SSO 機制，將使用者的登入名稱 (Login Attribute 的值) 傳遞給 EPM System 元件。請參閱[受支援的 SSO 方法](#)。

2. EPM System 元件為了要驗證使用者認證，會嘗試在使用者目錄中尋找該使用者。如果系統找到相符的使用者帳戶，就會將使用者資訊傳回至 EPM System 元件。EPM System 安全性會設定 SSO 憑證，它可用來能啟用各個 EPM System 元件的 SSO 機制。
3. EPM System 會利用已擷取的使用者資訊，向原生目錄查詢以取得該使用者的提供詳細資料。

當 EPM System 元件收到使用者提供資訊時，就會讓元件可供該使用者使用。提供給使用者的所有 EPM System 元件都會啟用 SSO 機制。

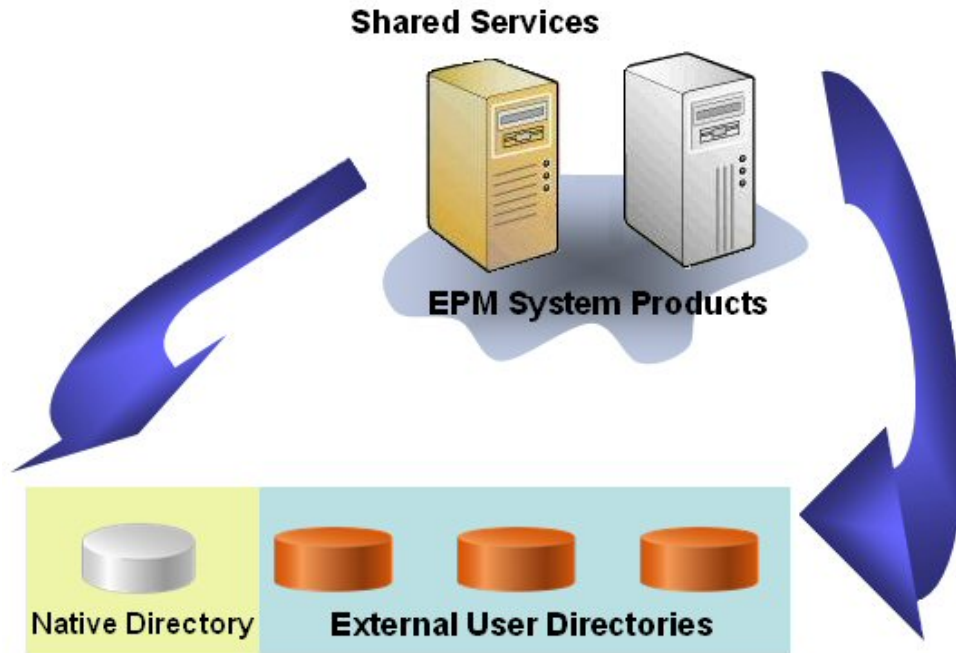
提供 (角色式授權)

Oracle Enterprise Performance Management System 安全性利用角色的概念，來決定使用者的應用程式存取權限。角色是決定使用者對於應用程式功能存取的權限，某些 EPM System 元件會強制執行物件層級的 ACL，以便進一步限制使用者對其人工因素 (例如報表和成員) 的存取。

每項 EPM System 元件皆會針對各種不同的業務需求，量身訂做數種預設角色。屬於某個 EPM System 元件的每個應用程式，都會繼承這些角色。在 Oracle Hyperion Shared Services 中登錄之應用程式的預先定義角色，都會出現在 Oracle Hyperion Shared Services Console 中。您也可以根據特定需求建立其他角色來彙總預設角色。這些角色會用於提供。將特定角色授予 EPM System 應用程式及其資源所屬使用者與群組的程序，即稱為提供。

原生目錄和已設定使用者目錄是提供程序的使用者和群組資訊來源。您可以在 Shared Services Console 中，巡覽並提供所有經過設定之使用者目錄下的使用者與群組。您也可以提供程序中，使用建立在原生目錄中的應用程式專有彙總角色。

下列圖例是授權程序的概觀：



1. 當使用者通過驗證之後，EPM System 元件會向使用者目錄查詢，以判斷該使用者的群組。
2. EPM System 元件會使用群組和使用者資訊，從 Shared Services 擷取使用者的提供資料。元件會利用此資料，判斷使用者能存取哪些資源。
系統會分別完成每個產品的產品特有提供工作，例如設定產品專有的存取控制。此資料會與提供資料合併，用於判別使用者的產品存取權。

EPM System 產品即是利用這些概念依角色進行提供。

角色

角色是一種構造 (類似於存取控制清單)，可定義授予使用者與群組的存取權限，讓其對 EPM System 資源執行功能。角色是資源或資源類型 (使用者能存取的內容，例如報表) 與使用者可對資源執行之動作 (例如檢視及編輯) 的組合。

對 EPM System 應用程式資源的存取是受到限制的。而只有該使用者或該使用者所屬的群組獲得提供該資源存取權的角色之後，該使用者才能夠存取這些資源。依角色限制存取可讓管理員控制及管理應用程式存取權。

全域角色

全域角色 (能跨多個產品使用的 Shared Services 角色) 可讓使用者能夠跨 EPM System 產品執行特定工作。例如，Shared Services 管理員可以為所有 EPM System 應用程式提供使用者。

預先定義的角色

預先定義的角色是 EPM System 產品的內建角色。您無法刪除這些角色。屬於某個 EPM System 產品的每個應用程式例項都會繼承該產品所有的預先定義角色。當您建立應用程式時，會針對每個應用程式透過 Shared Services 來登錄這些角色。

彙總角色

彙總角色 (又稱為自訂角色) 可彙總應用程式中的多個產品角色。彙總角色可包含其他彙總角色。例如，**Shared Services** 管理員或佈建管理員可以建立一個彙總角色，結合了某個 **Oracle Hyperion Planning** 應用程式的「計劃人員」與「檢視使用者」角色。彙總角色可簡化涵括數個精細角色之產品的管理。全域 **Shared Services** 角色也可包含在彙總角色中。您無法建立跨應用程式或產品的彙總角色。

使用者

使用者目錄可儲存能存取 **EPM System** 產品之使用者的相關資訊。驗證及授權程序皆會使用使用者資訊。您只能從 **Shared Services Console** 建立及管理原生目錄使用者。

Shared Services Console 會顯示所有已設定之使用者目錄中的使用者。這些使用者可個別提供，以授予已在 **Shared Services** 上登錄之 **EPM System** 應用程式的存取權。**Oracle** 不建議提供個別使用者。

預設的 EPM System 管理員

管理員帳戶 (預設名稱為 `admin`) 是在部署期間於原生目錄中建立的。這是權限最高的 **EPM System** 帳戶，只應該用來設定系統管理員，也就是受指派來管理 **EPM System** 安全性和環境的資訊技術專家。

EPM System 管理員的使用者名稱和密碼，是在 **Oracle Hyperion Foundation Services** 部署期間設定的。由於這個帳戶不受公司密碼原則的限制，**Oracle** 建議您，請在建立系統管理員帳戶之後停用這個帳戶。

預設的 **EPM System** 管理員帳戶通常會用來執行下列工作：

- 將公司目錄設定成外部使用者目錄。請參閱[設定使用者目錄](#)。
- 建立系統管理員，方法是將 **Shared Services** 管理員角色提供給某位公司資訊技術專家。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「提供使用者與群組」。

系統管理員

系統管理員通常是公司的資訊技術專家，負責讀取、寫入及執行與 **EPM System** 部署作業相關之所有伺服器的存取權限。

系統管理員通常會執行下列工作：

- 停用預設的 **EPM System** 管理員帳戶。
- 至少建立一個功能管理員。
- 使用 **Shared Services Console** 建立 **EPM System** 的安全性組態。
- (選用) 將使用者目錄設定成外部使用者目錄。
- 定期執行 **Log Analysis** 工具來監控 **EPM System**。

我們已經在本指南中說明功能管理員要執行的工作。

建立功能管理員的程序。

- 將公司目錄設定成外部使用者目錄。請參閱 [設定使用者目錄](#)。
- 將建立功能管理員所需的角色提供給某個使用者或群組。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「提供使用者與群組」。

您必須將下列角色提供給功能管理員：

- Shared Services 的 LCM 管理員
- 每個已部署 EPM System 元件的管理員和佈建管理員角色

功能管理員

功能管理員是身為 EPM System 專家的公司使用者。此使用者通常會在 Shared Services 中所設定之公司目錄中定義為外部使用者目錄。

功能管理員會執行 EPM System 管理工作，例如建立其他功能管理員、設定委派管理員、建立及提供應用程式和人工因素，以及設定 EPM System 稽核機制。我們已經在 *Oracle Enterprise Performance Management System User Security 管理手冊* 中說明功能管理員要執行的工作。

群組

群組是使用者或其他群組的容器。您可以從 Shared Services Console 建立及管理原生目錄群組。Shared Services Console 中會顯示所有已設定之使用者目錄下的群組。您可以提供這些群組，以授予已在 Shared Services 上登錄之 EPM System 產品的權限。

啟動 Shared Services Console

您使用 Oracle Hyperion Enterprise Performance Management Workspace 中的功能表選項存取 Oracle Hyperion Shared Services Console。

若要啟動 Shared Services Console：

1. 移至：

`http://web_server_name:port_number/workspace`

在上述 URL 中，`web_server_name` 代表正在執行 Oracle Hyperion Foundation Services 所用網路伺服器的電腦名稱，而 `port_number` 代表該網路伺服器的連接埠；例如 `http://myWebserver:19000/workspace`。

備註：

如果您是在安全的環境中存取 EPM Workspace，請使用 https (而非 http) 作為通訊協定，並使用安全的 Web 伺服器連接埠號碼。例如，使用下列 URL：`https://myserver:19043/workspace`。

2. 按一下 **啟動應用程式**。

備註：

快顯視窗封鎖程式可能會造成 EPM Workspace 無法開啟。

3. 在 **登入** 中，輸入您的使用者名稱和密碼。

起初，唯一可存取 Shared Services Console 的使用者是 Oracle Enterprise Performance Management System 管理員，其使用者名稱和密碼在部署處理程序期間就已指定。

4. 按一下**登入**。
5. 依序選取**導覽**、**管理**，然後選取 **Shared Services Console**。

2

已啟用 SSL 的 EPM System 元件

另請參閱：

- [假設](#)
- [資訊來源](#)
- [位置參照](#)
- [關於已啟用 SSL 的 EPM System 產品](#)
- [受支援的 SSL 案例](#)
- [必要的憑證](#)
- [在 SSL 卸載器終止 SSL](#)
- [EPM System 的完整 SSL 部署](#)
- [在 Web 伺服器終止 SSL](#)
- [適用於 Essbase 11.1.2.4 的 SSL](#)
- [適用於 Essbase 21c 的 SSL](#)

假設

- 您已決定部署拓樸，並指定要使用 SSL 保護的通訊連結。
- 您已從憑證授權單位 (CA) -- 不論是知名的 CA 或是您自己的 -- 取得必要的憑證，或是建立自簽憑證。請參閱[必要的憑證](#)。
- 您熟悉 SSL 的概念和程序，例如匯入憑證。
如需參考文件的清單，請參閱[資訊來源](#)。

資訊來源

已啟用 SSL 的 Oracle Enterprise Performance Management System 需要您先備妥幾個元件 (例如應用程式伺服器、Web 伺服器、資料庫及使用者目錄)，才能使用 SSL 來通訊。本文件假設您已經熟悉與這些已啟用 SSL 的元件相關的工作。

- **Oracle WebLogic Server**：請參閱 *Securing WebLogic Server Guide* 中的 [Configuring SSL](#)。
- **Oracle HTTP Server**：請參閱 *Oracle HTTP Server Administrator's Guide* 中的下列主題：
 - [Managing Security](#)
 - [Enabling SSL for Oracle HTTP Server](#)
- **使用者目錄**：請參閱使用者目錄廠商提供的文件。以下是幾個實用的連結：

- **Oracle Internet Directory**：請參閱 [Oracle Internet Directory Administrator's Guide](#) 和
- **Sun Java System Directory Server**：請參閱 *Sun Java System Directory Server Administration Guide* 中的 "[Directory Server Security](#)"
- **Active Directory**：請參閱 Microsoft 的相關文件。
- **資料庫**：請參閱資料庫廠商提供的文件。

位置參照

本文件參照至下列安裝及部署位置：

- **MIDDLEWARE_HOME** 參照至中介軟體元件 (例如 Oracle WebLogic Server) 以及 (選用) 至少一個 **EPM_ORACLE_HOME**。**MIDDLEWARE_HOME** 是在 Oracle Enterprise Performance Management System 產品安裝期間所定義的。預設的 **MIDDLEWARE_HOME** 目錄是 Oracle/Middleware。
- **EPM_ORACLE_HOME** 參照至包含支援 EPM System 產品所需之檔案的安裝目錄。**EPM_ORACLE_HOME** 位於 **MIDDLEWARE_HOME** 中。預設的 **EPM_ORACLE_HOME** 是 **MIDDLEWARE_HOME**/EPMSys11R1；例如 Oracle/Middleware/EPMSys11R1。

EPM System 產品安裝在 **EPM_ORACLE_HOME**/products 目錄中；例如 Oracle/Middleware/EPMSys11R1/products。

此外，在 EPM System 產品設定期間，部分產品會部署至 **MIDDLEWARE_HOME**/user_projects/epmsys1；例如 Oracle/Middleware/user_projects/epmsys1。

- **EPM_ORACLE_INSTANCE** 標示出部分產品部署元件的位置，這是在設定期間定義的。預設的 **EPM_ORACLE_INSTANCE** 位置是 **MIDDLEWARE_HOME**/user_projects/epmsys1；例如 Oracle/Middleware/user_projects/epmsys1。

關於已啟用 SSL 的 EPM System 產品

Oracle Enterprise Performance Management System 部署程序會自動將 Oracle 的 EPM System 產品部署成可同時在 SSL 模式和非 SSL 模式底下運作。

 **備註：**

- EPM System 僅支援透過 HTTP 與 JDBC 的 SSL。其不支援其他安全通訊標準，例如 Thrift 與 ODBC。
- 為了保護不受 Poodle (Padding Oracle On Downgraded Legacy Encryption) 漏洞威脅 (會對 SSLv3 通訊協定發動攻擊)，您必須停用伺服器的 SSLv3 支援，以及停用用來存取 EPM System 元件之瀏覽器的 SSLv3 支援。如需如何停用 SSLv3 支援的相關資訊，請參閱您伺服器和瀏覽器的文件。
- 如果您在設定 SSL 之後停用非 SSL 模式，EPM System 伺服器可能無法啟動。請啟用網域中所有 EPM System 伺服器的保護複製功能，以便讓這些伺服器在非 SSL 模式停用時能夠啟動。

當您在部署過程中指定 EPM System 的一般設定時，會指定您是否要啟用所有伺服器對伺服器通訊的 SSL 設定。

在部署期間選取 SSL 設定，並不會自動設定您的 SSL 環境。這只會在 Oracle Hyperion Shared Services Registry 中設下一個旗標，指出所有使用 Shared Services Registry 的 EPM System 元件在進行伺服器對伺服器的通訊時，都必須使用安全通訊協定 (HTTPS)。您必須完成其他程序，才能啟用環境的 SSL 設定。本文將會討論這些程序。

 **備註：**

當您重新部署應用程式時，系統會清除您指定要啟用 SSL 的自訂應用程式伺服器及 Web 伺服器設定。

 **備註：**

在 Enterprise Performance Management System 11.2.x 版中，不支援儲存區域建立公用程式 (RCU) 中 MS SQL 伺服器的安全通訊端層 (SSL)。

受支援的 SSL 案例

我們支援下列的 SSL 案例：

- 在 SSL 卸載器終止 SSL 請參閱在 [SSL 卸載器終止 SSL](#)。
- 完整的 SSL 部署。請參閱 [EPM System 的完整 SSL 部署](#)。

必要的憑證

SSL 通訊會使用憑證來建立元件與元件之間的信任。Oracle 建議您，使用知名第三方 CA 的憑證來啟用實際執行環境中 Oracle Enterprise Performance Management System 的 SSL 設定。

 **備註：**

EPM System 支援使用萬用字元憑證，讓您能使用單一 SSL 憑證來保護多個子網域。使用萬用字元憑證能減少管理時間和成本。

如果您要使用萬用字元憑證來為通訊加密，您必須停用 Oracle WebLogic Server 中的主機名稱驗證。

您必須為裝載 EPM System 元件的每個伺服器準備下列憑證：

- 根 CA 憑證

 **備註：**

如果您要使用來自知名第三方 CA 的憑證，且該 CA 的根憑證已經安裝在 Java 金鑰存放區中，您就不需要在 Java 金鑰存放區中安裝根 CA 憑證。

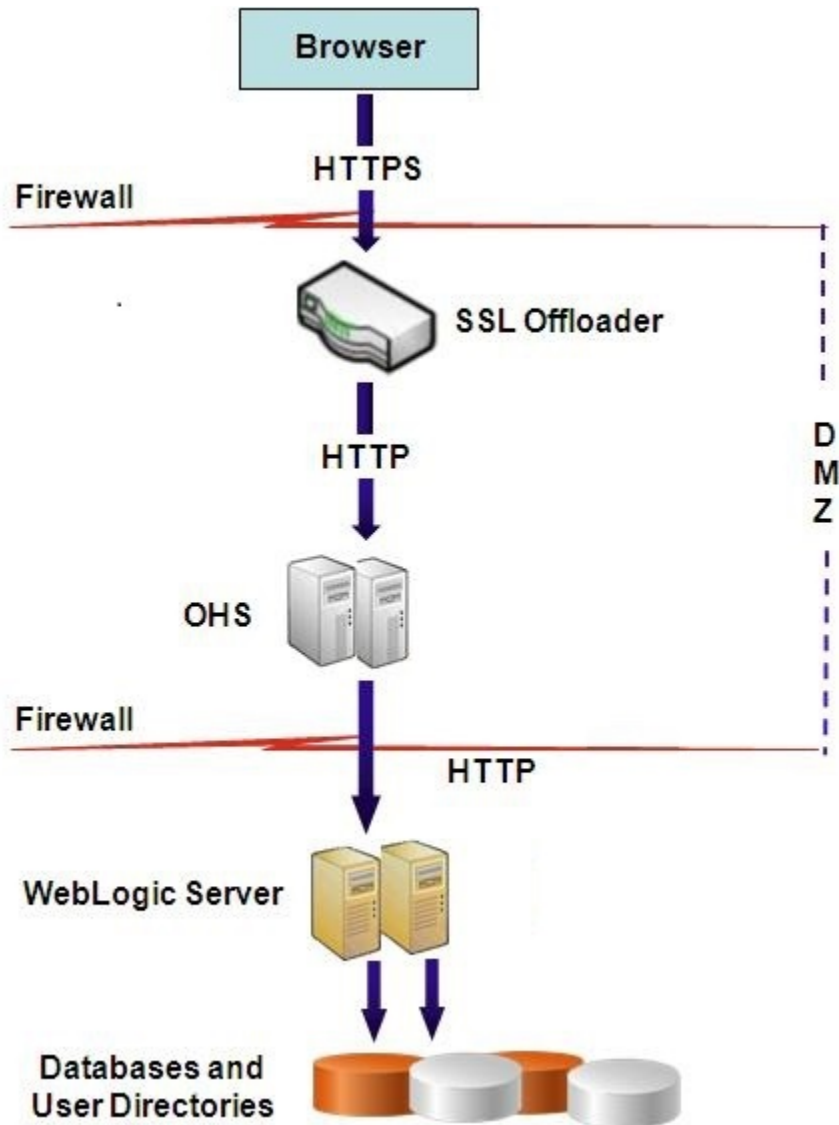
Firefox 和 Internet Explorer 都會預先載入知名第三方 CA 的憑證。如果您要擔任自己的 CA，就必須將您的 CA 根憑證匯入用戶端存取瀏覽器來使用的金鑰存放區。當您擔任自己的 CA 時，如果用戶端存取的瀏覽器無法使用您的 CA 根憑證，Web 用戶端就無法與伺服器建立 SSL 交握。

- 您部署中每個 Oracle HTTP Server 所用的已簽署憑證
- WebLogic Server 主機所用的已簽署憑證。這台機器上的受管理伺服器也會使用這個憑證。
- SSL 卸載器/負載平衡器所用的兩個憑證。其中一個憑證會用在外部通訊上，另一個則用於內部通訊

在 SSL 卸載器終止 SSL

部署架構

在這個案例中，SSL 是用來保護 Oracle Enterprise Performance Management System 用戶端 (例如瀏覽器) 和 SSL 卸載器之間的通訊連結。以下是概念的圖解說明：



假設

SSL 卸載器和負載平衡器

部署環境中必須要有設定完整的 SSL 卸載器，搭配負載平衡器。

您必須將負載平衡器設定成會把虛擬主機收到的所有要求轉送到 Oracle HTTP Server。

當 SSL 正在 Oracle HTTP Server (OHS) 或負載平衡器上終止時，您必須：

- 將每個 Logical Web Application 設為負載平衡器或 Oracle HTTP Server 的非 SSL 虛擬主機 (例如，empinternal.myCompany.com:80，其中 80 為非 SSL 連接埠)。開啟「組態」畫面，完成下列步驟：
 - 展開 **Hyperion Foundation** 組態任務。
 - 選取**設定 Web 應用程式的邏輯位址**。
 - 指定**主機名稱**、非 SSL 連接埠號碼及 SSL 連接埠號碼。

- 將外部 URL 設為負載平衡器或 Oracle HTTP Server 的啟用 SSL 虛擬主機 (例如，empexternal.myCompany.com:443，其中 443 為 SSL 連接埠)。開啟「組態」畫面，完成下列步驟：
 1. 展開 **Hyperion Foundation** 組態任務。
 2. 選取**設定公用設定**。
 3. 選取外部 URL 詳細資料底下的**啟用 SSL 卸載**。
 4. 指定**外部 URL 主機**和**外部 URL 連接埠**。

 **備註：**

使用 **configtool** 重新部署 Web 應用程式或設定 Web 伺服器將會取代 Logical Web Application 和外部 URL 的設定值。

虛擬主機

會在 SSL 卸載器終止的 SSL 組態在 SSL 卸載器/負載平衡器上會使用兩個伺服器別名 (例如 epm.myCompany.com 和 empinternal.myCompany.com)，其中一個用在卸載器與瀏覽器之間的外部通訊上，另一個則用於不同 EPM System 伺服器之間的內部通訊。請確保伺服器別名指向機器的 IP 位址，且可以透過 DNS 來解析。

您必須在卸載器/負載平衡器上安裝已簽署憑證，以支援卸載器與瀏覽器 (透過 epm.myCompany.com) 之間的外部通訊。

設定 EPM System

EPM System 元件的預設部署支援在 SSL 卸載器終止 SSL。您不需要進行其他的工作。

當您在設定 EPM System 時，請確保要把 Web 應用程式的邏輯位址指向專為內部通訊所建立的別名 (例如 empinternal.myCompany.com)。請參閱下列的資訊來源，以便安裝及設定 EPM System：

- *Oracle Enterprise Performance Management System 安裝與組態手冊*
- *Oracle Enterprise Performance Management System 安裝入門*
- *Oracle Enterprise Performance Management System Installation and Configuration Troubleshooting Guide (僅英文版)*

測試部署

當您完成部署程序之後，請連線到安全的 Oracle Hyperion Enterprise Performance Management Workspace URL，以確認一切都正常運作：

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

例如 https://epm.myCompany.com:443/workspace/index.jsp，其中 443 是 SSL 連接埠。

EPM System 的完整 SSL 部署

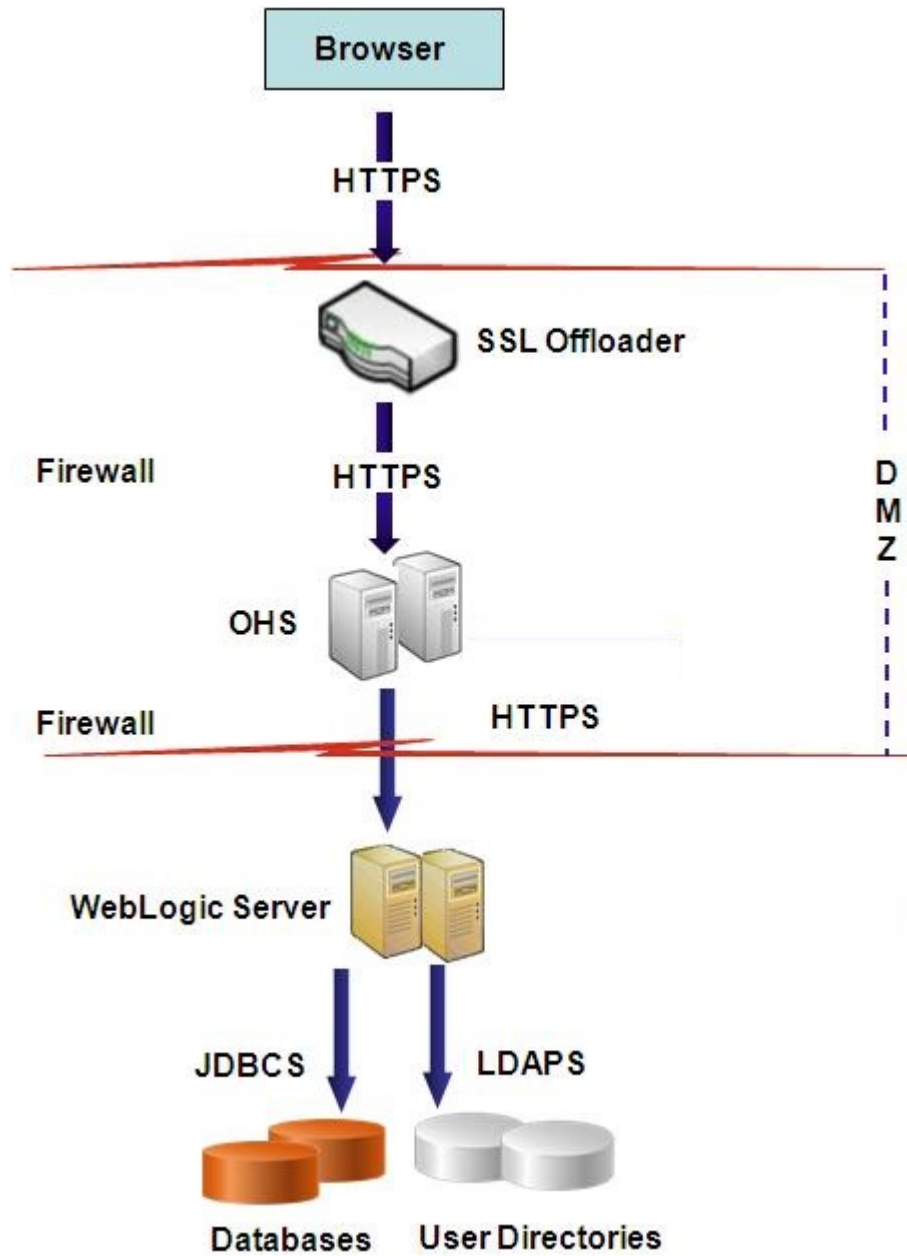
另請參閱：

- [部署架構](#)
- [假設](#)
- [針對完整 SSL 設定 EPM System](#)

部署架構

在完整的 SSL 模式中，跨所有受保護管道的通訊是利用 SSL 來保護的。這個 Oracle Enterprise Performance Management System 部署案例提供最嚴密的保護措施。

以下是概念的圖解說明：



假設

資料庫

資料庫和用戶端的 SSL 設定已啟用。如需瞭解如何啟用資料庫伺服器 and 用戶端的 SSL，請參閱該資料庫的相關文件。

EPM System

Oracle Enterprise Performance Management System 元件 (包括 Oracle WebLogic Server 和 Oracle HTTP Server) 都已安裝及部署。而且，您的 EPM System 環境已通過測試，確保一切在非 SSL 模式中都正常運作。請參閱下列資訊來源：

- [Oracle Enterprise Performance Management System 安裝與組態手冊](#)
- [Oracle Enterprise Performance Management System 安裝入門](#)
- [Oracle Enterprise Performance Management System Installation and Configuration Troubleshooting Guide \(僅英文版\)](#)

如果您計畫要啟用資料庫連線的 SSL，就必須在設定過程中，於每個資料庫的組態畫面上選取**進階設定**，然後指定下列必要的設定：

- 選取**使用安全的連線連接至資料庫 (SSL)**，然後輸入安全的資料庫 URL；例如
`jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=myDBhost)(PORT=1529)(CONNECT_DATA=(SERVICE_NAME=myDBhost.myCompany.com)))`
- **受信任的金鑰存放區**
- **受信任的金鑰存放區密碼**

如需詳細資料，請參閱 [Oracle Enterprise Performance Management System 安裝與組態手冊](#)。

SSL 卸載器和負載平衡器

部署環境中必須要有設定完整的 SSL 卸載器，搭配負載平衡器。

完整的 SSL 組態使用兩個伺服器別名，例如 `epm.myCompany.com`，以及 SSL 卸載器上的 `empinternal.myCompany.com`。其中一個會用在卸載器與瀏覽器之間的外部通訊上，另一個則用於不同 EPM System 伺服器之間的內部通訊。請確保伺服器別名指向機器的 IP 位址，且可以透過 DNS 來解析。

您必須將負載平衡器設定成會把虛擬主機收到的所有要求轉送到 Oracle HTTP Server。

您必須在卸載器/負載平衡器上安裝兩個已簽署憑證；其中一個 (透過 `epm.myCompany.com`) 支援卸載器與瀏覽器之間的外部通訊，另外一個則 (透過 `empinternal.myCompany.com`) 支援不同應用程式之間的內部通訊。Oracle 建議您，將這些憑證繫結至伺服器別名，以防止伺服器名稱曝光，同時提高安全性。

針對完整 SSL 設定 EPM System

另請參閱：

- [重新設定 EPM System 一般設定](#)
- [選用：安裝 WebLogic Server 的根 CA 憑證](#)
- [在 WebLogic Server 上安裝憑證](#)
- [設定 WebLogic Server](#)
- [啟用與已啟用 SSL 之 Oracle 資料庫的 HFM 伺服器連線](#)
- [Oracle HTTP Server 程序](#)
- [設定部署在 WebLogic Server 上的 EPM System Web 元件](#)
- [更新網域組態](#)
- [重新啟動伺服器和 EPM System](#)
- [測試部署](#)
- [設定已啟用 SSL 的外部使用者目錄](#)

重新設定 EPM System 的公用設定

您將在這個過程中，選取會強制 Oracle Enterprise Performance Management System 元件使用 SSL 通訊的設定。

備註：

如果您要啟用 Oracle Hyperion Financial Management Web 伺服器的 SSL：您在設定 Financial Management 之前，必須編輯 weblogic.xml 中 HFM WebApp 的階段作業描述元來保護 Cookie。

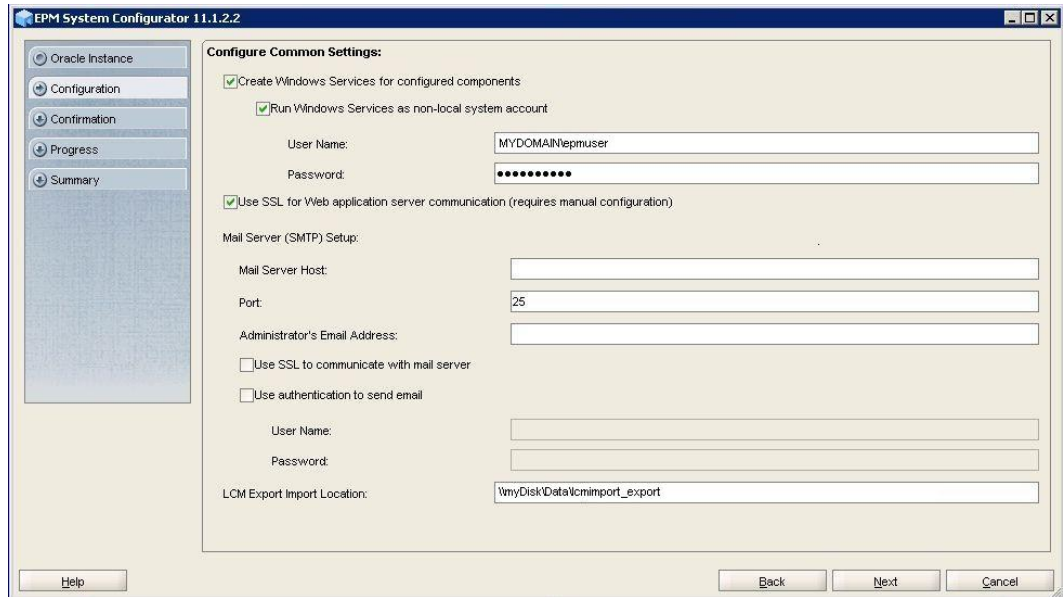
1. 利用 7 Zip 等工具來展開 Financial Management Web 存檔。
weblogic.xml 在存檔中的位置是
`EPM_ORACLE_HOME\products\FinancialManagement\AppServer\InstallableApps\HFMWebApplication.ear\HFMWeb.war\WEB-INF\weblogic.xml`。
2. 將下列指示詞新增到 weblogic.xml 的 HFM WebApp 階段作業描述元中：
`<cookie-secure>true</cookie-secure>`
3. 儲存 weblogic.xml
4. 當 7 Zip 詢問您是否要更新封存時，按一下**是**。

如何針對 SSL 重新設定 EPM System：

1. 啟動 EPM 系統組態程式。
2. 在**選取將套用組態的 EPM Oracle 例項**上，完成下列步驟：
 - a. 在 **EPM Oracle instance name** 中，輸入您一開始設定 EPM System 元件時所用的例項名稱。
 - b. 按一下**下一步**。
3. 在「設定」畫面中，完成下列步驟：
 - a. 清除**全部取消勾選**。
 - b. 展開 **Hyperion Foundation** 設定工作，然後選取**設定公用設定**。
 - c. 按一下**下一步**。
4. 在**設定公用設定**中，完成下列步驟：

注意：

在選取設定來使用 SSL 與電子郵件伺服器通訊之前，請確保該電子郵件伺服器已設定為使用 SSL。



- a. 選取**使用 SSL 以供 Java Web 應用程式伺服器通訊使用 (需要手動組態)**，以指定 EPM System 一定要使用 SSL 來通訊。
 - b. **選用：**在**郵件伺服器主機**和**連接埠**中輸入相關資訊。若要支援 SSL 通訊，您必須指定 SMTP 郵件伺服器要使用的安全連接埠。
 - c. **選用：**若要支援搭配 SMTP 郵件伺服器的 SSL 通訊，請選取**使用 SSL 與郵件伺服器通訊**。
 - d. 在剩餘的欄位中選取或輸入相關設定。
 - e. 按一下**下一步**。
5. 在隨後出現的 EPM 系統組態程式 畫面中，按一下**下一步**。
 6. 當您完成部署程序之後，系統會顯示「摘要」畫面。按一下**完成**。

選用：安裝 WebLogic Server 的根 CA 憑證

大多數知名第三方 CA 的根憑證，都已經安裝在 JVM 金鑰存放區中。如果您沒有要使用知名第三方 CA 的憑證，請完成本節提供的程序 (不建議)。預設的 JVM 金鑰存放區位置在 `MIDDLEWARE_HOME/jdk/jre/lib/security/cacerts`。

備註：

請在每個 Oracle Enterprise Performance Management System 伺服器上執行這個程序。

如何安裝根 CA 憑證：

1. 將根 CA 憑證複製到已安裝 Oracle WebLogic Server 之機器上的某個本機目錄中。
2. 在某個主控台中，將目錄變更至 `MIDDLEWARE_HOME/jdk/jre/bin`。

3. 執行 **Keytool** 命令 (例如下列命令)，以便將根 CA 憑證安裝到 JVM 金鑰存放區中：

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

例如，您可以使用下列命令，將儲存在目前目錄中的憑證 `CAcert.crt` 新增到 JVM 金鑰存放區中，其中 `Blister` 是該憑證在金鑰存放區中的別名。我們假設 `Storepass` 為 `example_pwd`。

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

 **備註：**

上述命令和範例使用某些在利用 **Keytool** 匯入憑證時所用的語法。如需完整的匯入語法清單，請參閱 **Keytool** 的相關文件。

在 WebLogic Server 上安裝憑證

預設的 Oracle WebLogic Server 安裝使用示範憑證來支援 SSL。Oracle 建議您，安裝來自知名第三方廠商的憑證，以強化您環境的安全防護。

請在每台裝載 WebLogic Server 的機器上，使用工具 (例如金鑰工具) 來建立自訂金鑰存放區，以便儲存 WebLogic Server 和 Oracle Enterprise Performance Management System Web 元件的已簽署憑證。

如何建立自訂金鑰存放區及匯入憑證：

1. 在某個主控台中，將目錄變更至 `MIDDLEWARE_HOME/jdk/jre/bin`。
2. 執行 **Keytool** 命令 (例如下列命令)，以便在現有的目錄中建立自訂金鑰存放區 (藉由 `-keystore` 指示詞來辨識)。

```
keytool -genkey -dname "cn=myserver, ou=EPM, o=myCompany, c=US" -alias epm_ssl -keypass password -keystore C:\oracle\Middleware\EPMSys11R1\ssl\keystore -storepass password -validity 365 -keyalg RSA
```

 **備註：**

您設定的一般名稱 (`cn`) 必須與伺服器名稱相符。如果您把完整的網域名稱 (FQDN) 當做 `cn`，就必須在部署 Web 元件時使用 FQDN。

3. 產生憑證要求。

```
keytool -certreq -alias epm_ssl -file C:/certs/epmssl_csr -keypass password -storetype jks -keystore C:\oracle\Middleware\EPMSys11R1\ssl\keystore -storepass password
```

4. 取得 WebLogic Server 機器的已簽署憑證。
5. 將已簽署憑證匯入金鑰存放區：

```
keytool -import -alias epm_ssl -file C:/certs/epmssl.crt -keypass
password -keystore C:\Oracle\Middleware\EPMSysstem11R1\ssl\keystore -
storepass password
```

設定 WebLogic Server

您在部署 Oracle Enterprise Performance Management System Web 元件之後，必須針對 SSL 通訊來設定這些元件。

如何針對 SSL 設定 Web 元件：

1. 啟動 Oracle WebLogic Server，方法是執行 `MIDDLEWARE_HOME/user_projects/domains/EPMSysstem/bin/startWebLogic.cmd`：
2. 存取下列 URL 來啟動 WebLogic Server 管理主控台：

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

例如，若要存取已部署到 myServer 上之預設連接埠的 WebLogic Server，您應該要使用 `http://myServer:7001/console`。

3. 在「歡迎」畫面中，輸入您在 EPM System Configurator 中指定的 WebLogic Server 管理員使用者名稱和密碼。
4. 在**變更中心**中，按一下**鎖定和編輯**。
5. 在主控台的左窗格中，展開**環境**，然後選取**伺服器**。
6. 在「伺服器摘要」畫面中，按一下您要啟用 SSL 的伺服器名稱。

例如，若要啟用 Oracle Hyperion Foundation Services 元件的 SSL，您要運用 EPMServer0 伺服器。

7. 清除**啟用監聽連接埠**以停用 HTTP 監聽連接埠。
8. 確保**啟用 SSL 監聽連接埠**已選取。
9. 在**SSL 監聽連接埠**中，輸入該伺服器要用來監聽要求的 SSL 監聽連接埠。
10. 若要指定您要使用的識別和信任金鑰存放區，請選取**金鑰存放區**來開啟「金鑰存放區」頁籤。
11. 按一下**變更**。
12. 選取某個選項：

- **自訂識別和自訂信任**：如果您不要使用知名第三方 CA 的伺服器憑證
- **自訂識別和 Java 標準信任**：如果您要使用知名第三方 CA 的伺服器憑證

13. 按一下**儲存**。
14. 在**自訂識別金鑰存放區**中，輸入已簽署 WebLogic Server 憑證安裝之處的金鑰存放區路徑。
15. 在**自訂識別金鑰存放區類型**中，輸入 `jks`。

16. 在**自訂識別金鑰存放區密碼**和**確認自訂識別金鑰存放區密碼**中，輸入金鑰存放區密碼。
17. 如果您已在**金鑰存放區**中選取**自訂識別和自訂信任**：
 - 在**自訂信任金鑰存放區**中，輸入已簽署您伺服器憑證之 CA 的根憑證所在之處的自訂金鑰存放區路徑。
 - 在**自訂信任金鑰存放區類型**中，輸入 jks。
 - 在**自訂信任金鑰存放區密碼**和**確認自訂信任金鑰存放區密碼**中，輸入金鑰存放區密碼。
18. 按一下**儲存**。
19. 指定 SSL 設定：
 - 選取 **SSL**。
 - 在**私密金鑰別名**中，輸入您在匯入已簽署 WebLogic Server 憑證時所指定的別名。
 - 在**私密金鑰密碼**和**確認私密金鑰密碼**中，輸入要用來擷取私密金鑰的密碼。
 - 按一下**儲存**。

 **備註：**

如果您要使用 SHA-2 憑證，就必須針對每個用來支援 EPM System 的受管理伺服器選取**使用 JSSE SSL** 設定。這個設定位於 SSL 頁面的「進階」頁籤上。您必須重新啟動 WebLogic Server，才能啟動這個變更。

20. 啟用伺服器的安全複製功能：
 - a. 在主控台的左窗格中，展開**環境**，然後按一下**叢集**。
 - b. 在「叢集摘要」中，按一下您要啟用安全複製功能的伺服器名稱 (例如 Foundation Services)。
 - 此時系統會顯示已選取伺服器之「設定」畫面的「組態」頁籤。
 - c. 按一下**複製**來開啟「複製」頁籤。
 - d. 選取**啟用安全複製**。您可能必須先按一下**鎖定和編輯**，才能選取這個選項。
 - e. 按一下**儲存**。
21. 針對屬於這個主機的每個受管理伺服器完成步驟 6 至步驟 20。
22. 啟用安全複製功能，以便提供叢集複製呼叫所用的通道。
如需詳細資料，請參閱 Oracle metalink 文件 1319381.1。
 - 在「管理主控台」中，展開**環境**，然後選取**叢集**。
 - 選取**複製**。
 - 在**複製**中，選取 (核取) **啟用安全複製**。
 - 按一下**儲存**。
23. 在**變更中心**中，按一下**啟動變更**。

啟用與已啟用 SSL 之 Oracle 資料庫的 HFM 伺服器連線

HFM 資料來源和 Oracle 資料庫之間的網路連線可以使用 SSL 進行加密。為使此功能正常運作，Oracle Wallet 必須按照 [Oracle 文件](#) 中所述進行設定。TNS 監聽程式也必須設定為在新的連接埠上監聽 SSL 加密連線。最後，需要將適當的憑證載入主控 HFM 資料來源之伺服器上的金鑰存放區和信任存放區中。以下指示來自 [Oracle 資料庫文件](#)。

先決條件

請確保滿足下列先決條件，再繼續進行以下步驟：

- 正常運作的資料庫伺服器。
- 請確保沒有本機或網路防火牆阻止與連接埠 (正在執行已啟用 SSL 之 TNS 監聽程式) 上的伺服器進行任何通訊。

在下方的範例中，使用了在 MS Windows Server 2016 上執行的 Oracle 12c (12.1.0.2) 版本。這些指示在 Linux 安裝上同樣適用，前提是為公事包檔案指定的路徑是 Linux 檔案系統路徑，並且為資料庫伺服器上使用的殼層正確變更了環境變數替代。這些相同的指示已成功用於 19c 開發和支援例項。

本文中的範例使用自簽憑證，但如果您偏好的話，也可以使用適當憑證授權單位頒發的憑證。請參閱 [Oracle 資料庫文件](#)，瞭解安裝憑證授權單位頒發的憑證時要遵循的確切步驟。

設定 Oracle 資料庫

若要設定 Oracle 資料庫，請執行以下步驟：

1. 在資料庫伺服器上建立新的自動登入公事包。

備註：

僅當先前未建立過 Oracle Wallet 時，才需要執行這些步驟。如果在資料庫伺服器上使用 GUI Oracle Wallet 工具，則不需要執行下列步驟。

```
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
```

您可以忽略任何提示您在 orapki 命令行上使用 -auto_login_local 的訊息。如果您遇到 SSL 驗證失敗錯誤，請參閱 [文件 ID 2238096.1](#) 以排解此問題。

此外，檢查檔案 cwallet.sso (在公事包目錄下) 的安全性權限，並確保 Oracle 監聽程式服務使用者具有此檔案的讀取權限。如果沒有讀取權限，SSL 交握將在稍後失敗。如果 Oracle 資料庫與建議的 Oracle 使用者一起安裝，而該使用者不被允許登入，則會出現此情況。如果 Oracle 資料庫與 Oracle 使用者一起安裝，則必須以其他使用者身分執行 TNS 監聽程式。

2. 建立自簽憑證並將它載入公事包

```
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
```

在上方的範例中，密碼 password1 必須與步驟 1 中指定的密碼相符。

3. 匯出新建立的自簽憑證

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

4. 將匯出的 Base64 憑證檔案複製到 HFM 伺服器。

5. 設定 SQL*NET 和 TNS 監聽程式：

- a. 識別資料庫伺服器上未使用的連接埠。下方的範例在連接埠 1522 上建立新的監聽程式。用於 SSL 連線的典型連接埠是 2484，您可以使用任何可用的連接埠。您必須檢查要使用的連接埠在資料庫伺服器上是否可用，才能繼續進行並視需要調整。

- b. 更新 SQLNET.ORA。WALLET_LOCATION 宣告的 DIRECTORY 元素必須指向在上述步驟 1 中建立的公事包。

```
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 以定義新的監聽程式。使用在上述 步驟 5a 中識別的連接埠。

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

- d. 在 TNSNAMES.ORA 中建立新的連接埠項目。

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

您必須指定在上述 **步驟 5a** 中識別並在 **步驟 5c** 中使用的相同連接埠。

- e. 重新啟動 TNS 監聽程式。

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

- f. 驗證新的 TNS 監聽程式是否正常運作

```
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)
```

設定 HFM 伺服器以使用 SSL 資料庫連線

將資料庫的憑證新增至 HFM 伺服器上的信任存放區

下列步驟必須在運行 HFM 資料來源的每部 EPM 伺服器上執行。下方使用的 `%MW_HOME%` 環境變數是 Oracle Middleware 安裝的位置。依預設，在 EPM 安裝過程中不會建立此環境變數，此處用於顯示 EPM 安裝的父目錄。

EPM 安裝的位置由 `EMP_ORACLE_HOME` 環境變數指定。下方的範例將金鑰存放區和信任存放區放置在與 EPM 安裝共存的目錄中。金鑰存放區和信任存放區檔案可以位於 HFM 伺服器檔案系統上的任何位置。

1. 在 `%MW_HOME%` 下建立新的目錄，以儲存 Java 金鑰存放區和 PKCS12 信任存放區。
 - a. `cd %MW_HOME%`
 - b. `mkdir certs`
2. 從 JDK 複製 Java 金鑰存放區的 `cacerts` 檔案。
 - a. `cd %MW_HOME%\certs`
 - b. 複製 `%MW_HOME%\jdk1.8.0_181\jre\lib\security\cacerts` 複製並使用 JDK 的金鑰存放區而不是 JDK 的預設金鑰存放區的原因是，如果升級 JDK 並刪除了先前的 JDK，則插入至預設金鑰存放區中的金鑰和憑證將會遺失。
3. 將 Base 64 憑證複製到 `%MW_HOME%\certs`。
4. 將憑證匯入 Java 金鑰存放區檔案 `testing_cacerts`。
 - a. 例如，`keytool -importcert -file bur00cbb-certificate.crt -keystore testing_cacerts -alias "myserver"`
 - i. 您必須指定檔案金鑰存放區的密碼。
 - ii. 您應將 "myserver" 取代為資料庫伺服器的完整網域。
 - b. 當系統提示您並詢問是否應信任憑證時，請指定 `y`。
5. 從 JDK 的 Java 金鑰存放區檔案建立 PKCS12 格式的信任存放區。例如，

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

更新 HFM JDBC 連線以使用 SSL

1. 重新設定 HFM 資料庫 JDBC 連線以使用 SSL。
 - a. 啟動 EPM 組態工具。
 - i. 選取 **Financial Management** 節點下的 **設定資料庫和部署到應用程式伺服器** 節點。
 - ii. 按一下 **下一步**。
 - iii. 對 HFM JDBC 連線執行上述每個步驟
 - i. 在連接埠、服務名稱、使用者名稱和密碼欄中輸入連線的 SSL 連接埠、服務名稱、使用者名稱和密碼。

- ii. 按一下 (+) 來開啟**進階資料庫選項**。
 - iii. 選取**使用安全的連線**核取方塊。
 - iv. 輸入在**步驟 2**中建立的 Java 金鑰存放區的位置。
 - v. 按一下**套用**。
 - vi. 按一下 (+) 來開啟**進階資料庫選項**。
 - vii. 按一下**編輯並使用已修改的 JDBC URL**。請注意，不應對顯示的 JDBC URL 進行任何變更。
 - viii. 按一下**套用**。
 - ix. 按一下**下一步**。
- b. 按照 EPM 文件中所述完成部署 HFM 應用程式的剩餘步驟。
2. 開啟命令視窗或殼層來手動更新 EPM 登錄，以便資料來源使用的 ODBC 連線可以啟用 SSL。
- 執行以下列出的每個命令：

```
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
```

在上方的範例中，路徑 C:\Oracle\Middleware 是步驟 1、2 和 3 中 %MW_HOME% 的值。

僅當使用自簽憑證時，特性 FINANCIAL_MANAGEMENT_PRODUCT/DATABASE_CONN/@ODBC_VALIDATESERVERCERTIFICATE 才應設定為偽。FINANCIAL_MANAGEMENT_PRODUCT/DATABASE_CONN/@ODBC_TRUSTSTOREPASSWORD 的值應該是在**步驟 2**中複製的原始 Java 金鑰存放區的密碼。

更新 HFM 使用的 TNS 名稱項目

編輯 TNSNAMES.ORA 以建立新的項目並重新命名舊的項目。下列範例顯示 HFM 伺服器上已套用必要變更的更新 TNSNAMES.ORA 檔案。進行這些變更的原因是 HFM 會尋找並使用名為 HFMTNS 的 TNS 名稱項目。此項目必須變更通訊協定和連接埠，XFMDDataSource 才能正常運作。

```
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 項目已重新命名為 HFMTNS_UNENC。新的 HFMTNS 是藉由複製 HFMTNS_UNENC 項目，將其重新命名為 HFMTNS. 而建立的。接著通訊協定更新為 TCPS，連接埠則變更為 1522。指定的連接埠必須與 TNS LISTENER.ORA 檔案中指定的連接埠相同。

Oracle HTTP Server 程序

針對 Oracle HTTP Server 建立公事包及安裝憑證

預設的公事包會自動與 Oracle HTTP Server 一起安裝。您必須針對部署中的每個 Oracle HTTP Server 設定真正的公事包。

注意：從 11.2.x 開始，Oracle Wallet Manager 不會與 Oracle HTTP 伺服器一起安裝。僅當您安裝 Oracle 資料庫用戶端時，才會安裝 Oracle Wallet Manager。您必須使用資料庫用戶端提供的公事包管理員來建立公事包並匯入憑證。如果您要為 SSL 設定 Oracle HTTP 伺服器，請確保一律在安裝 EPM 系統產品的過程中安裝 Oracle 資料庫用戶端 64 位元。

如何建立及安裝 Oracle HTTP Server 憑證：

1. 在裝載 Oracle HTTP Server 的每台機器上啟動 Wallet Manager。
 - 請依序選取 **Start**、**所有程式**、**Oracle-OHxxxxxx**、**整合式管理工具**及 **Wallet Manager**。
 - xxxxxx 是 Oracle HTTP Server 例項編號。
2. 建立空白的全新公事包。
 - a. 在 Oracle Wallet Manager 中，依序選取**公事包**及**新建**。
 - b. 按一下**是**來建立預設的公事包目錄，或是按一下**否**，以便在您選擇的位置建立公事包檔案。
 - c. 在「新公事包」畫面的**公事包密碼**和**確認密碼**中，輸入您要使用的密碼。
 - d. 按一下**確定**。
 - e. 在確認對話方塊中，按一下**否**。
3. **選用：**如果您沒有使用 Oracle HTTP Server 知道的 CA，請將根 CA 憑證匯入公事包。
 - a. 在 Oracle Wallet Manager 中，按滑鼠右鍵按一下**受信任的憑證**，然後選取**匯入受信任的憑證**。
 - b. 瀏覽並選取根 CA 憑證。
 - c. 選取**開啟**。

4. 建立憑證要求。
 - a. 在 Oracle Wallet Manager 中，按滑鼠右鍵按一下**憑證：[空白]**，然後選取**新增憑證要求**。
 - b. 在「建立憑證要求」中，輸入必要資訊。
針對一般名稱，請輸入完整的伺服器名稱；例如您系統之 hosts 檔案中的 `epm.myCompany.com` 或 `epminternal.myCompany.com`。
 - c. 按一下**確定**。
 - d. 在確認對話方塊中按一下**確定**。
 - e. 用滑鼠右鍵按一下您建立的憑證要求，然後選取**匯出憑證要求**。
 - f. 指定憑證要求檔案的名稱。
5. 利用憑證要求檔案，取得 CA 的已簽署憑證。
6. 匯入已簽署憑證。
 - a. 在 Oracle Wallet Manager 中，用滑鼠右鍵按一下您用來取得已簽署憑證的憑證要求，然後選取**匯入使用者憑證**。
 - b. 在「匯入憑證」中，按一下**確定**以便從檔案匯入憑證。
 - c. 在「匯入憑證」中，選取憑證檔案，然後按一下**開啟**。
7. 將公事包儲存在您方便存取的位置；例如 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/keystores/epmsystem`。
8. 選取**公事包**，然後選取**自動登入**以啟動自動登入功能。

使用 ORAPKI 設定 Oracle Wallet (Linux)

若要使用 ORAPKI 命令行設定 Oracle Wallet，請完成下列步驟：

1. 為您的公事包建立資料夾：

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

2. 將 `orapki` 公用程式位置新增至您的路徑：

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

3. 建立公事包來儲存您的憑證：

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

如果未在命令行中指定密碼，此命令會提示您輸入並重新輸入公事包密碼。它會在 `-wallet` 指定的位置建立公事包。

4. 產生憑證簽署要求 (CSR)，並將其新增至您的公事包：

```
$ 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. 將根目錄與中繼憑證新增至受信任 Keystore

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

6. 使用您的 CA (憑證授權單位) 來簽署 CSR (憑證簽署要求)。若要匯出 Oracle Wallet 中的憑證要求，請執行下列動作：

```
$ 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. 將簽署的 CSR 匯入至公事包：

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

8. 若要顯示公事包的內容，請執行下列動作：

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

啟用 Oracle HTTP Server 的 SSL 設定

當您設定用來裝載 Oracle HTTP Server 的每個機器上的 Web 伺服器之後，請更新 Oracle HTTP Server 設定檔案，方法是將預設公事包的位置，替換成您建立的公事包位置。

如何針對 SSL 設定 Oracle HTTP Server：

1. 重新設定您部署中每個 Oracle HTTP Server 主機上的 Web 伺服器。
2. 啟動例項的 EPM System Configurator。
3. 在選取設定工作畫面中完成下列步驟，然後按一下**下一步**。
 - a. 清除**全部取消勾選**的選取。
 - b. 展開 **Hyperion Foundation** 工作群組，然後選取**設定 Web 伺服器**。
4. 在**設定 Web 伺服器**中，按一下**下一步**。
5. 在**確認**中，按一下**下一步**。
6. 在**摘要**中，按一下**完成**。
7. 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/fmwconfig/components/OHS/ohs_component/ssl.conf`。
8. 確保您要使用的 SSL 連接埠列在 OHS 監聽連接埠下方，與下方範例類似：

如果您要把 19443 當做 SSL 通訊連接埠，該項目應該會是：

```
Listen 19443
```

9. 將 SSLSessionCache 參數值設定為無。
10. 更新您部署中每個 Oracle HTTP Server 的組態設定。
 - a. 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/httpConfig//ohs/config/fmwconfig/components/OHS/ohs_component/ssl.conf`。
 - b. 尋找 SSLWallet 指示詞，並將其值改變成會指向您安裝憑證所在的公事包。如果您之前將公事包建立在 `EPM_ORACLE_INSTANCEhttpConfig/ohs/config/OHS/ohs_component/keystores/epmsystem`，您的 SSLWallet 指示詞可能會跟下列範例一樣：

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

- c. 儲存並關閉 `ssl.conf`。
11. 更新您部署中每個 Oracle HTTP Server 上的 `mod_wl_ohs.conf`。
 - a. 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/httpConfig//ohs/config/fmwconfig/components/OHS/ohs_component/mod_wl_ohs.conf`。
 - b. 確保 WLSSLWallet 指示詞會指向 SSL 憑證所在的 Oracle Wallet。

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

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

- c. 將 SecureProxy 指示詞的值設定為 ON。
- d. 確保已部署的 Oracle Enterprise Performance Management System 元件的 LocationMatch 定義，類似於下列的 Oracle Hyperion Shared Services 範例；該定義假設一個 Oracle WebLogic Server 叢集 (在使用 SSL 連接埠 28443 的 myserver1 和 myserver2 上)：

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

- e. 儲存並關閉 `mod_wl_ohs.conf`。

設定部署在 WebLogic Server 上的 EPM System Web 元件

您在部署 Oracle Enterprise Performance Management System Web 元件之後，必須針對 SSL 通訊來設定這些元件。

如何針對 SSL 設定 Web 元件：

1. 執行儲存在 `EPM_ORACLE_INSTANCE/domains/EPMSysSystem/bin/startWebLogic.cmd` 的檔案來啟動 Oracle WebLogic Server：
2. 存取下列 URL 來啟動 WebLogic Server 管理主控台：

`http://SERVER_NAME:Port/console`

例如，若要存取已部署到 myServer 上之預設連接埠的 WebLogic Server，您應該要使用 `http://myServer:7001/console`。

3. 在「歡迎」畫面上，輸入使用者名稱和密碼來存取 EPMSysSystem。該使用者名稱和密碼是您於設定期間在 EPM System Configurator 中指定的。
4. 在**變更中心**中，按一下**鎖定和編輯**。
5. 在主控台的左窗格中，展開**環境**，然後選取**伺服器**。
6. 在「伺服器摘要」畫面中，按一下您要啟用 SSL 的伺服器名稱。

例如，如果您已安裝所有的 Oracle Hyperion Foundation Services 元件，您可以啟用下列伺服器的 SSL 設定：

- CalcManager
 - FoundationServices
7. 清除**啟用監聽連接埠**以停用 HTTP 監聽連接埠。
 8. 確保**啟用 SSL 監聽連接埠**已選取。
 9. 在 **SSL 監聽連接埠**中，輸入 WebLogic Server 的 SSL 監聽連接埠。
 10. 指定要使用的識別和信任金鑰存放區。
 - 選取**金鑰存放區**來開啟「金鑰存放區」頁籤。
 - 在**金鑰存放區**中，選取下列某個選項：
 - a. 選取**金鑰存放區**來開啟「金鑰存放區」頁籤。
 - b. 在**金鑰存放區**中，選取下列某個選項：
 - **自訂識別和自訂信任**：如果您不要使用知名第三方 CA 的伺服器憑證
 - **自訂識別和 Java 標準信任**：如果您要使用知名第三方 CA 的伺服器憑證
 - c. 在**自訂識別金鑰存放區**中，輸入已簽署 WebLogic Server 憑證安裝之處的金鑰存放區路徑。
 - d. 在**自訂識別金鑰存放區類型**中，輸入 `jks`。
 - e. 在**自訂識別金鑰存放區密碼**和**確認自訂識別金鑰存放區密碼**中，輸入金鑰存放區密碼。
 - f. 如果您已在**金鑰存放區**中選取**自訂識別和自訂信任**：
 - 在**自訂信任金鑰存放區**中，輸入已簽署您伺服器憑證之 CA 的根憑證所在之處的自訂金鑰存放區路徑。
 - 在**自訂信任金鑰存放區類型**中，輸入 `jks`。
 - 在**自訂信任金鑰存放區密碼**和**確認自訂信任金鑰存放區密碼**中，輸入金鑰存放區密碼。

- g. 按一下**儲存**。
11. 指定 SSL 設定。
 - 選取 **SSL**。
 - 在**私密金鑰別名**中，輸入您在匯入已簽署 WebLogic Server 憑證時所指定的別名。
 - 在**私密金鑰密碼**和**確認私密金鑰密碼**中，輸入要用來擷取私密金鑰的密碼。
 - **僅限 Oracle Hyperion Provider Services Web 應用程式**：如果您要使用萬用字元憑證來為 WebLogic Server 與其他 EPM System 伺服器元件之間的通訊加密，請停用 Provider Services Web 應用程式的主機名稱驗證功能。
 - 選取 **進階**。
 - 在**主機名稱驗證**中，選取**無**。
 - 按一下**儲存**。
12. 在**變更中心**中，按一下**啟動變更**。

更新網域組態

此程序會更新網域組態。開始此程序前，請先建立部署的完整備份。Oracle 建議您在變更生產部署前，先於測試部署中測試此程序。

更新網域組態：

1. 瀏覽至 MIDDLEWARE_HOME/oracle_common/bin directory 目錄：
cd MIDDLEWARE_HOME/oracle_common/bin
2. 設定 ORACLE_HOME、WL_HOME 及 JAVA_HOME。
set ORACLE_HOME= /Oracle/Middleware
set WL_HOME= /Oracle/Middleware/wlserver
set JAVA_HOME= /Oracle/Middleware/jdk
3. 在 WebLogic 主控台中，對管理伺服器啟用 HTTP 連接埠。
4. 使用類似下面的命令建立金鑰存放區：
libovdconfig.bat -host HOSTNAME -port 7001 -userName USERNAME -domainPath %MWH%\user_projects\domains\EPMSystem -createKeystore

在此命令中，分別以 WebLogic 伺服器的主機名稱與管理員的使用者名稱取代 HOSTNAME 與 USERNAME。確認輸出回報已順利建立 OVD 金鑰存放區。
5. 從管理伺服器匯出 SSL 憑證。

Note:

此步驟僅適用於內嵌的 LDAP (預設認證程式)。對於其他 LDAP，必須使用適當的 LDAP 專用命令才能匯出憑證。憑證檔案格式必須為 **Base 64 編碼 x.509**

- a. 使用 Internet Explorer 連線至 https://HOSTNAME:7002/console 以存取 WebLogic 管理主控台
- b. 依序按一下**檢視憑證與詳細資料**，然後選取**複製到檔案**以匯出 SSL 憑證。

- c. 將憑證另存為 **Base 64 編碼 x.509** 憑證檔案並儲存至本機目錄，例如
C:\certificate\slc17rby.cer。
 - d. 將憑證移至伺服器。
6. 使用金鑰工具將憑證匯入至您在步驟 4 所建立的金鑰存放區。使用類似下列內容的命令 (假設 *JAVA_HOME* 與 *keytool* 執行檔位於路徑中)：
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, for
example:

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

 **Note:**

- 在此命令中使用的密碼必須符合在步驟 4 產生金鑰存放區時的密碼。
- *CERTIFICATE_PATH* 是憑證的位置與名稱
- 別名可以是您選取的任意別名。

順利匯入憑證後，金鑰工具會顯示訊息：憑證已新增至金鑰儲存庫中。

7. 在 WebLogic 主控台中，除了啟用 HTTP 連接埠之外，也一併啟用管理伺服器的 SSL 連接埠。
8. 重新啟動 WebLogic 管理伺服器與受管理伺服器。
9. 使用安全的連線登入 Oracle Hyperion Enterprise Performance Management Workspace 以確認所有作業皆可正常運行。

重新啟動伺服器和 EPM System

請重新啟動部署中的所有伺服器，然後啟動每台伺服器上的 Oracle Enterprise Performance Management System。

測試部署

當您完成 SSL 的部署之後，請確認一切是否都正常運作。

如何測試部署：

1. 利用瀏覽器來存取安全的 Oracle Hyperion Enterprise Performance Management Workspace URL：

如果您把 `epm.myCompany.com` 當做外部通訊用的伺服器別名，且把 4443 當做 SSL 連接埠，則 EPM Workspace URL 為

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

2. 在「登入」畫面上，輸入使用者名稱與密碼。
3. 按一下**登入**。
4. 確認您可以安全地存取已部署的 Oracle Enterprise Performance Management System 元件。

設定已啟用 SSL 的外部使用者目錄

假設

- 您計畫在 Oracle Hyperion Shared Services Console 中設定的外部使用者目錄已啟用 SSL。
- 如果您沒有使用知名第三方 CA 的憑證來啟用使用者目錄的 SSL，您必須要有已簽署伺服器憑證之 CA 的根憑證副本。

匯入根 CA 憑證

如果您沒有使用知名第三方 CA 的憑證來啟用使用者目錄的 SSL，就必須將已簽署伺服器憑證之 CA 的根憑證匯入下列金鑰存放區：



備註：

在應用程式部署期間，WebLogic 會在 `setDomainEnv.sh` 或 `setDomainEnv.cmd` 中，新增指向 `DemoTrust.jks` 的 `-Djavax.net.ssl.trustStore` 指示詞。如果您不打算使用預設的 WebLogic 憑證，請移除 `setDomainEnv.sh` 或 `setDomainEnv.cmd` 中的 `-Djavax.net.ssl.trustStore`。

請使用某個工具 (例如金鑰工具) 來匯入根 CA 憑證。

- 所有的 Oracle Enterprise Performance Management System 伺服器：

JVM 金鑰存放區： `MIDDLEWARE_HOME/jdk/jre/lib/security/cacerts`

- 每台 EPM System 元件主機上的 JVM 所用的金鑰存放區。根據預設，EPM System 元件會使用下列金鑰存放區：

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

設定外部使用者目錄

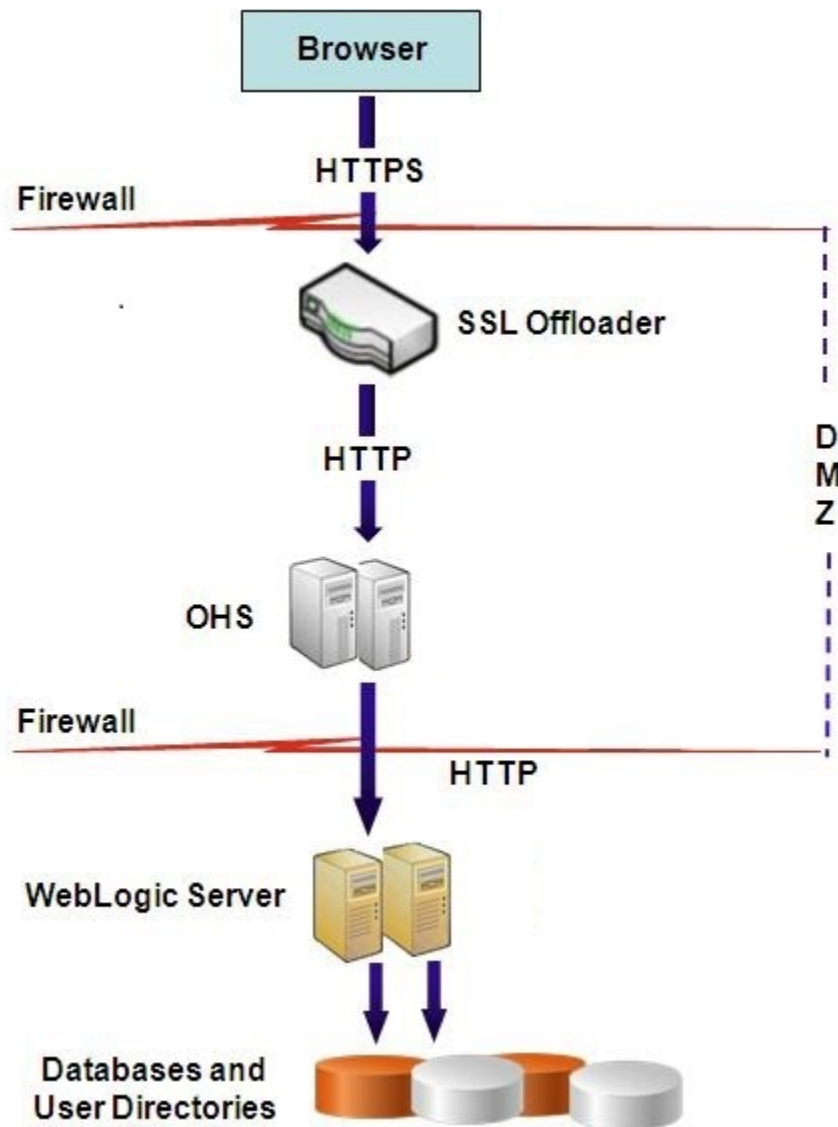
您可以使用 Shared Services Console 設定使用者目錄。您在設定使用者目錄時，必須選取 **SSL 已啟用** 選項；該選項會命令 EPM System 安全性在與使用者目錄通訊時，要使用安全通訊協定。您可以啟用 EPM System 安全性與已啟用 LDAP 的使用者目錄 (例如 Oracle Internet Directory 和 Microsoft Active Directory) 之間連線的 SSL。

請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定使用者目錄」。

在 Web 伺服器終止 SSL

部署架構

在這個案例中，SSL 是用來保護 Oracle Enterprise Performance Management System 用戶端 (例如瀏覽器) 和 Oracle HTTP Server 之間的通訊連結。以下是概念的圖解說明：



假設

這個組態在 Web 伺服器上會使用兩個伺服器別名 (例如 `epm.myCompany.com` 和 `empinternal.myCompany.com`)，其中一個用在 Web 伺服器與瀏覽器之間的外部通訊上，另一個則用於不同 EPM System 伺服器之間的內部通訊。請確保伺服器別名指向機器的 IP 位址，且可以透過 DNS 來解析。

您必須在 Web 伺服器上安裝已簽署憑證，以支援利用瀏覽器 (例如透過 `epm.myCompany.com`) 進行的外部通訊。這個虛擬主機應該要終止 SSL，並將 HTTP 要求轉送至 Oracle HTTP Server。

當 SSL 正在 Oracle HTTP Server (OHS) 或負載平衡器上終止時，您必須：

- 將每個 Logical Web Application 設為負載平衡器或 Oracle HTTP Server 的非 SSL 虛擬主機 (例如，`empinternal.myCompany.com:80`，其中 80 為非 SSL 連接埠)。開啟「組態」畫面，完成下列步驟：
 1. 展開 **Hyperion Foundation** 組態任務。
 2. 選取**設定 Web 應用程式的邏輯位址**。
 3. 指定**主機名稱**、非 SSL 連接埠號碼及 SSL 連接埠號碼。
- 將外部 URL 設為負載平衡器或 Oracle HTTP Server 的啟用 SSL 虛擬主機 (例如，`empexternal.myCompany.com:443`，其中 443 為 SSL 連接埠)。開啟「組態」畫面，完成下列步驟：
 1. 展開 **Hyperion Foundation** 組態任務。
 2. 選取**設定公用設定**。
 3. 選取外部 URL 詳細資料底下的**啟用 SSL 卸載**。
 4. 指定**外部 URL 主機**和**外部 URL 連接埠**。

 **備註：**

使用 **configtool** 重新部署 Web 應用程式或設定 Web 伺服器將會取代 Logical Web Application 和外部 URL 的設定值。

設定 EPM System

EPM System 元件的預設部署支援在 Web 伺服器終止 SSL。您不需要進行其他的工作。

當您在設定 EPM System 時，請確保要把邏輯 Web 應用程式指向專為內部通訊所建立的虛擬主機 (例如 `empinternal.myCompany.com`)。請參閱下列的資訊來源，以便安裝及設定 EPM System：

- *Oracle Enterprise Performance Management System 安裝與組態手冊*
- *Oracle Enterprise Performance Management System 安裝入門*

測試部署

當您完成部署程序之後，請連線到安全的 Oracle Hyperion Enterprise Performance Management Workspace URL，以確認一切都正常運作：

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

例如 `https://epm.myCompany.com:443/workspace/index.jsp`，其中 443 是 SSL 連接埠。

適用於 Essbase 11.1.2.4 的 SSL

簡介

本節說明取代預設憑證的程序，這些憑證用於保護 Oracle Essbase 例項與元件 (例如 MaxL、Oracle Essbase Administration Services 伺服器、Oracle Essbase Studio 伺服器、Oracle Hyperion Provider Services、Oracle Hyperion Foundation Services、Oracle Hyperion Planning、Oracle Hyperion Financial Management 和 Oracle Hyperion Shared Services Registry) 之間的通訊。

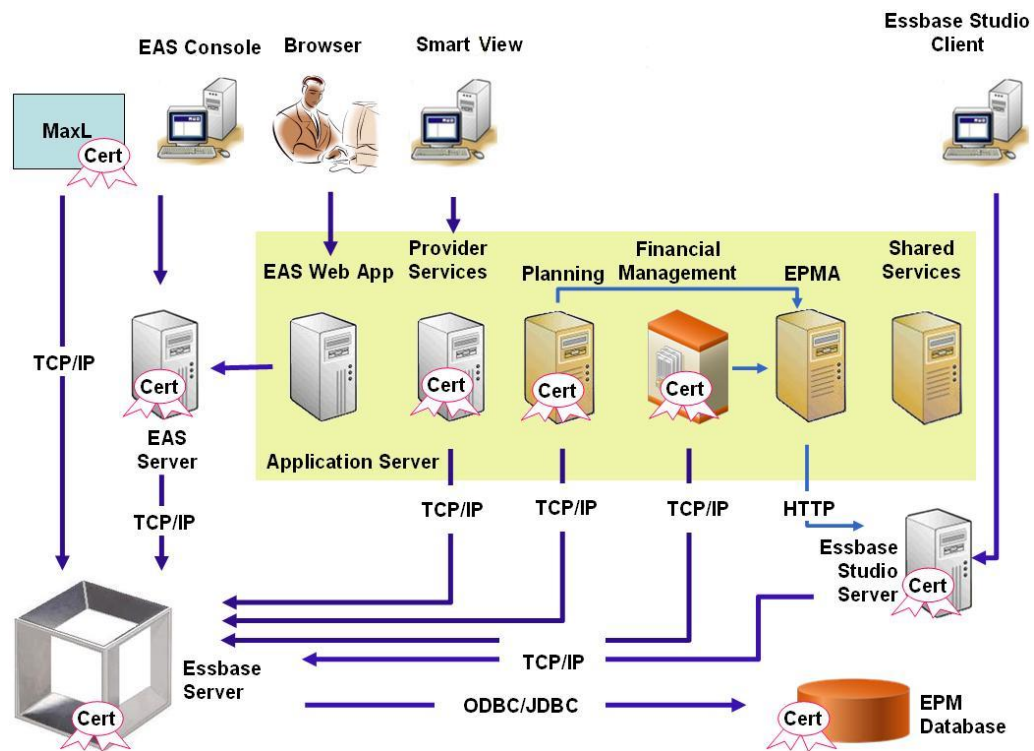
預設部署

您可以把 Essbase 部署成在 SSL 模式，以及在非 SSL 模式中運作。Essbase 代理程式會監聽非安全連接埠；您也可以把它設定成會監聽安全連接埠。所有存取安全連接埠的連線都會被視為 SSL 連線。如果某個用戶端連線到非 SSL 連接埠上的 Essbase 代理程式，該連線就會被視為非 SSL 連線。元件可建立會前往 Essbase 代理程式的並行非 SSL 和 SSL 連線。

您可以控制每個階段作業的 SSL，方法是在您登入時指定安全通訊協定和連接埠。請參閱[建立每一階段作業 SSL 連線](#)。

如果 SSL 已啟用，Essbase 例項內的所有通訊都會受到加密，以確保資料安全無虞。

採用安全模式的預設 Essbase 元件部署，會使用自我簽署憑證來啟用 SSL 通訊，這主要是用來測試的。Oracle 建議您，使用知名第三方 CA 的憑證來啟用實際執行環境中 Essbase 的 SSL 設定。



通常，Oracle Wallet 儲存會搭配使用 Essbase RTC 的用戶端來啟用 SSL 通訊的憑證，而 Java 金鑰存放區儲存會搭配利用 JAPI 來通訊的元件來啟用 SSL 通訊的憑證。如要建立 SSL 通訊，Essbase 用戶端和工具要儲存已簽署 Essbase 伺服器 and 代理程式憑證之 CA 的根憑證。請參閱[必要的憑證和它們的位置](#)。

必要的憑證和它們的位置

Oracle 建議您，使用知名第三方 CA 的憑證來啟用實際執行環境中 Essbase 的 SSL 設定。您可以針對測試，使用預設的自我簽署憑證。

備註：

Essbase 支援使用萬用字元憑證，讓您能使用單一 SSL 憑證來保護多個子網域。使用萬用字元憑證能減少管理時間和成本。

如果您已啟用主機名稱檢查功能，就無法使用萬用字元憑證。

您需要下列憑證：

- 根 CA 憑證。
使用 Essbase RTC 來建立至 Essbase 之連線的元件，需要讓根 CA 憑證儲存在 Oracle Wallet 中。使用 JAPI 來建立連線的元件，需要讓根 CA 憑證儲存在 Java 金鑰存放區中。下方的表格列出必要的憑證和它們的位置。

備註：

如果您要使用來自知名第三方 CA 的憑證，且該 CA 的根憑證已經安裝在 Oracle Wallet 中，您可能就不需要安裝根 CA 憑證。

- Essbase 伺服器和 Essbase 代理程式的已簽署憑證。

表格 2-1 必要的憑證和它們的位置

元件 ¹	金鑰存放區	憑證 ²
MaxL	Oracle Wallet	根 CA 憑證
Administration Services 伺服器	Oracle Wallet	根 CA 憑證
Provider Services	Oracle Wallet	根 CA 憑證
Oracle Enterprise Performance Management System 資料庫	Oracle Wallet	根 CA 憑證
Essbase Studio 伺服器	Java 金鑰存放區	根 CA 憑證
Planning	<ul style="list-style-type: none"> · Oracle Wallet · Java 金鑰存放區 	根 CA 憑證
Financial Management	Java 金鑰存放區	根 CA 憑證
Essbase (伺服器和代理程式) ³	<ul style="list-style-type: none"> · Oracle Wallet · Java 金鑰存放區 	<ul style="list-style-type: none"> · 根 CA 憑證 · Essbase 伺服器和代理程式的已簽署憑證。
Oracle Hyperion Shared Services 儲存庫		

表格 2-1 (續) 必要的憑證和它們的位置

元件 ¹	金鑰存放區	憑證 ²
1 您只需要金鑰存放區的一個例項即可支援使用類似金鑰存放區的多個元件。		
2 您可以讓多個元件使用安裝在金鑰存放區中的單一根憑證。		
3 憑證必須安裝在預設的 Oracle Wallet 中，以及在 Java 金鑰存放區中。		

安裝及部署 Essbase 元件

設定程序會讓您選取安全的代理連接埠 (預設值為 6423)，而您可以在設定 Oracle Essbase 時改變這個連接埠。根據預設，部署程序會安裝必要的自我簽署憑證，以便建立測試用的實用安全部署。

如果 Oracle HTTP Server 已安裝，EPM System Installer 會在裝載 Essbase 例項之機器上的 `ARBOR_PATH` 中，安裝 Oracle Wallet 和自我簽署憑證。在單一主機部署中，所有 Essbase 元件都會共用這個憑證。

針對 Essbase 使用受信任第三方 CA 憑證

建立憑證要求及取得憑證

您將產生憑證要求，為裝載 Oracle Essbase 伺服器及 Essbase 代理程式的伺服器取得憑證。憑證要求包含您辨別名稱 (DN) 特有的加密資訊。您要將憑證提交給簽署授權單位，已取得 SSL 憑證。

您可以使用工具 (例如金鑰工具或 Oracle Wallet Manager) 來建立憑證要求。如需如何建立憑證要求的詳細資訊，請參閱您要使用之工具的相關文件。

如果您要使用金鑰工具，請使用類似下列的命令來建立憑證要求：

```
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
```

取得及安裝根 CA 憑證

根 CA 憑證會驗證要用來支援 SSL 之憑證的有效性。它包含與用於簽署憑證之私密金鑰相符的公開金鑰，以便驗證憑證。您可以從簽署您 SSL 憑證的簽署授權單位取得根 CA 憑證。

請在連線到 Essbase 伺服器或代理程式的用戶端上，安裝簽署 Essbase 伺服器憑證之 CA 的根憑證。請務必將根憑證安裝在適合該用戶端的金鑰存放區中。請參閱[必要的憑證和它們的位置](#)。

備註：

您可以讓多個元件使用安裝在伺服器機器上的單一根憑證。

Oracle Wallet

如需 Oracle Wallet 中 CA 根憑證的元件清單，請參閱[必要的憑證和它們的位置](#)。您可以建立公事包，或是在已安裝預設自我簽署憑證的示範公事包中安裝憑證。

如需建立公事包和匯入根 CA 憑證的詳細程序，請參閱 Oracle Wallet Manager 的相關文件。

Java 金鑰存放區

如需 Java 金鑰存放區中根 CA 憑證的元件清單，請參閱[必要的憑證和它們的位置](#)。您可以將憑證新增到已安裝自我簽署憑證的金鑰存放區中，或是建立金鑰存放區來儲存憑證。

備註：

許多知名第三方 CA 的根憑證，都已經安裝在 Java 金鑰存放區中。

如需詳細的指示，請參閱您要使用之工具的相關文件。如果您要使用金鑰工具，請使用類似下列的命令來匯入根憑證：

```
keytool -import -alias blister_CA -file c:/certs/CA.crt -keypass  
password -trustcacerts -keystore  
C:\Oracle\Middleware\EPMSysstem11R1\Essbase_ssl  
\keystore -storepass password
```

安裝已簽署憑證

您將在裝載 Oracle Essbase 伺服器和 Essbase 代理程式的伺服器上安裝已簽署的 SSL 憑證。使用 Essbase RTC (C API) 來建立至 Essbase 伺服器或代理程式之連線的元件，需要讓根 CA 憑證儲存在擁有根 CA 憑證的 Oracle Wallet 中。使用 JAPI 建立至 Essbase 伺服器或代理程式之連線的元件，需要讓根 CA 憑證和已簽署的 SSL 憑證儲存在 Java 金鑰存放區中。如需詳細的程序，請參閱以下資訊來源：

- Oracle Wallet Manager 的相關文件
- 您用來匯入憑證之工具 (例如金鑰工具) 的相關文件或線上說明

如果您要使用金鑰工具，請使用類似下列的命令來匯入憑證：

```
keytool -import -alias essbase_ssl -file C:/certs/essbase_ssl.crt -keypass  
password -keystore  
C:\Oracle\Middleware\EPMSysstem11R1\Essbase_ssl\keystore -storepass password
```

更新 Essbase 伺服器登錄值

Windows

1. 在命令提示字元中，將目錄變更為 `EPMSysstem11R1\bin`。
2. 執行以下命令以更新 Windows 登錄：

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

請務必將 <Object ID> 取代為 Essbase 伺服器元件 ID，該 ID 會在完成 Essbase 伺服器組態程序之後產生的登錄報表中提供。

Linux

1. 在主控台中，將目錄變更為 `EPM_ORACLE_INSTANCE/epmsystem1/bin`。
2. 執行以下命令以更新登錄：


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

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

請務必將 <Object ID> 取代為 Essbase 伺服器元件 ID，該 ID 會在完成 Essbase 伺服器組態程序之後產生的登錄報表中提供。

更新 Essbase 的 SSL 設定

您可以藉由在 `essbase.cfg` 中指定以下值來自訂 Essbase 伺服器 and 用戶端的 SSL 設定。

- 啟用安全模式的設定
- 啟用清除模式的設定
- 您偏好用來與用戶端通訊的模式 (僅限讓用戶端使用)
- 安全連接埠
- 密碼組
- Oracle Wallet 路徑



備註：

請務必在 `essbase.cfg` 中，新增所有遺漏的必要參數 (具體而言，即 `EnableSecureMode`、`AgentSecurePort`)，並設定其值。

如何更新 `essbase.cfg`：

1. 將含有 Essbase 伺服器憑證的 Oracle Wallet 複製到 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/wallet`。
這是 Essbase 伺服器唯一可接受的 Oracle Wallet 位置。
2. 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg`。
3. 視需要輸入設定。預設的 Essbase 設定是受到默許的。如果您必須變更預設行為，請在 `essbase.cfg` 中新增自訂行為的設定。例如，`EnableClearMode` 是預設強制執行的，它讓 Essbase 伺服器能夠透過未加密管道來通訊。若要關閉 Essbase 伺服器透過未加密管道來通訊的能力，您必須在 `essbase.cfg` 中指定 `EnableClearMode FALSE`。請參閱下列表格。

表格 2-2 Essbase 的 SSL 設定

設定	描述 ¹
EnableClearMode ²	<p>啟用 Essbase 應用程式與 Essbase 代理程式之間的未加密通訊。如果這個特性設定為 FALSE，Essbase 就無法處理非 SSL 的要求。</p> <p>預設值：EnableClearMode TRUE</p> <p>範例：EnableClearMode FALSE</p>
EnableSecureMode	<p>啟用 Essbase 用戶端與 Essbase 代理程式之間的 SSL 加密通訊。這個特性必須設定為 TRUE，才能支援 SSL。</p> <p>預設值：FALSE</p> <p>範例：EnableSecureMode TRUE</p>
SSLCipherSuites	<p>用於 SSL 通訊的密碼組清單 (以慣用順序來排列)。Essbase 代理程式會使用其中一個密碼組來進行 SSL 通訊。當代理程式要選擇密碼組時，會優先選擇清單中的第一個密碼組。</p> <p>預設值：SSL_RSA_WITH_RC4_128_MD5</p> <p>範例：SSLCipherSuites SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</p>
APSRESOLVER	<p>Oracle Hyperion Provider Services URL。如果您使用多部 Provider Services 伺服器，請使用分號分隔每個 URL。</p> <p>範例：APSRESOLVER https://exampleAPShost1:PORT/aps;https://exampleAPShost2:PORT/aps</p>
AgentSecurePort	<p>代理程式會監聽的安全連接埠。</p> <p>預設值：6423</p> <p>範例：AgentSecurePort 16001</p>
WalletPath	<p>儲存根 CA 憑證和已簽署憑證之 Oracle Wallet 的所在位置 (少於 1,024 個字元)。</p> <p>預設值：ARBORPATH/bin/wallet</p> <p>範例：WalletPath/usr/local/wallet</p>
ClientPreferredMode ³	<p>用戶端階段作業的模式 (安全或清除)。如果這個特性設定為「Secure」，所有作業階段都會使用 SSL 模式。如果這個特性設定為「Clear」，系統會根據用戶端登入要求是否包含安全傳輸關鍵字來決定傳輸方式。請參閱 建立每一階段作業 SSL 連線。</p> <p>預設值：CLEAR</p> <p>範例：ClientPreferredMode SECURE</p>

¹ 如果 essbase.cfg 中沒有這些特性，系統會強制執行預設值。

² 如果 EnableClearMode 和 EnableSecureMode 設定為 FALSE，Essbase 將無法運作。

³ 用戶端會使用這個設定來決定，要建立與 Essbase 的安全連線，還是非安全連線。

4. 儲存並關閉 essbase.cfg。

更新分散式 essbase 節點以進行 SSL



備註：

本節僅適用於 Essbase 的分散式部署

請確保包含根 CA 憑證和簽署憑證的公事包資料夾 (例如，WalletPath/usr/local/wallet) 位於每個分散式節點上的所需位置。

1. 將公事包資料夾複製到每個分散式節點中的以下位置：
 - `EPM_ORACLE_HOME/common/EssbaseRTC/11.1.2.0/bin`
 - `EPM_ORACLE_HOME/common/EssbaseRTC-64/11.1.2.0/bin`
2. 將公事包資料夾複製到每個分散式節點中的以下位置 (若存在)：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer-32/bin`
 - `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin`
3. 將 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg` 複製到每個分散式節點上的以下位置：
 - `EPM_ORACLE_HOME/common/EssbaseRTC/11.1.2.0/bin`
 - `EPM_ORACLE_HOME/common/EssbaseRTC-64/11.1.2.0/bin`
4. 將 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg` 複製到每個分散式節點上的以下位置 (若存在)：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseServer-32/bin`
 - `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin`
5. 將公事包資料夾複製到每個分散式節點上的以下 Essbase 用戶端安裝位置：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient-32/bin`
6. 將 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/essbase.cfg` 複製到每個分散式節點上的以下 Essbase 用戶端安裝位置：
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient/bin`
 - `EPM_ORACLE_HOME/products/Essbase/EssbaseClient-32/bin`
7. 將這些特性新增至 `essbase.properties` 檔案：
 - `essbase.ssleverywhere=true`
 - `olap.server.ssl.alwaysSecure=true`
 - `APSRESOLVER=http[s]://host:httpsPort/aps`
請務必將此值取代為正確的 URL。

您必須在每個分散式節點中的以下位置 (若存在) 更新 `essbase.properties` 檔案：

- `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. 將 `EPM_ORACLE_HOME/products/Essbase/aps/bin/essbase.properties` 複製到每個分散式節點上的 `EPM_ORACLE_HOME/products/Essbase/eas` 目錄 (如果可用)。

9. **僅適用於 Oracle Hyperion Planning**：將以下三個特性新增至 `essbase.properties` 檔案：

- `essbase.ssleverywhere=true`
- `olap.server.ssl.alwaysSecure=true`
- `APSRESOLVER=APS_URL`
將 `APS_URL` 取代為 **Provider Services URL**。如果您使用多部 **Provider Services** 伺服器，請使用分號分隔每個 URL。例如，`https://exampleAPShost1:PORT/aps;https://exampleAPShost2:PORT/aps`。

您必須在每個分散式節點中的以下位置更新 `essbase.properties` 檔案：

- `EPM_ORACLE_HOME/products/Planning/config/essbase.properties`
- `EPM_ORACLE_HOME/products/Planning/lib/essbase.properties`

10. **僅適用於 Oracle Hyperion Financial Reporting**：將以下三個特性新增至 `EPM_ORACLE_HOME/products/financialreporting/bin/EssbaseJAPI/bin/essbase.properties` 檔案：

- `essbase.ssleverywhere=true`
- `olap.server.ssl.alwaysSecure=true`
- `APSRESOLVER=APS_URL`
將 `APS_URL` 取代為 **Provider Services URL**。如果您使用多部 **Provider Services** 伺服器，請使用分號分隔每個 URL。例如，`https://exampleAPShost1:PORT/aps;https://exampleAPShost2:PORT/aps`。

 **備註：**

在完整的 SSL 環境中，**Financial Reporting** 需要有 **Essbase** 叢集名稱，才能建立連線。若系統使用主機名稱來連線，連線就會失敗。

11. a. 設定環境變數：

- **Windows**：建立一個名為 `API_DISABLE_PEER_VERIFICATION` 的新系統變數，並將其值設為 1。
- **Linux**：在 `setCustomParamsPlanning.sh` 中新增指示詞 `API_DISABLE_PEER_VERIFICATION=1`。

b. 在 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/setEssbaseenv.bat` 或 `EPM_ORACLE_INSTANCE/EssbaseServer/essbaseserver1/bin/setEssbaseenv.sh` 中新增指示詞 `API_DISABLE_PEER_VERIFICATION=1`。

設定環境變數：

自訂 JAPI 用戶端的 SSL 特定

在需要 JAPI 之 Essbase 元件的預設特性中，有許多會遭到預先定義。您可以覆寫預設特性，方法是將該特性加入 `essbase.properties`。

備註：

下表所識別的 SSL 特性，只有少數在 `essbase.properties` 中被外部化。我們建議您新增沒有遭到外部化的特性。

如何更新 JAPI 用戶端的 SSL 特性：

1. 使用文字編輯器開啟 `EPM_ORACLE_HOME/common/EssbaseJavaAPI/11.1.2.0/bin/essbase.properties`。
2. 視需要更新特性。如需可自訂的 JAPI 用戶端特性的說明，請參閱下表。如果您想用的某個特性不在 `essbase.properties` 中，請新增該特性。

表格 2-3 JAPI 用戶端的預設 SSL 特性

特性	說明
<code>olap.server.ssl.alwaysSecure</code>	設定用戶端在面對所有 Essbase 例項時使用的模式。如果您將這個特性值變更為 <code>true</code> ，即可強制採用 SSL 模式。 預設值： <code>false</code>
<code>olap.server.ssl.securityHandler</code>	處理通訊協定的套裝程式名稱您可以變更此值來指定另一個處理程式。 預設值： <code>java.protocol.handler.pkgs</code>
<code>olap.server.ssl.securityProvider</code>	Oracle 使用 Sun SSL 通訊協定實作。您可以變更此值來指定另一個供應商。 預設值： <code>com.sun.net.ssl.internal.www.protocol</code>
<code>olap.server.ssl.supportedCiphers</code>	用於啟用安全通訊之其他密碼組的逗號分隔清單。您只能指定 Essbase 支援的密碼組。 範例： <code>SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</code>

表格 2-3 (續) JAPI 用戶端的預設 SSL 特性

特性	說明
<code>olap.server.ssl.trustManagerClass</code>	<p>用於驗證 SSL 憑證的 TrustManager 類別，方法是驗證簽章和查看憑證過期日期。根據預設，這個特性並沒有設定來強制執行所有的驗證檢查。</p> <p>如果您不想強制執行驗證檢查，請將該參數值設定成 <code>com.essbase.services.olap.security.EssDefaultTrustManager</code>；這是能讓所有驗證檢查發生的預設 TrustManager 類別。</p> <p>若要實作自訂的 TrustManager，請指定實作 <code>javax.net.ssl.X509TrustManager</code> 介面的 TrustManager 類別的完整類別名稱。</p> <p>範例： <code>com.essbase.services.olap.security.EssDefaultTrustManager</code></p>

3. 儲存並關閉 `essbase.properties`。
4. 重新啟動所有 Essbase 元件。

建立每一階段作業 SSL 連線

Oracle Essbase 元件 (例如 MaxL) 可以控制階段作業層級的 SSL，方法是把 `secure` 當做傳輸關鍵字以連線至 Essbase 代理程式。例如，您可以在 MaxL 主控台執行下列其中一個命令，以建立 MaxL 與 Essbase 代理程式之間的安全連線：

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

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

系統會優先採用每一階段作業控制，而非您在 `essbase.cfg` 中指定的組態設定。如果您沒有指定運輸關鍵字，Essbase 用戶端會使用您為 `ClientPreferredMode` 設定的值，來決定是否要將該用戶端與 Essbase 之間的安全連線初始化。如果 `ClientPreferredMode` 沒有設定成「Secure」，通訊就會透過不安全的管道來進行。

適用於 Essbase 21c 的 SSL

簡介

本節說明取代預設憑證的程序，這些憑證用於保護 Oracle Essbase 例項與元件 (例如 MaxL、Oracle Essbase Administration Services 伺服器、Oracle Hyperion Provider Services、Oracle Hyperion Foundation Services、Oracle Hyperion Planning、Oracle Hyperion Financial Management 和 Oracle Hyperion Shared Services Registry) 之間的通訊。

 **備註：**

Essbase Administration Services (EAS) Lite 不使用利用 EPM 組態程式設定的 HTTP 伺服器 SSL 連接埠 (例如，443)。easconsole.jnlp 檔案中的安全 URL 預設為非 SSL 連接埠 (80)。

因應措施：以更新過的安全 URL 取代 easconsole.jnlp 中所識別安全 URL 中的預設非 SSL 連接埠：

預設安全 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

請參閱 My Oracle Support (MOS) 文章 - [文件 ID 1926558.1 - EAS Web 主控台的 easconsole.jnlp 不包含 SSL 連接埠](#) 取得更多詳細資訊。

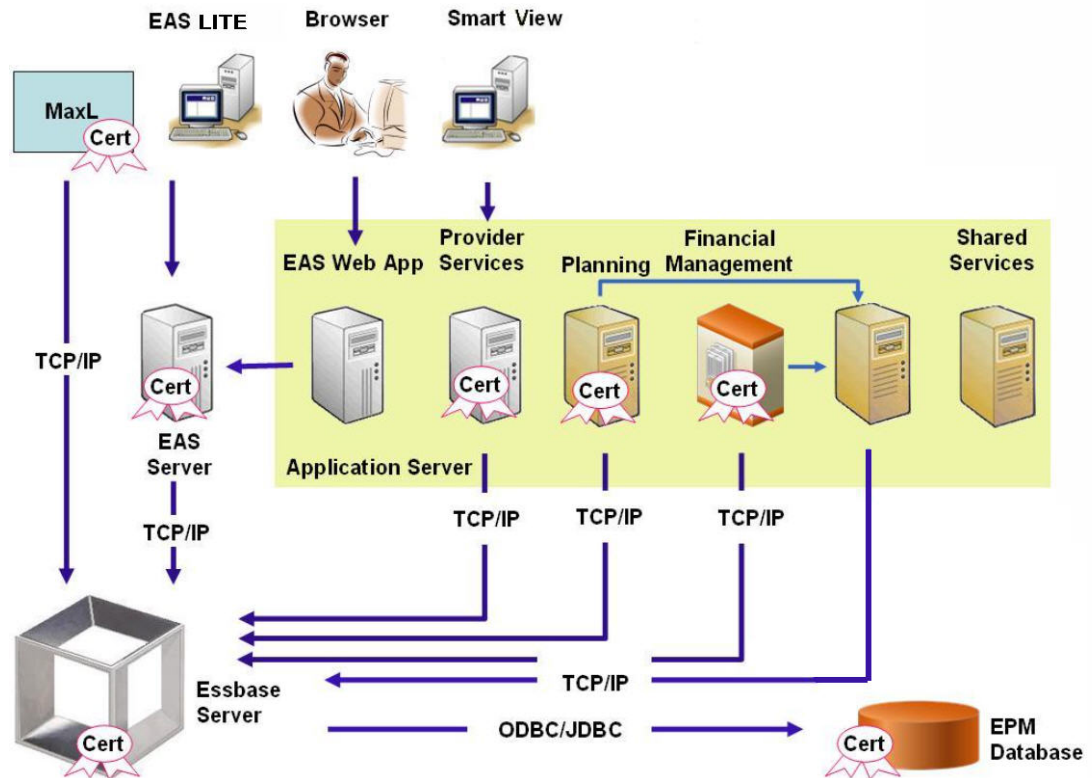
預設部署

您可以把 Essbase 部署成在 SSL 模式，以及在非 SSL 模式中運作。Essbase 代理程式會監聽非安全連接埠；您也可以把它設定成會監聽安全連結埠。所有存取安全連接埠的連線都會被視為 SSL 連線。如果某個用戶端連線到非 SSL 連接埠上的 Essbase 代理程式，該連線就會被視為非 SSL 連線。元件可建立會前往 Essbase 代理程式的並行非 SSL 和 SSL 連線。

您可以控制每個階段作業的 SSL，方法是在您登入時指定安全通訊協定和連接埠。請參閱 [建立每一階段作業 SSL 連線](#)。

如果 SSL 已啟用，Essbase 例項內的所有通訊都會受到加密，以確保資料安全無虞。

採用安全模式的預設 Essbase 元件部署，會使用自我簽署憑證來啟用 SSL 通訊，這主要是用來測試的。Oracle 建議您，使用知名第三方 CA 的憑證來啟用實際執行環境中 Essbase 的 SSL 設定。



通常，Oracle Wallet 儲存會搭配使用 Essbase RTC 的用戶端來啟用 SSL 通訊的憑證，而 Java 金鑰存放區儲存會搭配利用 JAPI 來通訊的元件來啟用 SSL 通訊的憑證。如要建立 SSL 通訊，Essbase 用戶端和工具要儲存已簽署 Essbase 伺服器 and 代理程式憑證之 CA 的根憑證。

必要的憑證和它們的位置

Oracle 建議您，使用知名第三方 CA 的憑證來啟用實際執行環境中 Essbase 的 SSL 設定。您可以針對測試，使用預設的自我簽署憑證。

備註：

Essbase 支援使用萬用字元憑證，讓您使用單一 SSL 憑證來保護多個子網域。使用萬用字元憑證能減少管理時間和成本。

如果您已啟用主機名稱檢查功能，就無法使用萬用字元憑證。

您需要下列憑證：

- 根 CA 憑證。
使用 Essbase RTC 來建立至 Essbase 之連線的元件，需要讓根 CA 憑證儲存在 Oracle Wallet 中。使用 JAPI 來建立連線的元件，需要讓根 CA 憑證儲存在 Java 金鑰存放區中。下方的表格列出必要的憑證和它們的位置。

 **備註：**

如果您要使用來自知名第三方 CA 的憑證，且該 CA 的根憑證已經安裝在 Oracle Wallet 中，您可能就不需要安裝根 CA 憑證。

- Essbase 伺服器 and Essbase 代理程式的已簽署憑證。

表格 2-4 必要的憑證和它們的位置

元件 ¹	金鑰存放區	憑證 ²
MaxL	Oracle Wallet	根 CA 憑證
Administration Services 伺服器	Oracle Wallet	根 CA 憑證
Provider Services	Oracle Wallet	根 CA 憑證
Oracle Enterprise Performance Management System 資料庫	Oracle Wallet	根 CA 憑證
Planning	<ul style="list-style-type: none"> · Oracle Wallet · Java 金鑰存放區 	根 CA 憑證
Financial Management	Java 金鑰存放區	根 CA 憑證
Essbase (伺服器和代理程式) ³	<ul style="list-style-type: none"> · Oracle Wallet · Java 金鑰存放區 	<ul style="list-style-type: none"> · 根 CA 憑證 · Essbase 伺服器和代理程式的已簽署憑證。

Oracle Hyperion Shared Services
儲存庫

- ¹ 您只需要金鑰存放區的一個例項即可支援使用類似金鑰存放區的多個元件。
- ² 您可以讓多個元件使用安裝在金鑰存放區中的單一根憑證。
- ³ 憑證必須安裝在預設的 Oracle Wallet 中，以及在 Java 金鑰存放區中。

安裝及部署 Essbase 元件

設定程序會讓您選取安全的代理連接埠 (預設值為 6423)，而您可以在設定 Oracle Essbase 時改變這個連接埠。根據預設，部署程序會安裝必要的自我簽署憑證，以便建立測試用的實用安全部署。

如果 Oracle HTTP Server 已安裝，EPM System Installer 會在裝載 Essbase 例項之機器上的 `ARBOR_PATH` 中，安裝 Oracle Wallet 和自我簽署憑證。在單一主機部署中，所有 Essbase 元件都會共用這個憑證。

針對 Essbase 使用受信任第三方 CA 憑證

建立憑證要求及取得憑證

您將產生憑證要求，為裝載 Oracle Essbase 伺服器和 Essbase 代理程式的伺服器取得憑證。憑證要求包含您伺服器一般名稱 (CN=) 特有的加密資訊。您要將憑證提交給簽署授權單位，已取得 SSL 憑證。

您可以使用工具 (例如金鑰工具或 Oracle Wallet Manager) 來建立憑證要求。如需如何建立憑證要求的詳細資訊，請參閱您要使用之工具的相關文件。

使用金鑰工具的範例：

建立 Java 金鑰存放區 (JKS) 並產生私密金鑰：

```
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
```

產生憑證要求：

```
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
```

匯出私密金鑰 (需有 openssl 公用程式才能執行以下步驟)：

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 (憑證授權單位) 簽署新產生的憑證要求，並將它貼入以下檔案：
C:\oracle\Middleware\EPMSysstem11R1\ssl\essbase.cer。

取得及安裝根 CA 憑證

根 CA 憑證會驗證要用來支援 SSL 之憑證的有效性。它包含與用於簽署憑證之私密金鑰相符的公開金鑰，以便驗證憑證。您可以從簽署您 SSL 憑證的簽署授權單位取得根 CA 憑證。

請在連線到 Essbase 伺服器或代理程式的用戶端上，安裝簽署 Essbase 伺服器憑證之 CA 的根憑證。請務必要將根憑證安裝在適合該用戶端的金鑰存放區中。請參閱[必要的憑證和它們的位置](#)。

備註：

您可以讓多個元件使用安裝在伺服器機器上的單一根憑證。

安裝 CA 簽署的憑證

針對安裝 CA 簽署的憑證，請參閱以下連結：

- [設定 Essbase 的 Weblogic TLS 連線](#)
- [更新 TLS 憑證](#)

更新以下路徑底下的 tls.properties 檔案

```
%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.
```

執行以下程式碼，以新憑證更新 Essbase 伺服器：

```
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 的 SSL 設定

您可以藉由在 `essbase.cfg` 中指定以下值來自訂 Essbase 伺服器和用戶端的 SSL 設定。

- 啟用安全模式的設定
- 啟用清除模式的設定
- 您偏好用來與用戶端通訊的模式 (僅限讓用戶端使用)
- 安全連接埠
- 密碼組
- Oracle Wallet 路徑

備註：

請務必在 `essbase.cfg` 中，新增所有遺漏的必要參數 (具體而言，即 `EnableSecureMode`、`AgentSecurePort`)，並設定其值。

若要更新以下路徑底下的 `essbase.cfg`：

```
ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\essbase
```

1. 視需要輸入設定。預設的 Essbase 設定是受到默許的。如果您必須變更預設行為，請在 `essbase.cfg` 中新增自訂行為的設定。例如，`EnableClearMode` 是預設強制執行的，它讓 Essbase 伺服器能夠透過未加密管道來通訊。若要關閉 Essbase 伺服器透過未加密管道來通訊的能力，您必須在 `essbase.cfg` 中指定 `EnableClearMode FALSE`。請參閱下列表格：

表格 2-5 Essbase 的 SSL 設定

設定	描述 ¹
EnableClearMode ²	<p>啟用 Essbase 應用程式與 Essbase 代理程式之間的未加密通訊。如果這個特性設定為 FALSE，Essbase 就無法處理非 SSL 的要求。</p> <p>預設值：EnableClearMode TRUE</p> <p>範例：EnableClearMode FALSE</p>
EnableSecureMode	<p>啟用 Essbase 用戶端與 Essbase 代理程式之間的 SSL 加密通訊。這個特性必須設定為 TRUE，才能支援 SSL。</p> <p>預設值：FALSE</p> <p>範例：EnableSecureMode TRUE</p>
SSLCipherSuites	<p>用於 SSL 通訊的密碼組清單 (以慣用順序來排列)。Essbase 代理程式會使用其中一個密碼組來進行 SSL 通訊。當代理程式要選擇密碼組時，會優先選擇清單中的第一個密碼組。</p> <p>預設值：SSL_RSA_WITH_RC4_128_MD5</p> <p>範例：SSLCipherSuites SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</p>
APSRESOLVER	<p>Oracle Hyperion Provider Services URL。如果您使用多部 Provider Services 伺服器，請使用分號分隔每個 URL。</p> <p>範例：https://exampleAPShost1:PORT/essbase;https://exampleAPShost2:PORT/essbase</p>
AgentSecurePort	<p>代理程式會監聽的安全連接埠。</p> <p>預設值：6423</p> <p>範例：AgentSecurePort 16001</p>
WalletPath	<p>儲存根 CA 憑證和已簽署憑證之 Oracle Wallet 的所在位置 (少於 1,024 個字元)。</p> <p>預設值：ARBORPATH/bin/wallet</p> <p>範例：WalletPath/usr/local/wallet</p>
ClientPreferredMode ³	<p>用戶端階段作業的模式 (安全或清除)。如果這個特性設定為「Secure」，所有作業階段都會使用 SSL 模式。如果這個特性設定為「Clear」，系統會根據用戶端登入要求是否包含安全傳輸關鍵字來決定傳輸方式。請參閱 建立每一階段作業 SSL 連線。</p> <p>預設值：CLEAR</p> <p>範例：ClientPreferredMode SECURE</p>

- ¹ 如果 essbase.cfg 中沒有這些特性，系統會強制執行預設值。
- ² 如果 EnableClearMode 和 EnableSecureMode 設定為 FALSE，Essbase 將無法運作。
- ³ 用戶端會使用這個設定來決定，要建立與 Essbase 的安全連線，還是非安全連線。

2. 儲存並關閉 essbase.cfg。

更新分散式 essbase 節點以進行 SSL



備註：

本節僅適用於 Essbase 的分散式部署

請確保包含根 CA 憑證和簽署憑證的公事包資料夾 (例如，WalletPath/usr/local/wallet) 位於每個分散式節點上的所需位置。

1. 使用 TLS 工具匯入所有新的 CA 憑證。

如需更多資訊，請參閱下列連結：

- 設定 Essbase 的 Weblogic TLS 連線
- 更新 TLS 憑證

2. 前往來源位置：ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\essbase 並修改 essbase.properties 檔案中的以下特性：

- essbase.ssleverywhere=true
- olap.server.ssl.alwaysSecure=true
- APSRESOLVER=APS_URL
將 APS_URL 取代為 Provider Services URL。如果您使用多部 Provider Services 伺服器，請使用分號分隔每個 URL。

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

3. 將 Wallet 資料夾、Walletssl 資料夾、essbase.cfg 檔案和 essbase.properties 檔案複製到以下目標路徑。

表格 2-6 目標路徑

目標路徑	Wallet	Walletssl	essbase.cfg	essbase.properties
EPM_ORACLE_HOME\common\EssbaseRTC-21c\11.1.2.0\bin	是	是	是	是
EPM_ORACLE_HOME\common\EssbaseJavaAPI-21c\11.1.2.0\bin	是	是	是	是
ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\aps	是	是	是	是
ESSBASE_DOMAIN_HOME\config\fmwconfig\essconfig\essbase	是	是	是	是
MIDDLEWARE_HOME\essbase\products\Essbase\template_files\essbase	是	是	是	是
MIDDLEWARE_HOME\essbase\products\Essbase\EssbaseServer\bin	是	是	是	是
MIDDLEWARE_HOME\essbase\products\Essbase\aps\bin	是	是	是	是
MIDDLEWARE_HOME\essbase\products\Essbase\ears	是	是	是	是
MIDDLEWARE_HOME\essbase\common\EssbaseJavaAPI\bin	是	是	是	是

表格 2-6 (續) 目標路徑

目標路徑	Walle t	Walle tssl	essb ase.c fg	essbas e. properti es
僅限 Oracle Hyperion Financial Reporting EPM_ORACLE_HOME/products/ financialreporting/bin/EssbaseJAPI/bin/ 注意： 在完整的 SSL 環境中，Financial Reporting 需要 有 Essbase 叢集名稱，才能建立連線。若系統使用主機 名稱來連線，連線就會失敗。	是	是	是	是
僅限 Oracle Hyperion Planning EPM_ORACLE_HOME/products/Planning/config/ EPM_ORACLE_HOME/products/Planning/lib/	是	是	是	是

4. 設定環境變數：

- **Windows：**建立一個名為 API_DISABLE_PEER_VERIFICATION 的新系統變數，並將其值設為 1。
- **Linux：**在 setCustomParamsPlanning.sh 中新增指示詞
API_DISABLE_PEER_VERIFICATION=1。

自訂 JAPI 用戶端的 SSL 特定

在需要 JAPI 之 Essbase 元件的預設特性中，有許多會遭到預先定義。您可以覆寫預設特性，方法是將該特性加入 essbase.properties。

 **備註：**

下表所識別的 SSL 特性，只有少數在 essbase.properties 中被外部化。我們建議您新增沒有遭到外部化的特性。

如何更新 JAPI 用戶端的 SSL 特性：

1. 使用文字編輯器開啟 EPM_ORACLE_HOME/common/EssbaseJavaAPI-21c/11.2.0/bin/essbase.properties。
2. 視需要更新特性。如需可自訂的 JAPI 用戶端特性的說明，請參閱下表。如果您想用的某個特性不在 essbase.properties 中，請新增該特性。

表格 2-7 JAPI 用戶端的預設 SSL 特性

特性	說明
olap.server.ssl.alwaysSecure	設定用戶端在面對所有 Essbase 例項時使用的模式。如果您將這個特性值變更為 true，即可強制採用 SSL 模式。 預設值： false
olap.server.ssl.securityHandler	處理通訊協定的套裝程式名稱您可以變更此值來指定另一個處理程式。 預設值： java.protocol.handler.pkgs

表格 2-7 (續) JAPI 用戶端的預設 SSL 特性

特性	說明
<code>olap.server.ssl.securityProvider</code>	Oracle 使用 Sun SSL 通訊協定實作。您可以變更此值來指定另一個供應商。 預設值： <code>com.sun.net.ssl.internal.www.protocol</code>
<code>olap.server.ssl.supportedCiphers</code>	用於啟用安全通訊之其他密碼組的逗號分隔清單。您只能指定 Essbase 支援的密碼組。 範例： <code>SSL_RSA_WITH_AES_256_CBC_SHA256,SSL_RSA_WITH_AES_256_GCM_SHA384</code>
<code>olap.server.ssl.trustManagerClass</code>	用於驗證 SSL 憑證的 TrustManager 類別，方法是驗證簽章和查看憑證過期日期。 根據預設，這個特性並沒有設定來強制執行所有的驗證檢查。如果您不想強制執行驗證檢查，請將該參數值設定成 <code>com.essbase.services.olap.security.EssDefaultTrustManager</code> ；這是能讓所有驗證檢查發生的預設 TrustManager 類別。 若要實作自訂的 TrustManager ，請指定實作 <code>javax.net.ssl.X509TrustManager</code> 介面的 TrustManager 類別的完整類別名稱。 範例： <code>com.essbase.services.olap.security.EssDefaultTrustManager</code>

3. 儲存並關閉 `essbase.properties`。
4. 重新啟動所有 Essbase 元件。

建立每一階段作業 SSL 連線

Oracle Essbase 元件 (例如 MaxL) 可以控制階段作業層級的 SSL，方法是把 `secure` 當做傳輸關鍵字以連線至 Essbase 代理程式。例如，您可以在 MaxL 主控台執行下列其中一個命令，以便建立 MaxL 與 Essbase 代理程式之間的安全連線：

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

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

系統會優先採用每一階段作業控制，而非您在 `essbase.cfg` 中指定的組態設定。如果您沒有指定運輸關鍵字，Essbase 用戶端會使用您為 `ClientPreferredMode` 設定的值，來決定是否要將該用戶端與 Essbase 之間的安全連線初始化。如果 `ClientPreferredMode` 沒有設定成「Secure」，通訊就會透過不安全的管道來進行。

3

啟用安全性代理程式的 SSO

另請參閱：

- [受支援的 SSO 方法](#)
- [來自 Oracle Access Manager 的單一登入](#)
- [OracleAS Single Sign-on](#)
- [保護 EPM System 產品進行 SSO](#)
- [搭配識別管理產品的標頭型 SSO](#)
- [針對搭配 Oracle Identity Cloud Services 的標頭型 SSO 設定 EPM System](#)
- [SiteMinder SSO](#)
- [Kerberos 單一登入](#)
- [設定 EPM System 進行 SSO](#)
- [Smart View 的單一登入選項](#)

受支援的 SSO 方法

SSO 需要 Web 識別管理解決方案將已通過驗證之使用者的登入名稱傳遞給 Oracle Enterprise Performance Management System 產品。您可以使用下列的標準 EPM System 方法，將 EPM System 與商業和自訂的 Web 型 SSO 解決方案整合。

- [HTTP 標頭](#)
- [自訂登入類別](#)
- [HTTP 授權標頭](#)
- [從 HTTP 要求中取得遠端使用者](#)
- [搭配識別管理產品的標頭型驗證](#)

▲ 注意：

如果您的組織會以利用標頭傳送使用者識別的方式來進行識別傳播，Oracle 建議您，在 Web 伺服器與應用程式伺服器之間採用戶端憑證驗證 (雙向 SSL) 來作為安全措施。

HTTP 標頭

如果您要把 Oracle 單一登入 (OSSO)、SiteMinder 或 Oracle Access Manager 作為 Web 識別管理解決方案，EPM System 安全性會自動選取自訂 HTTP 標頭，以便將已驗證使用者的登入名稱傳遞給 EPM System 元件。

EPM System 產品使用者的登入名稱，取決於您在 Oracle Hyperion Shared Services 中設定使用者目錄時所指定的 Login Attribute。如需 Login Attribute 的簡短說明，請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。

HTTP 標頭必須包含設為 Login Attribute 的屬性值。例如，若 uid 為 Login Attribute 值，則 HTTP 標頭必須包含 uid 屬性的值。

如需如何定義及發出自訂 HTTP 標頭的詳細資訊，請參閱您 Web 識別管理解決方案的文件。

EPM System 安全性會剖析 HTTP 標頭，並根據您在 Shared Services 中設定的使用者目錄，驗證該標頭隨附的登入名稱是否有效。

自訂登入類別

當使用者登入時，Web 識別管理解決方案會根據目錄伺服器驗證使用者，並採用 SSO 機制來封裝已驗證使用者的認證，以便啟用下游系統的 SSO。若 Web 識別管理解決方案使用了 EPM System 產品不支援的機制，或是 SSO 機制未提供 Login Attribute 的值，請使用自訂登入類別來衍生 Login Attribute 的值，並將該值傳遞至 EPM System 產品。

使用自訂登入類別，會讓 EPM System 與使用 X509 憑證式驗證的安全性代理程式整合。若要使用這個驗證機制時，您必須實作標準 Shared Services API 來定義 EPM System 產品與 Web 識別管理解決方案之間的 SSO 介面。自訂登入類別必須將 Login Attribute 的值傳遞至 EPM System 產品。如需 Login Attribute 的簡短說明，請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。如需範例程式碼和實作步驟，請參閱 [實作自訂登入類別](#)。

如要使用自訂登入類別 (預設名稱為 `com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl`)，類別路徑中必須要有 `com.hyperion.css.CSSSecurityAgentIF` 介面的實作。`CSSSecurityAgentIF` 會定義擷取使用者名稱與密碼的 `Getter` 方法 (選用)。介面若傳回 `null` 密碼，安全性驗證會將提供者視為信任的提供者，並確認已設定之提供者中確實存在該使用者。若介面傳回非空值的密碼，EPM System 會嘗試使用此實作所傳回的使用者名稱與密碼來驗證要求。

`CSSSecurityAgentIF` 包含兩種方法：`getUserName` 與 `getPassword`。

`getUserName` 方法

此方法會傳回使用者名稱進行驗證。

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

`req` 參數會指定內含可用於判斷使用者名稱之資訊的 HTTP 要求。不使用 `res` 參數 (回溯相容性的預設值)。

getPassword 方法

此方法會傳回純文字密碼進行驗證。密碼擷取為選用性。

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

req 參數會指定內含可用於判斷密碼之資訊的 HTTP 要求。不使用 res 參數 (回溯相容性的預設值)。

HTTP 授權標頭

EPM System 安全性支援使用 HTTP 授權標頭，將 Login Attribute 的值從 Web 識別管理解決方案傳遞至 EPM System 產品。EPM System 產品會剖析授權標頭來擷取使用者的登入名稱。

從 HTTP 要求中取得遠端使用者

EPM System 安全性支援使用 HTTP 要求，將 Login Attribute 的值從 Web 識別管理解決方案傳遞至 EPM System 產品。如果 Web 識別管理解決方案會傳遞包含 Login Attribute 值的 HTTP 要求，且該要求使用 setRemoteUser 函數，請使用這個 SSO 方法。

搭配識別管理產品的標頭型驗證

EPM System 支援任何識別管理產品 (例如 Oracle Identity Cloud Services、Microsoft Azure AD、Okta)，這些產品皆支援標頭型驗證。概念工作流程如下：

- 閘道應用程式可作為反向代理主機，藉由限制未驗證的網路存取來保護 EPM System 元件。
- 閘道應用程式會攔截對 EPM System 元件所發出的 HTTP 要求，並確保識別管理產品在將要求轉送至 EPM System 元件之前先驗證使用者。
- 將要求轉送至 EPM System 元件時，閘道應用程式會透過 HTTP 標頭要求，將已驗證的使用者識別傳播至 EPM System 元件。

若要支援此驗證案例，EPM System 應設定為與透過 HTTP 標頭要求傳播的已驗證使用者識別搭配使用。

來自 Oracle Access Manager 的單一登入

Oracle Enterprise Performance Management System 會藉由接受包含登入屬性值的自訂 HTTP 標頭 (預設名稱為 HYPLOGIN)，與 Oracle Access Manager 整合。而該登入屬性是您在 Oracle Hyperion Shared Services 中設定使用者目錄時所設定的。如需 Login Attribute 的簡短說明，請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。

您可以為提供登入屬性至 EPM System 的標頭設定任何名稱。您在 Oracle Access Manager 針對 SSO 設定 Shared Services 時，就會使用這個標頭名稱。

EPM System 會使用登入屬性值，根據已設定的使用者目錄 (在本案例中，就是 Oracle Access Manager 用來驗證使用者的使用者目錄) 來驗證使用者，然後產生可啟用整個 EPM System 之 SSO 的 EPM SSO 憑證。您必須勾選原生目錄中提供使用者資訊的選項，才能授權使用者存取 EPM System 資源。

 **備註：**

Oracle Essbase Administration Services 主控台 (豐富型用戶端) 並不支援來自 Oracle Access Manager 的 SSO。

如需如何設定 Oracle Access Manager，以及如何執行設定 HTTP 標頭和原則網域等工作的相關資訊，請參閱 Oracle Access Manager 文件。本手冊假設，您已經在運作中的 Oracle Access Manager 部署上完成下列工作：

- 已針對 EPM System 元件設定必要的原則網域
- 已設定 HTTP 標頭，以便將登入屬性值傳遞到 EPM System
- 已保護列在**要保護的資源**中的 EPM System 資源。存取受保護資源的要求，會受到 Oracle Access Manager 的查問。
- 已取消保護列在**要取消保護的資源**中的 EPM System 資源。存取不受保護之資源的要求，不會受到 Oracle Access Manager 的查問。

如何針對 Oracle Access Manager 的 SSO 設定 EPM System：

1. 將 Oracle Access Manager 用來驗證使用者的目錄，新增為 EPM System 中的外部使用者目錄。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。

 **備註：**

請確保「連線資訊」畫面中的**受信任**核取方塊已勾選，以便指出該使用者目錄是受信任的 SSO 來源。

2. 針對 SSO 設定 EPM System。請參閱 [設定 EPM System 進行 SSO](#)。

選取 **SSO 提供者或代理程式**清單中的 Oracle Access Manager。如果來自 Oracle Access Manager 的 HTTP 標頭使用 HYPLOGIN 以外的名稱，請在 **SSO 機制**清單旁邊的文字方塊中，輸入該自訂標頭的名稱。

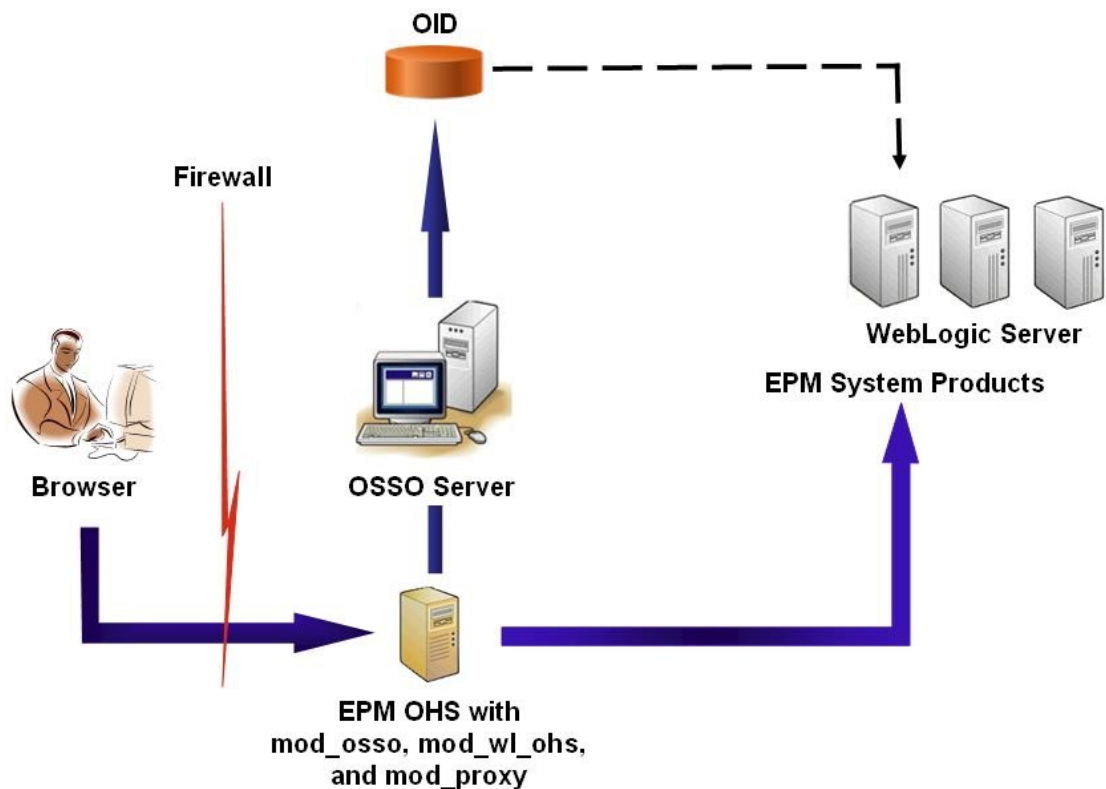
3. 僅限 Oracle Data Relationship Management：

- a. 針對 Shared Services 驗證來設定 Data Relationship Management。
- b. 在 Data Relationship Management 主控台中啟用 SSO。
如需詳細資訊，請參閱 Data Relationship Management 文件。

OracleAS Single Sign-on

OracleAS Single Sign-on (OSSO) 解決方案藉由把 Oracle Internet Directory (OID) 作為使用者目錄，提供 Web 應用程式的 SSO 存取權。使用者可以使用在 OID 中定義的使用者名稱和密碼，登入 Oracle Enterprise Performance Management System 產品。

處理流程



OSSO 程序：

1. 使用者利用某個 EPM System URL (例如 `http://OSSO_OHS_Server_NAME:OSSO_OHS_Server_PORT/interop/index.jsp`)，存取某個已定義為受 OSSO 保護之應用程式的 EPM System 元件。
2. 由於該 URL 受到 OSSO 的保護，佈署在 Oracle HTTP Server 上的 `mod_osso` 會攔截要求。Oracle HTTP Server 會使用 `mod_osso` 來尋找有效的 Cookie。如果要求中沒有有效的 Cookie，Oracle HTTP Server 就會將使用者重新導向至 OSSO 伺服器，而該伺服器會查問使用者以取得認證，並根據 OID 來驗證該使用者。
3. OSSO 伺服器會建立 `obSSOCookie`，並將控制權歸還給 Oracle HTTP Server 上的 `mod_osso` 模組，而該模組會在瀏覽器中設定 `obSSOCookie`。它也會透過 `mod_wl_ohs` (Oracle WebLogic Server) 將要求重新導向至 EPM System 資源。Oracle HTTP Server 在將要求轉送至 EPM System 資源之前，會先設定 `Proxy-Remote-User` 標頭，讓 EPM System 安全性能用來啟用 SSO。

4. EPM System 元件會確認，該使用者 (系統會從 Proxy-Remote-User 擷取其識別) 是否在 OID 中。如要讓該程序成功運作，您必須在 Oracle Hyperion Shared Services 中，將利用 OSSO 伺服器設定的 OID 設定為外部使用者目錄。

先決條件

1. 功能完整的 Oracle Application Server 基礎架構。

若要建立 Oracle Application Server 基礎架構，請安裝並設定 Oracle Identity Management 基礎架構 10.1.4。請確保 OSSO 已啟用。Oracle Identity Management 基礎架構 10.1.4 安裝包含下列元件來支援 OSSO。

- Oracle 10g OSSO 伺服器。
- OID，能讓 OSSO 伺服器用來驗證認證。請參閱以下手冊：
 - *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*
 - *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*
- Oracle HTTP Server，作為 OSSO 伺服器的前端。這個安裝包含 mod_osso，它可讓您針對 OSSO 定義合作夥伴應用程式。

備註：

這個 Oracle HTTP Server 例項是 OSSO 基礎架構的一部分，不會直接用來設定 EPM System 元件的 OSSO。

請在安裝過程中，確保您已經在 OSSO 伺服器上將 mod_osso 登錄為合作夥伴應用程式。

2. 功能完整的 EPM System 部署。
當您設定 EPM System 元件的 Web 伺服器時，EPM System Configurator 會在 Oracle HTTP Server 上設定 mod_wl_ohs.conf，以將代理主機要求傳送至 WebLogic Server。

測試部署

當您完成 SSL 的部署之後，請確認一切是否都正常運作。

如何測試部署：

1. 利用瀏覽器來存取安全的 Oracle Hyperion Enterprise Performance Management Workspace URL：

如果您把 epm.myCompany.com 當做外部通訊用的伺服器別名，且把 4443 當做 SSL 連接埠，則 EPM Workspace URL 為

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

2. 在「登入」畫面上，輸入使用者名稱與密碼。
3. 按一下**登入**。

4. 確認您可以安全地存取已部署的 Oracle Enterprise Performance Management System 元件。

啟用 EPM System 的 OSSO

本節假設，您已經完整設定 OSSO 基礎架構。請參閱 *Oracle Application Server Administrator's Guide*。

將 EPM System Web 伺服器登錄為合作夥伴應用程式

您將使用 Oracle Identity Manager SSO 登錄工具 (ssoreg.sh 或 ssoreg.bat)，將 Oracle Enterprise Performance Management System Web 伺服器，登錄為當作 OSSO 伺服器前端之 Oracle HTTP Server 上的合作夥伴應用程式。

請在裝載作為 OSSO 伺服器前端之 Oracle HTTP Server 的伺服器上，執行這個程序。這個程序將產生經過混淆處理的 osso.conf，並將它儲存您選擇的位置中。

如何將 EPM System Web 伺服器登錄為合作夥伴應用程式：

1. 在裝載作為 OSSO 前端之 Oracle HTTP Server 的伺服器上開啟主控台，然後瀏覽至 Oracle HTTP Server 的 `ORACLE_HOME/sso/bin` 目錄，例如 `C:\OraHome_1\sso/bin` (Windows)。
2. 執行類似下列的命令，但要加上 `-remote_midtier` 選項：

```
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
```

我們將在下方說明此命令中所用的參數。在以下說明中，合作夥伴應用程式是指作為 EPM System Web 伺服器的 Oracle HTTP Server。

- `-site_name`：指出合作夥伴應用程式的網站，例如 `epm.myCompany.com`。
- `-mod_osso_url`：指定合作夥伴應用程式的 URL，格式為 `PROTOCOL://HOST_NAME:PORT`。這是 EPM System Web 伺服器用來接受傳入的用戶端要求之 URL，例如 `http://epm.myCompany.com:19000`。
- `-config_mod_osso`：指出合作夥伴應用程式使用 `mod_osso`。您必須納入 `config_mod_osso` 參數，才能產生 `osso.conf`。
- `-update_mode`：指定更新模式。請使用預設的 `CREATE` 模式來產生新的記錄。
- `-remote_midtier`：指定 `mod_osso` 合作夥伴應用程式位於遠端中層。當合作夥伴應用程式不在 OSSO 伺服器的 `ORACLE_HOME` 中，而是位於另一個 `ORACLE_HOME` 中的時候，請使用這個選項。
- `-virtualhost`：指出合作夥伴應用程式 URL 是虛擬主機。如果您沒有要使用虛擬主機，請勿使用這個參數。如果您要登錄繫結至虛擬主機的合作夥伴應用程式，就必須在 `httpd.conf` 中定義虛擬主機。請參閱選用：定義虛擬主機。
- `-config_file`：指定 `osso.conf` 檔案產生之處的路徑。

選用：定義虛擬主機

如果您在登錄合作夥伴應用程式時使用虛擬主機 URL，您就必須定義虛擬主機，方法是更新當作 EPM System Web 伺服器之 Oracle HTTP Server 上的 httpd.conf。

如何定義虛擬主機：

1. 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/httpd.conf`。
2. 新增類似下列內容的定義。這個定義假設，Web 伺服器是在位於連接埠 `epm.myCompany.com:19400` 的虛擬伺服器 `epm.myCompany.com` 上執行。請依照您的需求來修改這些設定。

```
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>
```

建立 mod_osso.conf

請在作為 EPM System Web 伺服器前端的 Oracle HTTP Server 上建立 `mod_osso.conf`。

如何建立 `mod_osso.conf`：

1. 使用文字編輯器建立一個檔案。
2. 把以下內容複製到該檔案，然後針對您的環境來修改檔案。

```
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` 定義中，納入類似下列內容的位置定義，以便指出您計畫要使用 OSSO 保護的每個資源。

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

4. 將檔案儲存成 `mod_osso.conf`。

改變 `osso.conf` 的位置

將 EPM System Web 伺服器登錄為合作夥伴應用程式的程序 (請參閱[將 EPM System Web 伺服器登錄為合作夥伴應用程式](#))，會在 `-config_file` 指示詞指定的位置中建立經過混淆處理的 `osso.conf` 檔案。

如何改變 `osso.conf` 的位置：

1. 尋找當您在將 EPM System Web 伺服器登錄為合作夥伴應用程式時所建立的 `osso.conf` (請參閱[將 EPM System Web 伺服器登錄為合作夥伴應用程式](#))。
2. 將 `osso.conf` 複製到您在 `mod_osso.conf` 中定義的 `OssosConfigFile` 特性所指定的目錄 (在作為 OSSO 伺服器前端的 Oracle HTTP Server 上) 中 (請參閱[建立 mod_osso.conf](#))。

針對 OSSO 來設定 EPM System

將與 OSSO 解決方案整合的 OID，設定成 EPM System 中的外部使用者目錄，然後啟用 SSO。

如何針對 OSSO 來設定 EPM System：

1. 將 OSSO 解決方案所用的 OID 設定成外部使用者目錄。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。
2. 啟用 EPM System 中的 SSO。設定 [EPM System 進行 SSO](#)

備註：

如要將 OSSO 設定為識別管理解決方案，您必須在 **SSO 提供者或代理程式** 中選擇其他，以及在 **SSO 機制** 中選擇自訂 HTTP 標頭，然後輸入 `Proxy-Remote-User` 來做為自訂 HTTP 標頭的名稱。

3. 至少提供一位 OID 使用者來作為 Oracle Hyperion Shared Services 管理員。
4. 重新啟動使用 Shared Services 安全性 API 的 EPM System 產品與自訂應用程式。

備註：

請在重新啟動 EPM System 產品之前，確保利用 Shared Services 設定的 OID 正在執行中。

選用：啟用 OSSO 伺服器上的偵錯訊息

如要記錄 OSSO 伺服器的偵錯訊息，請修改 `policy.properties`。系統會把偵錯訊息寫入 `ORACLE_HOME/sso/log/ssoServer.log`。

如何記錄偵錯訊息：

1. 使用文字編輯器，開啟 OSSO 伺服器上的 `ORACLE_HOME/sso/conf/policy.properties`，例如 `C:\OraHome_1\sso\conf\policy.properties`。

- 將 debugLevel 特性的值設定為 DEBUG。

```
debugLevel = DEBUG
```

- 儲存並關閉 policy.properties。

選用：啟用受保護資源的偵錯訊息

如要記錄使用 mod_osso.conf 保護之資源的 OSSO 偵錯訊息，請修改 EPM System Web 伺服器上的 httpd.conf。系統會把偵錯訊息寫入 `EPM_ORACLE_INSTANCE/httpConfig/ohs/diagnostics/logs/OHS/ohs_component/ohs_component.log`。

如何記錄受保護資源的偵錯訊息：

- 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/OHS/ohs_component/httpd.conf`。
- 將 OraLogSeverity 特性的值設定為 TRACE。

```
OraLogSeverity TRACE:32
```

- 儲存並關閉 httpd.conf。

保護 EPM System 產品進行 SSO

您必須保護 Oracle Enterprise Performance Management System 資源，才能將使用者的 SSO 要求重新導向至安全性代理程式 (OAS、OSSO 或 SiteMinder)。

Oracle HTTP Server 會使用 mod_osso，將使用者重新導向至 OSSO 伺服器。除非使用者所要求之 URL 在 mod_osso 中設為保護，否則將不會重新導向使用者。請參閱 *Oracle HTTP Server Administrator's Guide* 中的 [Managing Security](#)。

如需保護 SiteMinder SSO 資源的相關資訊，請參閱 SiteMinder 文件。

僅限 OAM：防止預設標頭新增至回應

根據預設，OAM 會新增兩個標頭來保護 URL：Pragma: no-cache 和 Cache-Control: no-cache。因為這些標頭會與 EPM System 和 Web 應用程式新增的類似快取指令發生衝突，所有瀏覽器可能不會快取造成效能緩慢的受保護 URL 內容。

如須防止這些 OAM 標頭新增至回應的詳細資訊，請參閱 *Oracle Access Manager 搭配 Oracle Security Token Service 的 Fusion Middleware 管理員手冊* 中 [Oracle Access Management 效能調整](#) 一節的 *調整 OAM 代理程式*。

要保護的資源

下表列出必須要保護的環境定義。保護 OSSO 資源的語法 (以 interop 為範例)：

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

```
satisfy any
</Location>
```

allow from 參數可指定略過內容保護的伺服器。

針對 Oracle Hyperion Enterprise Performance Management Workspace 和 Oracle Hyperion Financial Reporting，您只需要設定下列範例中的參數：

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

表格 3-1 要保護的 EPM System 資源

EPM System 產品	要保護的環境定義
Oracle Hyperion Shared Services	<ul style="list-style-type: none"> · /interop · /interop/.../*
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/.../* · /hfmofficeprovider · /hfmofficeprovider/.../* · /hfmsmartviewprovider · /hfmsmartviewprovider/.../*
Oracle Hyperion Financial Reporting Web Studio	<ul style="list-style-type: none"> · /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"> · /hblauncher · /hblauncher/.../*
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/.../*

表格 3-1 (續) 要保護的 EPM System 資源

EPM System 產品	要保護的環境定義
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/**
Oracle Essbase	<ul style="list-style-type: none"> · /essbase/.../* · /essbase/** · /essbase*

要取消保護的資源

下表列出必須要取消保護的環境定義。取消保護 OSSO 資源的語法 (以 /interop/framework(.*) 為範例) 如下：

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

表格 3-2 要取消保護的 EPM System 資源

EPM System 產品	要取消保護的環境定義
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

表格 3-2 (續) 要取消保護的 EPM System 資源

EPM System 產品	要取消保護的環境定義
EPM Workspace	<ul style="list-style-type: none"> · /epmstatic/.../* · /workspace/bpmstatic/.../* · /workspace/static/.../* · /workspace/cache/.../*
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 Calculation Manager	<ul style="list-style-type: none"> /drm-migration-client · /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 · /hfmlcmsservice

表格 3-2 (續) 要取消保護的 EPM System 資源

EPM System 產品	要取消保護的環境定義
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/.../*
Performance Management Architect	<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

搭配識別管理產品的標頭型 SSO

先決條件

- 完整設定的 Oracle Enterprise Performance Management System。識別管理產品的目錄伺服器必須設定為 EPM System 中用於授權使用者的使用者目錄。
- 可支援標頭型驗證之完整設定的識別管理產品 (Microsoft Azure AD、Okta 等)。

下列一般程序與針對搭配相容識別管理產品的標頭型 SSO 設定 EPM System 相關。因為相關的特定步驟取決於您所使用的產品，請參閱識別管理產品手冊以瞭解詳細步驟。

如需設定搭配 Oracle Identity Cloud Services 之標頭型驗證的詳細步驟，請參閱[針對搭配 Oracle Identity Cloud Services 的標頭型 SSO 設定 EPM System](#)。

1. 將 EPM System 登錄為識別管理產品中的企業應用程式。此步驟允許識別管理的管理員設定企業應用程式的驗證，包含像是多重因子驗證等支援功能。
將附加 workspace/index.jsp (例如，https://gateway.server.example.com:443/workspace/index.jsp) 之閘道的完整網域名稱 (FQDN) 用作 EPM System 的企業應用程式 URL。
設定 EPM System 企業應用程式以傳播 HTTP 標頭。
您可以選擇任何非保留標頭名稱作為 HTTP 標頭名稱。標頭值應是能唯一識別 EPM System 使用者的特性。
2. 安裝、設定並登錄應用程式閘道，以確保企業應用程式只會將已驗證的要求轉送至 EPM System。
使用下列組態設定：
 - 將閘道伺服器的 FQDN (例如，gateway.server.example.com:443) 當作存取點。
 - 將 EPM System 的 FQDN (例如，epm.server.example.com:443) 當作應對其轉送已驗證之 HTTP 要求的資源。
3. 在 EPM System 中啟用 SSO 來支援應用程式閘道所傳播的 HTTP 標頭。如需詳細資料，請參閱[設定安全性選項](#)。
若要啟用 SSO，請執行下列動作：
 - a. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
 - b. 依序選取**管理**及**設定使用者目錄**。
 - c. 按一下**安全性選項**。
 - d. 在**單一登入組態**區段中：
 - i. 選取**啟用 SSO** 核取方塊。
 - ii. 在 **SSO 提供者或安全性代理程式** 下拉清單中，選取**其他**。
 - iii. 在 **SSO 機制** 下拉清單中，選取**自訂 HTTP 標頭**，然後指定安全性代理程式傳遞至 EPM System 的標頭名稱。
 - e. 按一下**確定**。
4. 將 Oracle Hyperion Enterprise Performance Management Workspace 登出後的 URL 設定更新為您想要使用者在登出 EPM System 時會看到的網頁 URL。
若要更新 EPM Workspace 中的登出後的 URL 設定，請執行下列動作：

- a. 以系統管理員的身分存取 EPM Workspace。請參閱 [存取 EPM Workspace](#)。
 - b. 依序選取**導覽**、**Workspace 設定**，然後選取**伺服器設定**。
 - c. 在 **Workspace Server 設定**中，將**登出後的 URL** 變更為您想要使用者在登出 EPM System 時會看到的網頁 URL。
 - d. 按一下**確定**。
5. 重新啟動 Oracle Hyperion Foundation Services 和所有 EPM System 管理的伺服器。

針對搭配 Oracle Identity Cloud Services 的標頭型 SSO 設定 EPM System

在此案例中，Oracle Identity Cloud Services 會驗證 Oracle Enterprise Performance Management System 使用者，並傳播所需的 HTTP 標頭來啟用 SSO。

本節討論設定 EPM System 以支援搭配 Oracle Identity Cloud Services 之 SSO 的相關步驟。您可以推斷這些步驟以支援 EPM System 的標頭型驗證、任何識別管理系統 (例如，Azure AD)，或支援標頭型驗證的基礎架構作為服務 (IaaS) 提供者。

概念工作流程如下：

- 閘道應用程式可作為反向代理主機，藉由限制未驗證的網路存取來保護 EPM System 元件。
- 閘道應用程式會攔截對 EPM System 元件所發出的 HTTP 要求，並確保識別管理產品在將要求轉送至 EPM System 元件之前先驗證使用者。
- 將要求轉送至 EPM System 元件時，閘道應用程式會透過 HTTP 標頭要求，將已驗證的使用者識別傳播至 EPM System 元件。

先決條件與範例 URL

若要建立搭配 Oracle Identity Cloud Services 的標頭型 SSO，請執行下列動作：

- 完整設定的 Oracle Enterprise Performance Management System。
- 具有完整設定之 Oracle App Gateway 的主機或容器，可作為反向代理主機或容器，藉由限制未經授權的存取來保護 EPM System。Oracle App Gateway 應設定為攔截對 EPM System 元件所發出的 HTTP 要求，並確保 Oracle Identity Cloud Services 在將要求轉送至 EPM System 之前已驗證使用者。將要求轉送至 EPM System 元件時，Oracle App Gateway 應透過 HTTP 標頭要求來傳播已驗證的使用者識別。
- Oracle Identity Cloud Services 的網域管理員存取權。

本討論使用下列範例 URL：

- Oracle Identity Cloud Services 伺服器 (識別提供者) 的完整網域名稱 (FQDN) 基礎 URL：
`https://identity.server.example.com:443/`
- Oracle App Gateway 伺服器 (管理閘道應用程式) 的 FQDN：
`https://gateway.server.example.com:443/`

- EPM System 的企業應用程式 URL。這是附加 workspace/index.jsp 之 Oracle App Gateway 伺服器的 FQDN：
https://gateway.server.example.com:443/workspace/index.jsp

 **Note:**

Oracle Identity Cloud Services 與 Oracle App Gateway 皆已設定 HTTPS 支援。EPM System 的 HTTPS 支援為選擇性的。此討論假設 EPM System 已設定 HTTPS 支援。

啟用 EPM System 的標頭型驗證

啟用 Oracle Enterprise Performance Management System 的標頭型驗證包含下列步驟：

- 將 EPM System 應用程式與閘道新增至 Oracle Identity Cloud Services
- 設定 App Gateway
- 設定用於授權的使用者目錄
- 在 EPM System 中啟用 SSO
- 更新 EPM Workspace 設定

將 EPM System 應用程式與閘道新增至 Oracle Identity Cloud Services

若要設定標頭型驗證，您必須將 Oracle Enterprise Performance Management System 建立為企業應用程式。

在 Oracle Cloud Identity Console 中將 EPM System 新增為企業應用程式

若要將 EPM System 新增為企業應用程式，請執行下列動作：

1. 以網域管理員的身分存取 Oracle Cloud Identity Console。
 - a. 使用瀏覽器，前往 <https://www.oracle.com/cloud/sign-in.html>。
 - b. 輸入您的 Oracle Fusion Cloud EPM 帳戶名稱。
 - c. 在 Oracle Fusion Cloud EPM 帳戶登入頁面中，輸入您的使用者名稱與密碼，然後按一下**登入**。
 - d. 在**導覽側邊功能表**中，按一下**使用者**，然後按一下**識別 (主要)**。
 - e. 按一下**識別主控台**。
2. 將 EPM System 新增為企業應用程式。
 - a. 在**導覽側邊功能表**中，按一下**應用程式**
 - b. 按一下**新增**，然後按一下**企業應用程式**。

The screenshot shows the Oracle Identity Cloud Service console. The left sidebar contains navigation options: Dashboard, Users, Groups, Applications (selected), Oracle Cloud Services, Jobs, Reports, Settings, and Security. The main content area is titled 'Add Enterprise Application' and has a progress indicator with three steps: 1. Details (active), 2. OAuth Configuration, and 3. SSO Configuration. The 'Details' section includes the following fields:

- Name: EPM System
- Description: On-Premises EPM 11.2
- Application Icon: A cloud icon with a document, with an 'Upload' button below it.
- Application URL: r.example.com:443/workspace/index.jsp
- Custom Login URL: (empty)
- Custom Logout URL: (empty)
- Custom Error URL: (empty)
- Linking callback URL: (empty)

Below the form are sections for 'Tags' and 'Settings'.

Tags: Add tags to your applications to organize and identify them. A tag consists of a key-value pair. + Add Tag

Settings:

- Display in My Apps
- User can request access
- User must be granted the app

3. 新增應用程式詳細資料：
 - a. 在**名稱**中，輸入識別 EPM System 企業應用程式的唯一名稱。
 - b. 輸入描述 (選用性)。
 - c. 選擇性地上傳 EPM System 的應用程式圖示。按一下**上傳**來選取並上傳圖示。
 - d. 在**應用程式 URL**中，輸入閘道應將使用者重新導向的目標啟動 URL。此 URL 是附加 workspace/index.jsp 之 Oracle App Gateway 的 FQDN，也就是 EPM System 應用程式內容。
 - e. 在**設定**下，選取**在我的應用程式中顯示**，來在 Oracle Cloud Identity Console 之**我的應用程式**頁面的 **SSO 組態**頁籤中顯示 EPM System 企業應用程式。
 - f. 按一下**下一步**。
4. 指定 SSO 組態詳細資料。
 - a. 按一下 **SSO 組態**。
 - b. 新增企業應用程式的資源。
 - i. 按一下**新增**。

Add Resource [X]

* Resource Name

* Resource URL

URL Query String

Regex

Description

OK

- ii. 指定唯一的資源名稱。
 - iii. 在**資源 URL** 中，輸入 /.*。
 - iv. 選取 **Regex** 核取方塊。
 - v. 按一下**確定**。
 - vi. 在 **SSO 組態** 中，展開**資源**。
- c. 新增驗證原則。
- 在 **SSO 組態** 中，展開**驗證原則**。
 - i. 選取**允許 CORS** 與**需要安全 Cookie** 核取方塊。
 - ii. 按一下**受管理的資源**下的**新增**，然後將**表單或存取憑證**定義為 SSO 資源的驗證方法。

Add Resource [X]

* Resource [Search]

* Authentication Method [Dropdown]

Authentication Method Overrides **+**

Headers **+**

Name	Value
<input type="text" value="HYPLOGIN"/>	<input type="text" value="Work Email"/> [Dropdown] [X]

Add

- iii. 在**資源**中，選取您在前一個步驟新增的 SSO 資源。
- iv. 展開**標頭**。
- v. 輸入要傳播至 EPM System 的 HTTP 標頭名稱。

預設驗證標頭名稱為 HYPLOGIN。您可以使用您所選擇的任何名稱。

- vi. 在**值**中，選取能唯一識別 EPM System 使用者的特性。
此欄位值應符合 EPM System 中的使用者識別。例如，如果 EPM System 中的使用者識別是電子郵件 ID，則將值選取為「工作電子郵件」。
 - vii. 按一下**儲存**。
5. 按一下**完成**來建立企業應用程式。
 6. 按一下**啟動**來啟用應用程式。
 7. 登錄 App Gateway 並設定 EPM System 的主機與應用程式。
 - a. 在**導覽側邊功能表**中，按一下**安全性**，然後按一下**應用程式閘道**。
 - b. 按一下**新增**。
 - c. 在**詳細資料**中，輸入閘道的唯一名稱與選擇性描述。
 - d. 按一下**下一步**來開啟「主機」畫面。
 - e. 新增 EPM System 的 App Gateway 主機。
 - i. 在「主機」畫面中，按一下**新增**。

- ii. 在**主機識別碼**中，輸入 EPMAppGateway。
- iii. 在**主機**中，輸入管理 App Gateway 伺服器之電腦的完整網域名稱，例如，gateway.server.example.com。
- iv. 在**連接埠**中，輸入 App Gateway 伺服器回應 HTTP 要求的連接埠。
- v. 選取 **SSL 已啟用**核取方塊。
- vi. 在**其他特性**中，輸入下列資訊：
 - SSL 憑證位置
 - SSL 憑證金鑰
 - SSL 密碼檔案 (如有需要)

如需詳細資訊，請參閱 *管理 Oracle Identity Cloud Service* 中「設定 App Gateway」的[登錄 App Gateway](#)。

- vii. 按一下**儲存**。
- viii. 按一下**下一步**來開啟「應用程式」畫面。
- f. 將 EPM System 企業應用程式新增至 App Gateway。
 - i. 在**應用程式**中，按一下**新增**。
 - ii. 在**應用程式**中，選取您先前新增至 Oracle Cloud Identity Console 的 EPM System 企業應用程式。

The screenshot shows a dialog box titled "Assign an App to gate" with a close button (X) in the top right corner. The dialog 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**選取主機**中，選取 EPMAAppGateway (您新增至 App Gateway 的 EPM System 主機)。
 - iv. 在**資源首碼**中，輸入 / 以將所有要求轉送至 EPM System 主機。
 - v. 在**原始伺服器**中，輸入管理 Oracle Hyperion Enterprise Performance Management Workspace 之電腦的完整網域名稱，以及 EPM Workspace 使用的連接埠號碼。
 - vi. 按一下**儲存**。
8. 記錄 App Gateway 的用戶端 ID 與用戶端密碼。設定 App Gateway 需要這些值。
- a. 在**導覽側邊功能表**中，按一下**安全性**，然後按一下**應用程式閘道**。
 - b. 按一下您針對 EPM System 企業應用程式新增的閘道名稱。
 - c. 將用戶端 ID (英數字元字串) 複製到文字編輯器。
 - d. 按一下**顯示密碼**來顯示用戶端密碼。
 - e. 將用戶端密碼 (英數字元字串) 複製到文字編輯器。
 - f. 儲存文字檔。

 **Note:**

每次更新 Oracle Identity Cloud Services 組態時，都必須重新啟動 App Gateway 伺服器。若要啟動與停止 App Gateway 伺服器，請參閱[啟動與停止 App Gateway](#)。

設定 App Gateway

如需詳細資訊，請參閱 *管理 Oracle Identity Cloud Service* 中的[設定 App Gateway](#)。

您需要使用在前一節記錄的用戶端 ID 與用戶端密碼來設定 App Gateway 伺服器。

設定用於授權的使用者目錄

某些識別管理產品 (例如，Oracle Identity Cloud Services 與 Microsoft Azure) 無法直接設定為 Oracle Enterprise Performance Management System 中的使用者目錄。您可以對此類產品設定 Oracle 統一目錄或 Oracle 虛擬目錄，然後將後者設定為 EPM System 中的使用者目錄。如需設定使用者目錄的詳細步驟，請參閱[設定使用者目錄](#)。

在 EPM System 中啟用 SSO

您可以設定 Oracle Enterprise Performance Management System 中的安全性選項來啟用 SSO。如需詳細的指示，請參閱[設定安全性選項](#)。

若要啟用 SSO，請執行下列動作：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理**及**設定使用者目錄**。
3. 按一下**安全性選項**。
4. 在**單一登入組態**區段中：
 - a. 選取**啟用 SSO** 核取方塊。
 - b. 在 **SSO 提供者或安全性代理程式** 下拉清單中，選取**其他**。
 - c. 在 **SSO 機制** 下拉清單中，選取**自訂 HTTP 標頭**，然後指定安全性代理程式傳遞至 EPM System 的標頭名稱 (HYPLOGIN 或您在 Oracle Cloud Identity Console 中新增企業應用程式資源時指定的自訂名稱)。
5. 按一下**確定**。

 **Note:**

確定您在任一 SSO 組態變更之後重新啟動所有 EPM System 服務。

更新 EPM Workspace 設定

1. 以系統管理員的身分存取 Oracle Hyperion Enterprise Performance Management Workspace。請參閱[存取 EPM Workspace](#)。

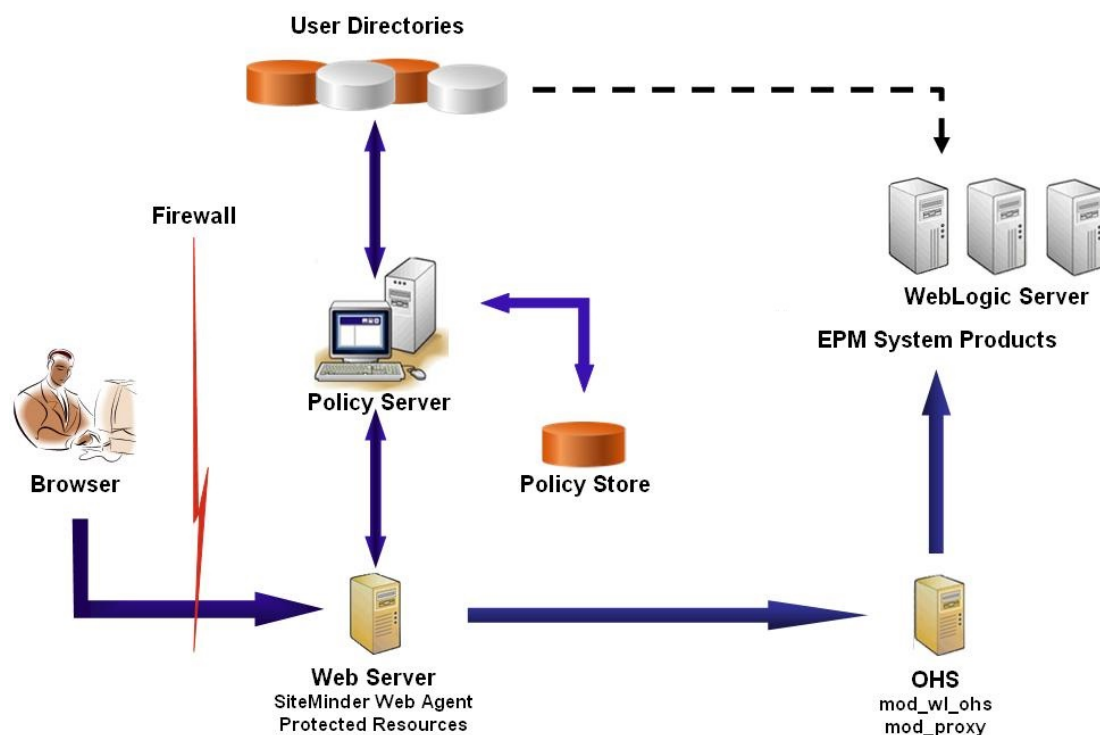
- 依序選取**導覽**、**Workspace 設定**，然後選取**伺服器設定**。
- 在 **Workspace Server 設定**中，將**登出後的 URL** 變更為您想要使用者在登出 Oracle Enterprise Performance Management System 時會看到的網頁 URL。
- 按一下**確定**。
- 重新啟動 Oracle Hyperion Foundation Services 和所有 EPM System 元件。

SiteMinder SSO

SiteMinder 是僅限 Web 的解決方案。桌面應用程式及其增益集 (如 Microsoft Excel 與 Report Designer) 皆無法透過 SiteMinder 使用驗證。然而，Oracle Smart View for Office 可以使用 SiteMinder 驗證機制。

處理流程

以下圖例說明已啟用 SiteMinder 的 SSO 概觀：



SiteMinder SSO 程序：

- 使用者嘗試存取受 SiteMinder 保護的 Oracle Enterprise Performance Management System 資源。他們會使用可連線至作為 SiteMinder 原則伺服器前端之 Web 伺服器的連線，例如 `http://WebAgent_Web_Server_Name:WebAgent_Web_ServerPort/interop/index.jsp`。
- Web 伺服器會將使用者重新導向至原則伺服器，讓該原則伺服器查問使用者以取得認證。當原則伺服器根據已設定的使用者目錄來驗證該認證之後，就會將該認證傳遞到裝載 SiteMinder Web 代理程式的 Web 伺服器上。

3. 裝載 SiteMinder Web 代理程式的 Web 伺服器會將要求重新導向到作為 EPM System 前端的 Oracle HTTP Server。Oracle HTTP Server 會將使用者重新導向到部署在 Oracle WebLogic Server 上使用者要求存取的應用程式。
4. EPM System 元件會檢查提供資訊，然後提供內容。而您必須將 SiteMinder 用來驗證使用者的使用者目錄，設定為 EPM System 中的外部使用者目錄，這個程序才能成功運作。這些目錄都必須設定為受信任。

特殊考量

SiteMinder 是僅限 Web 的解決方案。桌面應用程式及其增益集 (如 Microsoft Excel 與 Report Designer) 皆無法透過 SiteMinder 使用驗證。然而，Smart View 可以使用 SiteMinder 驗證機制。

先決條件

1. 功能完整的 SiteMinder 安裝包含下列元件：
 - SiteMinder 原則伺服器：您會在該伺服器上定義原則和代理程式物件
 - SiteMinder Web 代理程式：安裝在作為 SiteMinder 原則伺服器前端的 Web 伺服器上
2. 功能完整的 EPM System 部署。
當您設定 EPM System 元件的 Web 伺服器時，EPM System Configurator 會設定 `mod_wl_ohs.conf`，以將要求代理至 WebLogic Server。

建立 SiteMinder Web 代理程式

Web 代理程式安裝在會攔截針對 EPM System 資源之要求的 Web 伺服器上。若有未通過驗證的使用者嘗試存取受保護的 EPM System 資源，Web 代理程式就會查問使用者以取得 SSO 認證。使用者經過驗證後，原則伺服器會新增已驗證使用者的登入名稱，並在標頭中包含此登入名稱。之後，系統會把 HTTP 要求傳遞到 EPM System Web 伺服器，讓該伺服器重新導向要求。EPM System 元件會從標頭擷取已通過驗證的使用者認證。

SiteMinder 支援在異質 Web 伺服器平台上執行之 EMP System 產品的 SSO 機制。如果 EMP System 產品使用不同的 Web 伺服器，您必須確保 SiteMinder Cookie 可以在相同網域的不同 Web 伺服器之間傳遞。方法是指定適當的 EPM System 應用程式網域，作為每台 Web 伺服器上 `WebAgent.conf` 檔案中的 `Cookiedomain` 特性值。

請參閱 *Netegrity SiteMinder Agent Guide* 中的 "Configuring Web Agents"。

備註：

由於 Oracle Hyperion Shared Services 會使用基本的驗證功能來保護自己的內容，因此我們建議您在攔截 Shared Services 要求的 Web 伺服器上啟用基本的驗證功能，以支援使用 SiteMinder 的 SSO。

您將設定 Web 代理程式，方法是執行 SiteMinder Web 代理程式組態精靈 (藉由執行 `WEBAGENT_HOME/install_config_info/nete-wa-config` 來進行，例如 Windows 上的 `C:\netegrity\webagent\install_config_info\nete-wa-config.exe`)。這個設定程序將會為 SiteMinder Web 伺服器建立 `WebAgent.conf`。

如何啟用 SiteMinder Web 代理程式：

1. 使用文字編輯器開啟 `WebAgent.conf`。這個檔案的位置，取決於您要使用的 Web 伺服器。
2. 將 `enableWebAgent` 特性的值設定為 `Yes`。
`enableWebAgent="YES"`
3. 儲存並關閉 Web 代理程式組態檔。

範例 3-1 設定 SiteMinder 原則伺服器

SiteMinder 管理員必須設定原則伺服器，才能啟用 EMP System 產品的 SSO。

設定程序包括：

- 建立 SiteMinder Web 代理程式，並新增適用於 SiteMinder Web 伺服器的組態物件。
- 為每個應受保護的 EPM System 資源建立領域，然後將該 Web 代理程式新增到該領域中。請參閱[要保護的資源](#)
- 在為受保護的 EPM System 資源建立的領域中，建立未受保護資源的領域。請參閱[要取消保護的資源](#)
- 建立 HTTP 標頭參照。標頭應該要將 Login Attribute 的值提供給 EPM System 應用程式。如需 Login Attribute 的簡短說明，請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。
- 利用 Get、Post 和 Put 等 Web 代理程式動作，在領域中建立規則
- 建立回應屬性，並把該屬性的值設定為 `hyplogin=<%userattr="SM_USERLOGINNAME"%>`
- 建立原則、指派使用者目錄存取權，以及將您為 EPM System 建立的規則新增到「目前成員」清單中
- 為您針對 EPM System 元件所建立的規則設定回應

範例 3-2 設定 SiteMinder Web 伺服器，讓該伺服器會將要求轉送到 EPM System Web 伺服器

您將設定裝載 SiteMinder Web 代理程式的 Web 伺服器，讓該伺服器會將已通過驗證之使用者 (包含可識別使用者的標頭) 的要求轉送到 EPM System Web 伺服器。

針對 Apache 式的 Web 伺服器，請使用類似下列內容的指示詞來轉送已通過驗證的要求：

```
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
```

請在上述指示詞中，將 `EPM_WEB_SERVER` 和 `EPM_WEB_SERVER_PORT` 替換成您環境中實際的值。

範例 3-3 在 EMP System 中啟用 SiteMinder

您必須啟用 EMP System 產品的 SiteMinder 驗證，才能與 SiteMinder 進行整合。請參閱 [設定 EPM System 進行 SSO](#)。

Kerberos 單一登入

簡介

Oracle Enterprise Performance Management System 產品支援 Kerberos SSO 機制，但前提是裝載 EPM System 產品的應用程式伺服器已設定成可使用 Kerberos 驗證機制。

Kerberos 是受信任的驗證服務，其中的每個 Kerberos 用戶端都會信任另一個 Kerberos (使用者、網路服務等) 的識別。

當使用者存取 EPM System 產品時，會發生下列情況：

1. 在 Windows 電腦上，使用者會登入某個 Windows 網域，而這也是個 Kerberos 領域。
2. 使用者會使用已設定要使用整合式 Windows 驗證的瀏覽器，嘗試登入在應用程式伺服器上執行的 EPM System 產品。
3. 應用程式伺服器 (協商識別宣告器) 會攔截要求，並利用瀏覽器之授權標頭中的 Kerberos 票證，取得 Simple and Protected Generic Security Services API (GSSAPI) Negotiation Mechanism (SPNEGO) 憑證。
4. 宣告器會根據自己的識別儲存庫，驗證包含在憑證中的使用者識別，以便將該使用者的資訊傳遞到 EPM System 產品。EPM System 產品會根據 Active Directory 驗證使用者名稱。EPM System 產品會發出 SSO 憑證，而該憑證會支援所有 EPM System 產品的 SSO 機制。

支援限制

除下列例外之外，所有 EPM System 產品皆支援 Kerberos SSO：

- 豐富型用戶端並不支援 Kerberos SSO，但 Oracle Smart View for Office 除外。
- Smart View 只支援針對 Oracle Essbase、Oracle Hyperion Planning 和 Oracle Hyperion Financial Management 提供者的 Kerberos 整合。

假設

本文件 (包含應用程式層級的 Kerberos 設定步驟) 假設，您熟悉系統層級的 Kerberos 組態。在您開始執行這些程序之前，請先確認您已滿足這些工作的先決條件。

本文件假設，您已啟用 Kerberos 的網路環境功能齊全，且其中的 Windows 用戶端機器已針對 Kerberos 驗證機制來設定。

- 公司的 Active Directory 已針對 Kerberos 授權機制來設定。請參閱 [Microsoft Windows Server 文件](#)。
- 用來存取 EPM System 產品的瀏覽器，已設定成會使用 Kerberos 票證來協商。
- KDC 與用戶端機器之間時間不同步的偏差必須少於 5 分鐘。請參閱位於 [http://technet.microsoft.com/en-us/library/cc780011\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc780011(WS.10).aspx) 的「Authentication Errors are Caused by Unsynchronized Clocks」。

搭配 WebLogic Server 的 Kerberos SSO

Oracle WebLogic Server Kerberos SSO 會使用「協商識別宣告器」來協商，並將 SPNEGO 憑證解碼以啟用 Microsoft 用戶端的 SSO。WebLogic Server 會將 SPNEGO 憑證解碼以取得 Kerberos 票證，然後驗證該票證，並將它對映至 WebLogic Server 使用者。您可以搭配「協商識別宣告器」來使用 WebLogic Server 的 Active Directory 認證者，將 Active Directory 設定成 WebLogic Server 使用者的使用者目錄。

當瀏覽器要求存取 EMP System 產品時，KDC 會發出 Kerberos 票證給瀏覽器，以建立包含受支援 GSS 憑證類型的 SPNEGO 憑證。「協商識別宣告器」會將 SPNEGO 憑證解碼，並使用 GSSAPI 來接受安全相關資訊環境。發起請求的使用者識別會先對映至使用者名稱，然後再傳回 WebLogic Server。此外，WebLogic Server 會決定使用者所屬的群組。在此階段，要求的 EMP System 產品即可供使用者使用。

備註：

使用者必須使用支援 SPNEGO 的瀏覽器 (例如 Internet Explorer 或 Firefox)，存取在 WebLogic Server 上執行的 EMP System 產品。

EMP System 產品授權程序會使用衍生自驗證程序的使用者 ID，檢查提供資料。EMP System 產品存取權的限制取決於提供資料。

支援 Kerberos 驗證的 WebLogic Server 程序

管理員必須完成下列工作來支援 Kerberos 驗證：

- 建立 EPM System 的 WebLogic 網域。請參閱[建立 EPM System 的 WebLogic 網域](#)。
- 建立驗證提供者。請參閱 [在 WebLogic Server 中建立 LDAP 驗證提供者](#)。
- 建立協商識別宣告器。請參閱 [建立協商識別宣告器](#)。
- 建立 Kerberos 識別。請參閱[建立 WebLogic Server 的 Kerberos 識別](#)。
- 更新 Kerberos 的 JVM 選項。請參閱[更新 Kerberos 的 JVM 選項](#)。
- 設定授權原則。請參閱[設定授權原則](#)。
- 部署並使用 SSODiag 來確認 WebLogic Server 是否已準備好支援 EPM System 的 Kerberos SSO 機制。請參閱[使用 SSODiag 來測試 Kerberos 環境](#)。

建立 EPM System 的 WebLogic 網域

一般來說，EPM System 元件已部署到 EPMSYSTEM WebLogic 網域中 (預設位置為 `MIDDLEWARE_HOME/user_projects/domains/EPMSYSTEM`)。

如何針對 Kerberos 驗證機制來設定 EPM System WebLogic 網域：

1. 安裝 EPM System 元件。
2. 只部署 Oracle Hyperion Foundation Services。
Foundation Services 部署會建立預設的 EPM System WebLogic 網域。
3. 登入 Oracle Hyperion Shared Services Console 來確認 Foundation Services 是否已部署成功。請參閱[啟動 Shared Services Console](#)。

在 WebLogic Server 中建立 LDAP 驗證提供者

WebLogic Server 管理員將會建立 LDAP 驗證提供者，而該提供者會在外部 LDAP 伺服器中儲存使用者和群組資訊。與 LDAP v2 或 v3 相容的 LDAP 伺服器能與 WebLogic Server 搭配使用。請參閱下列參考資料。

- [設定 LDAP 驗證提供者](#) (位於 *Oracle Fusion Middleware Securing Oracle WebLogic Server* 手冊中)。
- [設定認證和識別宣告提供者](#) (位於 *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help* 中)。

建立協商識別宣告器

「協商識別宣告」提供者會搭配 Microsoft 用戶端啟用 SSO。這會將 SPNEGO 解碼以取得 Kerberos 憑證、驗證 Kerberos 憑證，並將憑證對映到 WebLogic 使用者。「協商識別宣告」提供者是安全服務提供者介面 (SSPI) 的實作 (如 WebLogic 安全性架構所定義)，它會提供必要的邏輯，以便根據用戶端的 SPNEGO 憑證來驗證使用者。

- [建立協商識別宣告提供者](#) (位於 *Oracle Fusion Middleware Securing Oracle WebLogic Server* 手冊中)。
- [設定認證和識別宣告提供者](#) (位於 *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help* 中)。

當您建立「協商識別宣告」提供者時，請將所有認證者的「JAAS 控制旗標」選項設定成 SUFFICIENT。請參閱 [Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help](#) 中的「設定 JAAS 控制旗標」。

建立 WebLogic Server 的 Kerberos 識別

請在 Active Directory 網域控制站機器上，建立代表 WebLogic Server 和 EPM System Web 伺服器的使用者物件，並將該物件對映到代表您的 WebLogic Server 和 Kerberos 領域中之 Web 伺服器的服務主體名稱 (SPN)。用戶端無法找到沒有 SPN 的服務。您將 SPN 儲存在已複製到 WebLogic Server 網域的 keytab 檔案，以便在登入程序中使用。

如需詳細程序，請參閱 *Oracle Fusion Middleware Securing Oracle WebLogic Server* 手冊中的 [建立 WebLogic Server 的識別](#)。

如何建立 WebLogic Server 的 Kerberos 識別：

1. 在 Active Directory 網域控制站機器上，為裝載 WebLogic Server 網域的電腦建立使用者帳戶 (例如 epmHost)。

 **備註：**

請將識別建立成使用者物件，而非機器。
請使用電腦的簡單名稱；例如，如果主機名稱為 `epmHost.example.com`，請使用 `epmHost`。

請記錄您在建立使用者物件的過程中使用的密碼。您將會在建立 **SPN** 時使用這個密碼。

請勿選取任何密碼選項，尤其是 `User must change password at next logon` 選項。

2. 修改使用者物件來遵守 Kerberos 通訊協定。帳戶必須要求 Kerberos 預先驗證。
 - 在**帳戶**頁籤上，選取您要使用的加密方式。
 - 確保您沒有選取其他的帳戶選項 (尤其是 `Do not require Kerberos pre-authentication`)。
 - 由於設定加密類型可能會損毀物件的密碼，請將密碼重設成您在建立物件時所設定的密碼。
3. 在裝載 Active Directory 網域控制站的電腦上，開啟命令提示字元視窗，然後瀏覽至安裝 Active Directory 支援工具的目錄。
4. 建立並設定必要的 SPN。
 - a. 使用類似下列內容的指令，來確認 SPN 是否已與您在本程序的步驟 1 中建立的使用者物件 (`epmHost`) 建立關聯。

```
setspn -L epmHost
```

- b. 使用類似下列內容的命令，來設定 Active Directory Domain Services (AD DS) 中 WebLogic Server 的 SPN，然後產生包含共用祕密金鑰的 `keytab` 檔案。

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

5. 在裝載 WebLogic Server 的電腦上建立 `keytab` 檔案。
 - a. 開啟命令提示字元。
 - b. 瀏覽至 `MIDDLEWARE_HOME/jdk/bin`。
 - c. 執行類似以下的命令：

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

- d. 當系統提示您輸入密碼時，請輸入您在本程序的步驟 1 中建立使用者時所設定的密碼。

6. 請將 `keytab` 檔案複製到 WebLogic 網域內部的啟動目錄中，例如複製到 `C:\Oracle\Middleware\user_projects\domains\EPMSystem`。

7. 確認 Kerberos 驗證機制的運作正常。

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

在上述命令中，`account-name` 會指定 Kerberos 原則，例如 `HTTP/epmHost.example.com@EXAMPLE.COM`。這個命令的輸出內容應該與下列內容類似：

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

更新 Kerberos 的 JVM 選項

請參閱 *Oracle Fusion Middleware Securing Oracle WebLogic Server 11g Release 1 (10.3.1)* 中的 [搭配 WebLogic Server 來使用 Kerberos 驗證機制的啟動引數](#) 和 [建立 JAAS 登入檔案](#)。

如果您把 EPM System 受管理伺服器當作 Windows 服務來執行，請更新 Windows 登錄來設定 JVM 啟動選項。

如何更新 Windows 登錄中的 JVM 啟動選項：

1. 開啟 Windows 登錄編輯程式。
2. 依序選取 **我的電腦**、**HKEY_LOCAL_MACHINE**、**Software**、**Hyperion Solutions**、**Foundationservices0**，以及 **HyS9EPMServer_epmsystem1**。
3. 建立下列字串值：

備註：

下表中所列的名稱是範例。

表格 3-3 Kerberos 驗證機制的 JVM 啟動選項

名稱	類型	資料
JVMOption44	REG_SZ	-Djava.security.krb5.realm=Active Directory 領域名稱
JVMOption45	REG_SZ	-Djava.security.krb5.kdc=Active Directory 主機名稱或 IP 位址
JVMOption46	REG_SZ	- Djava.security.auth.login.config=Kerberos 登入組態檔的位置
JVMOption47	REG_SZ	- Djavax.security.auth.useSubjectCredsOnly=false

4. 更新 JVMOptionCount DWord，以反映新增的 JVMOptions (請將目前的十進位值加 4)。

設定授權原則

如需如何針對會存取 EPM System 的 Active Directory 使用者設定授權原則的相關資訊，請參閱 *Oracle Fusion Middleware Securing Resources Using Roles and Policies for Oracle WebLogic Server* 手冊中的 [Options for Securing Web Application and EJB Resources](#)。

如需原則設定步驟的範例，請參閱[建立 SSODiag 適用的原則](#)。

使用 SSODiag 來測試 Kerberos 環境

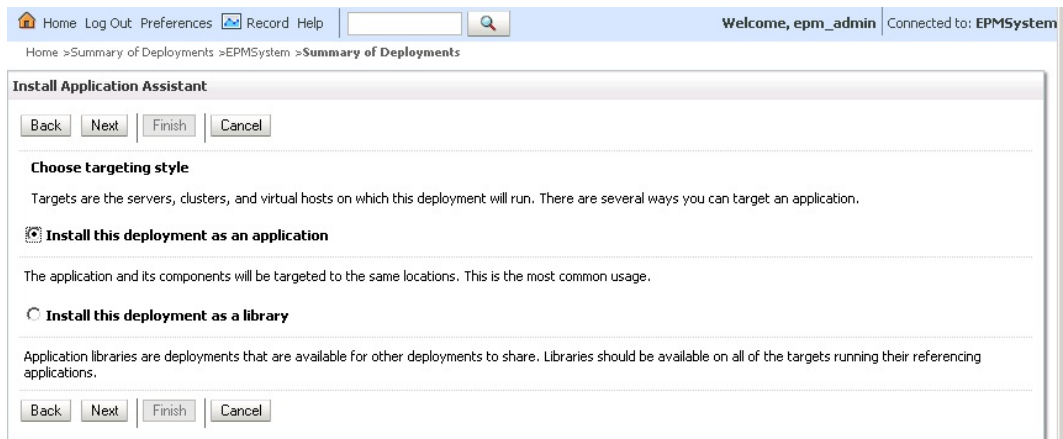
SSODiag 是診斷用的 Web 應用程式，可用來測試您 Kerberos 環境中的 WebLogic Server 是否已準備好支援 EPM System。

部署 SSODiag

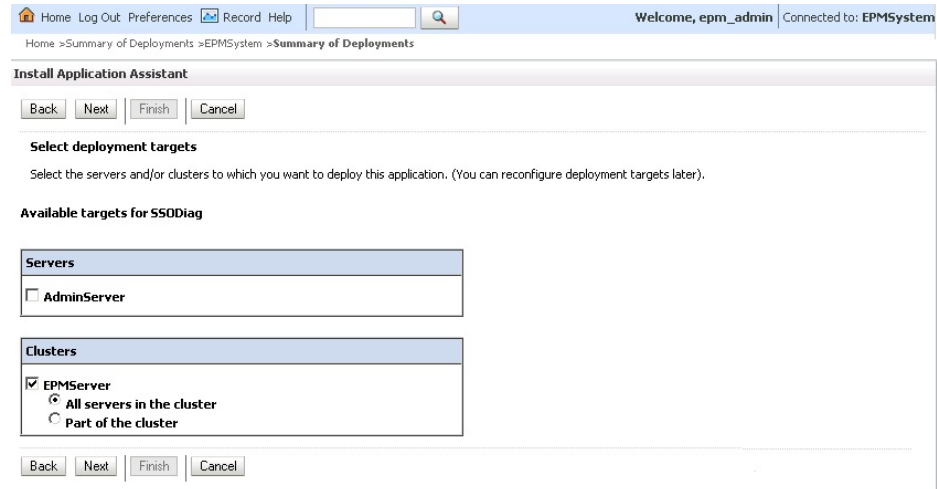
請使用您在部署 Foundation Services 時指定的 WebLogic Server 管理員認證 (預設的使用者名稱是 `epm_admin`) 來部署 SSODiag。

如何部署及設定 SSODiag：

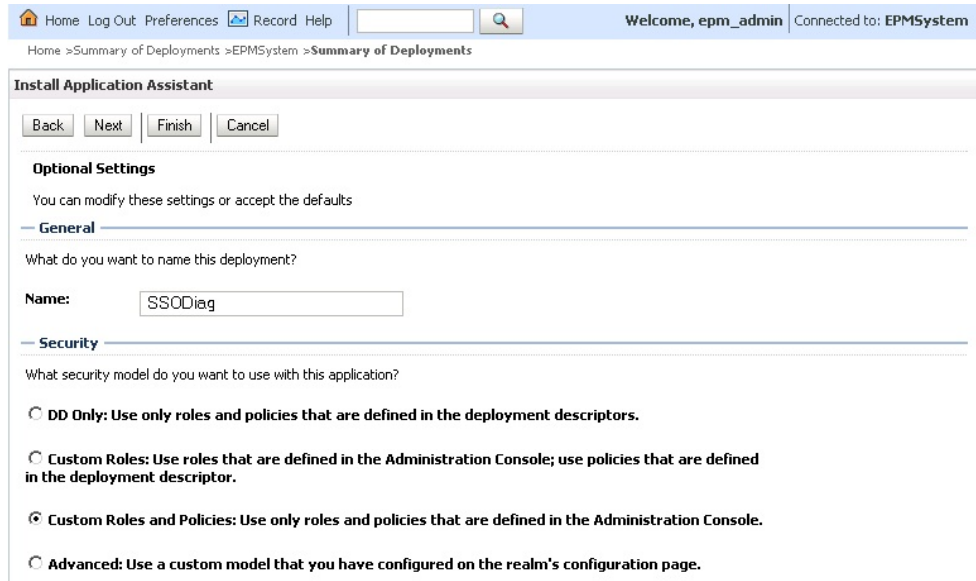
1. 登入 EPM System 網域的 WebLogic Server 管理主控台。
2. 在「變更中心」中，選取**鎖定和編輯**
3. 在**網域結構**的 **EPMSystem** 中，按一下**部署**。
4. 在**部署摘要**中，按一下**安裝**。
5. 在**路徑**中，選取 `EPM_ORACLE_HOME/products/Foundation/AppServer/InstallableApps/common/SSODiag.war`
6. 按一下**下一步**。
7. 在**選擇目標樣式**中，確定**將此部署安裝為應用程式**已勾選，然後按一下**下一步**。



8. 在**選取部署目標**中，選取下列項目，然後按一下**下一步**。
 - **EPMServer**
 - **叢集中的所有伺服器**



9. 在選擇性設定中，選取自訂角色和原則：僅使用在管理主控台中定義的角色和原則來作為安全性模型。



10. 按一下下一步。
11. 在複查畫面中，選取否，在稍後複查組態。
12. 按一下完成。
13. 在「變更中心」中，選取啟動變更。

針對 SSODiag 設定 Oracle HTTP Server

請更新 `mod_wl_ohs.conf` 來設定 Oracle HTTP Server，以便將 SSODiag URL 要求轉送到 WebLogic Server。

如何設定 Oracle HTTP Server 中的 URL 轉送功能：

1. 使用文字編輯器開啟 `EPM_ORACLE_INSTANCE/httpConfig/ohs/config/fmwconfig/components/OHS/ohs_component/mod_wl_ohs.conf`。
2. 新增 SSODiag 適用的 LocationMatch 定義：

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

在以上範例中，myServer 代表 Foundation Services 主機，28080 則代表 Oracle Hyperion Shared Services 用來監聽要求的連接埠。

3. 儲存並關閉 mod_wl_ohs.conf。
4. 重新啟動 Oracle HTTP Server。

建立 SSODiag 適用的原則

請在 WebLogic Server 管理主控台建立原則來保護以下 SSODiag URL。

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

在以上範例中，OHS_HOST_NAME 代表裝載 Oracle HTTP Server 之伺服器的名稱，PORT 則代表 Oracle HTTP Server 用來監聽要求的連接埠。

如何建立原則來保護 SSODiag：

1. 在「變更中心」中，於 EPM System 網域的 WebLogic Server 管理主控台上，選取 **鎖定和編輯**。
2. 依序選取 **部署**、**SSODiag**、**安全性**，然後 **URLPatterns 與原則**。
3. 建立下列 URL 樣式：
 - /
 - /index.jsp
4. 修改您建立的每個 URL 樣式：
 - a. 在 **獨立 Web 應用程式 URL 樣式** 的 URL 樣式清單中，按一下您建立的樣式 (/) 來開啟它。
 - b. 選取 **新增條件**。
 - c. 選取 **述詞清單** 中的 **使用者**。
 - d. 選取 **下一步**。
 - e. 在 **使用者引數名稱** 中，輸入您用來存取已設定為 Kerberos 驗證用的用戶端桌上型電腦之帳戶所屬的 Active Directory 使用者 (例如 krbuser1)，然後選取 **新增**。krbuser1 為 Active Directory 或 Windows 桌面使用者。
 - f. 選取 **完成**。
5. 選取 **儲存**。

使用 SSODiag 來測試 Kerberos 驗證適用的 WebLogic Server 組態

如果 Kerberos 驗證適用的 WebLogic Server 組態能正常運作，*Oracle Hyperion Kerberos SSO 診斷公用程式 V 1.0* 頁面就會顯示下列訊息：

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

▲ 注意：

如果 SSODiag 無法擷取 Kerberos 原則名稱，請勿針對 Kerberos 驗證機制來設定 EPM System 元件。

如何測試 Kerberos 驗證適用的 WebLogic Server 組態：

1. 啟動 Foundation Services 和 Oracle HTTP Server。
2. 使用 WebLogic Server 管理主控台來啟動 SSODiag Web 應用程式，以處理所有要求。
3. 使用有效的 Active Directory 認證，登入已針對 Kerberos 驗證設定的用戶端機器。
4. 使用瀏覽器連線到下列 SSODiag URL：

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

在以上範例中，*OHS_HOST_NAME* 代表裝載 Oracle HTTP Server 之伺服器名稱，*PORT* 則代表 Oracle HTTP Server 用來監聽要求的連接埠。

如果 Kerberos 驗證機制的運作正常，SSODiag 會顯示下列資訊：

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

如果 Kerberos 驗證機制沒有正常運作，SSODiag 會顯示下列資訊：

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

變更安全性模型

受安全性領域保護之 Web 應用程式的預設安全性模型為 `DOnly`。您必須將安全性模型變更成 `CustomRolesAndPolicies`。

如何變更安全性模型：

1. 使用文字編輯器開啟 `MIDDLEWARE_HOME/user_projects/domains/EPMSystem/config/config.xml`。

2. 在每個 Foundation Services 元件的應用程式部署描述元中尋找下列元素：

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

3. 將每個元件的安全性模性變更成以下內容：

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

4. 儲存並關閉 config.xml。

更新 EPM System 安全性組態

請變更 EPM System 安全性組態以啟用 Kerberos SSO。

如何針對 Kerberos 驗證機制來設定 EPM System：

1. 以管理員的身分登入 Shared Services Console。
2. 在 Shared Services 中，將您針對 Kerberos 驗證機制設定的 Active Directory 網域新增為外部使用者目錄。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定 OID、Active Directory 及其他 LDAP 型的使用者目錄」。
3. 啟用 SSO。請參閱 [設定 OID、Active Directory 及其他 LDAP 型的使用者目錄](#)。在 **安全性選項**，選取下表中的設定值以啟用 Kerberos SSO。

表格 3-4 啟用 Kerberos SSO 的設定

欄位	必要設定
啟用 SSO	已選取
SSO 提供者或代理程式	其他
SSO 機制	從 HTTP 要求中取得遠端使用者

4. 重新啟動 Foundation Services。

測試 Kerberos SSO

登入 Foundation Services 來確認 Kerberos SSO 是否運作正常。

如何測試 Kerberos SSO：

1. 確認 Foundation Services 和 Oracle HTTP Server 是否在執行中。
2. 使用有效的 Active Directory 認證，登入已針對 Kerberos 驗證設定的用戶端機器。
3. 使用瀏覽器連線到 Foundation Services URL：

設定 EPM System 元件

請使用 EPM System Configurator 設定其他 EPM System 元件，並將這些元件部署到 Foundation Services 部署之處的 WebLogic 網域中。

針對 Kerberos 驗證機制來設定 EPM System 受管理伺服器。

在 Microsoft Windows 環境中，EPM System 受管理伺服器會被當作 Windows 服務來執行。您必須修改每個 WebLogic 受管理伺服器的啟動 JVM 選項。以下是採非精簡部署模式之受管理伺服器的完整清單：

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

如果 EPM System Web 應用程式是採用精簡模式來部署的，您就只需要更新 EPMSys0 受管理伺服器的啟動 JVM 選項。如果您有多個精簡受管理伺服器，就必須更新所有受管理伺服器的啟動 JVM 選項。

請參閱 *Oracle Fusion Middleware Securing Oracle WebLogic Server* 手冊中的[搭配 WebLogic Server 來使用 Kerberos 驗證機制的啟動引數](#)。

備註：

以下程序說明，如何設定 Foundation Services 受管理伺服器的啟動 JVM 選項。您必須針對部署中的每個 WebLogic 受管理伺服器執行該工作。

如需在 WebLogic Server 啟動命令檔中設定 JVM 選項的詳細程序，請參閱[更新 Kerberos 的 JVM 選項](#)。

如何在 WebLogic Server 啟動命令檔中設定 JVM 選項

設定授權原則

請針對將會存取 Foundation Services 以外之 EPM System 元件的 Active Directory 使用者設定授權原則。如需如何在 WebLogic 管理主控台中設定安全性原則的相關資訊，請參閱 [設定授權原則](#)。

變更 EPM System 元件的預設安全性模型

您將編輯 EPM System 組態檔，以變更預設的安全性模型。針對非精簡的 EPM System 部署，您必須變更記錄在 config.xml 中的每個 EPM System Web 應用程式的預設安全性模型。以下是 EPM System Web 應用程式的清單：

- AIF
- APS
- CALC
- EAS
- FINANCIALREPORTING

- PLANNING
- PROFITABILITY
- SHARED SERVICES
- WORKSPACE

如何變更安全性模型：

1. 使用文字編輯器開啟 `MIDDLEWARE_HOME/user_projects/domains/EPMSysstem/config/config.xml`
2. 在每個 EPM System 元件的 `app-deployment` 定義中，將 `<security-dd-model>` 的值設定成 `CustomRolesAndPolicies`，如以下範例所示：

```
<app-deployment>
  <name>SHARED SERVICES#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. 儲存並關閉 `config.xml`。
4. 重新啟動 WebLogic Server。

建立 EPM System 元件的 URL 保護原則

請在 WebLogic Server 管理主控台中建立 URL 保護原則，以保護每個 EPM System 元件 URL。如需詳細資訊，請參閱 *Oracle Fusion Middleware Securing Oracle WebLogic Server* 手冊中的 [Options for Securing Web Applications and EJB Resources](#)。

如何建立 URL 保護原則：

1. 在「變更中心」中，於 EPM System 網域的 WebLogic Server 管理主控台上，按一下 **鎖定和編輯**。
2. 按一下 **部署**。
3. 展開部署中的某個 EPM System 企業應用程式 (例如 PLANNING)，然後按一下其 Web 應用程式 (例如 HyperionPlanning)。如需 EPM System 元件的清單，請參閱 [變更 EPM System 元件的預設安全性模型](#)。

備註：

某些企業應用程式 (例如 Oracle Essbase Administration Services) 包含多個 Web 應用程式，您必須分別定義它們的 URL 樣式。

4. 建立 Web 應用程式的 URL 樣式作用領域的原則。
 - AIF
 - APS

- CALC
 - EAS
 - FINANCIALREPORTING
 - PLANNING
 - PROFITABILITY
 - SHARED SERVICES
 - WORKSPACE
- a. 依序按一下**安全性、原則及新建**。
 - b. 在 **URL 樣式** 中，輸入受保護與未受保護的 EPM 系統產品 URL。請參閱[受保護與未受保護的 EPM 系統資源](#)以取得更多詳細資料。
 - c. 按一下**確定**。
 - d. 按一下您建立的 URL 樣式。
 - e. 按一下**新增條件**。
 - f. 選取**述詞清單**中的某個原則條件，然後按一下**下一步**。
Oracle 建議您使用 Group 條件，因為它會將這個安全性原則授予給特定群組的所有成員。
 - g. 指定與您選擇的述詞相關的引數。例如，如果您在前一個步驟選擇 Group，就必須完成接下來的步驟：
 - h. 在**群組引數名稱**中，輸入應該可存取該 Web 應用程式之使用者所屬的群組名稱。而您輸入的名稱，必須與某個 Active Directory 群組名稱完全相符。
 - 按一下**新增**。
 - 重複上述步驟來新增其他群組。
 - i. 按一下**完成**。
如果 WebLogic Server 無法在 Active Directory 中找到該群組，就會顯示錯誤訊息。您必須修正該錯誤，才能繼續進行。
 - j. 選取**儲存**。
5. 請針對部署中的其他 EPM System 元件重複本程序的步驟 3 和步驟 4。
 6. 在「變更中心」中，按一下**釋放組態**。
 7. 重新啟動 WebLogic Server。

啟用 Web 應用程式中的用戶端憑證式驗證機制

在下列位於 `EPM_ORACLE_HOME/products/` 內部之應用程式封存的組態檔中，插入 login-config 定義。

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

如何啟用用戶端憑證式的驗證機制

1. 停止 EPM System 元件和程序的執行。
2. 使用 7 Zip 來解壓縮包含在企業存檔中的 Web 封存檔，例如 `EPM_ORACLE_HOME/products/Essbase/eas/server/AppServer/InstallableApps/Common/eas.ear/eas.war`。
3. 瀏覽至 WEB-INF。
4. 修改 web.xml，方法是將下列 login_config 定義新增到 `</webapp>` 元素的正前方：

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

5. 儲存 web.xml。
6. 當 7 Zip 詢問您是否要更新封存時，按一下**是**。

更新 EPM System 安全性組態

請設定 EPM System 安全性來支援 SSO。請參閱[設定 EPM System 進行 SSO](#)。

設定 EPM System 進行 SSO

您必須將 Oracle Enterprise Performance Management System 產品設定成支援 SSO 適用的安全性代理程式。您在 Oracle Hyperion Shared Services 中指定的組態，決定了所有 EPM System 產品的下列事項：

- 是否接受來自安全性代理程式的 SSO
- 要用於進行 SSO 的驗證機制

在啟用 SSO 的環境中，使用者第一次存取的 EPM System 產品會剖析 SSO 機制，從中擷取其所包含的已驗證使用者 ID。EPM System 產品會根據您在 Shared Services 中設定的使用者目錄檢查使用者 ID，判斷該使用者是否為有效的 EPM System 使用者。同時還會發出可以在所有 EPM System 產品中啟用 SSO 的票證。

Shared Services 中所指定的組態會啟用 SSO，並決定可接受所有 EPM System 產品之 SSO 的驗證機制。

如何利用 Web 識別管理解決方案啟用 SSO：

1. 以 Shared Services 管理員的身分啟動 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理**及**設定使用者目錄**。
3. 確認您是否已將 Web 識別管理解決方案所用的使用者目錄，設定為 Shared Services 中的外部使用者目錄。

例如，如要啟用 Kerberos SSO，您必須把已針對 Kerberos 驗證機制設定的 Active Directory 設定成外部使用者目錄。

如需指示，請參閱設定使用者目錄。

4. 選取**安全性選項**。
5. 選取**顯示進階選項**。
6. 在「已定義的使用者目錄」畫面的**單一登入組態**中執行下列步驟：

- a. 選取**啟用 SSO**。
- b. 在 **SSO 提供者或代理程式**中，選取某個 Web 識別管理解決方案。如果您要使用 Kerberos 設定 SSO，請選擇**其他**。

建議的 SSO 機制會自動加以選擇。請參閱下列表格。另請參閱[受支援的 SSO 方法](#)。

 **備註：**

若不想使用建議的 SSO 機制，必須在 **SSO 提供者或代理程式**中選擇 Other。例如，如要使用 SiteMinder 的 HTTP 標頭以外的機制，請在 **SSO 提供者或代理程式**中選擇其他，然後在 **SSO 機制**中選取您要使用的 SSO 機制。

表格 3-5 Web 識別管理解決方案偏好的 SSO 機制

Web 識別管理解決方案	建議的 SSO 機制
Oracle Access Manager	自訂 HTTP 標頭 ¹
OSSO	自訂 HTTP 標頭
SiteMinder	自訂 HTTP 標頭
Kerberos	從 HTTP 要求中取得遠端使用者

¹ 預設的 HTTP 標頭名稱為 HYPLOGIN。若要使用自訂 HTTP 標頭，請置換該名稱。

7. 按一下**確定**。

Smart View 的單一登入選項

雖然 Oracle Smart View for Office 是豐富型用戶端，而非瀏覽器，但它會使用 HTTP 連線至伺服器元件，且它的表現從系統的觀點看來就像是瀏覽器。Smart View 支援瀏覽器介面支援的所有標準 Web 式整合方法。不過，其中有些限制：

- 如果 Smart View 是從已連線到某個 Oracle Enterprise Performance Management System 元件的現有瀏覽器階段作業啟動的，使用者必須再次登入 Smart View，因為它不會共用現有作業階段的 Cookie。
- 如果您要使用自訂的 HTML 式登入表單，而非預設的 Oracle Access Manager 登入表單，請確保該自訂表單的來源包括 loginform 字串。若要讓 Smart View 與 Oracle Access Manager 的整合能夠運作，這個字串是必要的。

4

設定使用者目錄

另請參閱：

- [使用者目錄和 EPM System 安全性](#)
- [使用者目錄組態的相關作業](#)
- [Oracle Identity Manager 與 EPM System](#)
- [Active Directory 資訊](#)
- [設定 OID、Active Directory 及其他 LDAP 型的使用者目錄](#)
- [將關聯式資料庫設定為使用者目錄](#)
- [測試使用者目錄連線](#)
- [編輯使用者目錄設定](#)
- [刪除使用者目錄組態](#)
- [管理使用者目錄搜尋順序](#)
- [設定安全性選項](#)
- [重新產生加密金鑰](#)
- [使用特殊字元](#)

使用者目錄和 EPM System 安全性

許多使用者和識別管理系統均支援 Oracle Enterprise Performance Management System 產品，而這些系統統稱為使用者目錄。其中包括已啟用「輕量型目錄存取通訊協定」(LDAP) 的使用者目錄，例如 Sun Java System Directory Server (前身為 SunONE Directory Server) 和 Active Directory。EPM System 也支援 Rational 資料庫作為外部使用者目錄。

一般而言，EPM System 產品在提供時會使用原生目錄和外部使用者目錄。如需受支援的使用者目錄清單，請參閱 [Oracle Enterprise Performance Management System Certification Matrix \(僅英文版\)](#)。

EPM System 產品要求存取產品的每個使用者都要有一個使用者目錄帳戶。您可以將這些使用者指派到群組中以協助提供。還可以提供 EPM System 角色與物件 ACL 給使用者和群組。基於管理支出的考量，Oracle 並不建議提供這些項目給個別使用者。所有經過設定之使用者目錄中的使用者和群組會顯示在 Oracle Hyperion Shared Services Console 中。

根據預設，EPM System Configurator 會把 Shared Services 儲存庫設定成原生目錄，以便支援 EPM System 產品。目錄管理員會使用 Shared Services Console 存取和管理原生目錄。

使用者目錄組態的相關作業

若要支援 SSO 與授權，系統管理員必須設定外部使用者目錄。系統管理員可以在 Oracle Hyperion Shared Services Console 中，執行與使用者目錄的設定與管理相關的多種工作。下列主題提供相關指示：

- 設定使用者目錄：
 - 設定 OID、Active Directory 及其他 LDAP 型的使用者目錄
 - 將關聯式資料庫設定為使用者目錄
- 測試使用者目錄連線
- 編輯使用者目錄設定
- 刪除使用者目錄組態
- 管理使用者目錄搜尋順序
- 設定安全性選項

Oracle Identity Manager 與 EPM System

Oracle Identity Manager 是一個角色與使用者管理解決方案，可自動處理新增、更新及刪除跨企業資源的使用者帳戶與屬性層級權益。Oracle Identity Manager 可作為獨立的產品或是作為 Oracle Identity and Access Management Suite Plus 的一部分。

Oracle Enterprise Performance Management System 使用企業角色 (LDAP 群組)，與 Oracle Identity Manager 整合。EPM System 元件的角色可以指派給企業角色。加入 Oracle Identity Manager 企業角色的使用者或群組會自動繼承指派的 EPM System 角色。

例如，假設您具有名為 *Budget Planning* 的 Oracle Hyperion Planning 應用程式。若要支援此應用程式，您可以在 Oracle Identity Manager 中建立三個企業角色—預算規劃互動使用者、預算規劃一般使用者及預算規劃管理員。當您提供 EPM System 角色時，請確保「佈建管理員」提供來自 Oracle Identity Manager 的企業角色，與來自 *Budget Planning* 及其他 EPM System 元件 (包含 Shared Services) 的必要角色。指派給 Oracle Identity Manager 之企業角色的所有使用者與群組都會繼承 EPM System 角色。如需部署與管理 Oracle Identity Manager 的相關資訊，請參閱 Oracle Identity Manager 文件。

若要將 Oracle Identity Manager 與 EPM System 進行整合，管理員必須執行以下步驟：

- 請確保將會在 EPM System 提供作業中使用之 Oracle Identity Manager 企業角色的成員 (使用者與群組)，已在啟用 LDAP 之使用者目錄 (例如 OID 或 Active Directory) 中定義。
- 將啟用 LDAP 的使用者目錄 (也就是定義企業角色成員的位置) 設定為 EPM System 中的外部使用者目錄。請參閱設定 OID、Active Directory 及其他 LDAP 型的使用者目錄。

Active Directory 資訊

本節說明在本文件中使用的 Microsoft Active Directory 概念。

DNS 查閱與主機名稱查閱

系統管理員可以設定 Active Directory，讓 Oracle Hyperion Shared Services 可以執行靜態主機名稱查閱或 DNS 查閱來尋找 Active Directory。靜態主機名稱查閱並不支援 Active Directory 容錯移轉。

使用 DNS 查閱可讓您在多個網域控制器上設定 Active Directory 來確保高可用性的情況下，能夠確保 Active Directory 的高可用性。設定為執行 DNS 查閱時，Shared Services 會查詢 DNS 伺服器，找出已登錄的網域控制器，然後再連線至優先順序最高的網域控制器。若與 Shared Services 連線的網域控制器無法運作，Shared Services 就會動態切換至下一個優先順序最高的可使用網域控制器。

備註：

您只能在擁有支援容錯移轉的備援 Active Directory 設定時，才可以設定 DNS 查閱。如需相關資訊，請參閱 Microsoft 文件。

全域目錄

全域目錄就是可將所有 Active Directory 物件副本儲存在單一樹系中的網域控制器。其可在目錄中儲存主機網域之所有物件的完整副本，並可在樹系中儲存其他所有網域之所有物件的局部副本，以用於一般使用者搜尋作業。如需設定全域目錄的相關資訊，請參閱 Microsoft 文件。

若您的組織使用全域目錄，請使用下列其中一種方法設定 Active Directory：

- 將全域目錄伺服器設定為外部使用者目錄 (建議使用)。
- 將每個 Active Directory 網域設定為不同的外部使用者目錄。

當您設定全域目錄，而非設定個別 Active Directory 網域時，就能讓 Oracle Enterprise Performance Management System 產品存取樹系內的本機和通用群組。

設定 OID、Active Directory 及其他 LDAP 型的使用者目錄

系統管理員會使用本節所述的程序來設定 LDAP 型的公司使用者目錄，例如 OID、Sun Java System Directory Server、Oracle Virtual Directory、Active Directory、IBM Tivoli Directory Server，或是未列於組態畫面中的 LDAP 型使用者目錄。

如何設定 OID、Active Directory 及其他的 LDAP 型使用者目錄。

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理及設定使用者目錄**。
提供者組態頁籤會隨即開啟。此畫面會列出所有已設定的使用者目錄，包括原生目錄。
3. 按一下**新增**。

4. 在**目錄類型**下，選取下列一個選項：
 - **輕量型目錄存取通訊協定 (LDAP)**，以便設定非 Active Directory 的 LDAP 型使用者目錄。選取此選項以設定 Oracle Virtual Directory。
 - **Microsoft Active Directory (MSAD)**，以便設定 Active Directory。

僅限 Active Directory 和 Active Directory 應用程式模式 (ADAM)：如果您想要搭配 Active Directory 或 ADAM 使用自訂 ID 屬性 (ObjectGUID 以外的屬性；例如 sAMAccountName)，請選取**輕量型目錄存取通訊協定 (LDAP)**，並將其「目錄類型」設為其他。

5. 按一下**下一步**。

The screenshot shows the 'Configure User Directories' wizard in the Oracle Enterprise Performance Management System. The current step is '1. MSAD Connection Information'. The interface includes a left-hand navigation pane with 'User Directories' selected. The main area contains several sections: 'Server Information' with fields for Name, Host Name, Port (389), Base DN, ID Attribute (objectguid), Maximum Size (0), Trusted (checked), Anonymous Bind, User DN, and Password; 'LDAP Options' with Referrals (ignore), Dereference Aliases (Always), and Connection Read Timeout (60 sec); 'Connection Pooling' with Max Connections (100), Timeout (300000 ms), Evict Interval (120 mins), Allowed Idle Connection Time (120 mins), and Grow Connections (checked); and 'Custom Module' with 'Enable Custom Authentication Module' (unchecked). A 'Fetch DNS' button is located next to the Base DN field. At the bottom, there are 'Help', 'Back', 'Next', 'Finish', and 'Cancel' buttons.

6. 輸入必要的參數。

表格 4-1 連線資訊畫面

標籤	說明
目錄伺服器	<p>選取使用者目錄。ID 屬性值會變更為所選產品的建議常數唯一識別屬性。如果您在步驟 4 選取 Active Directory，系統會自動選取此特性。</p> <p>在下列案例中選取 Other：</p> <ul style="list-style-type: none"> 您設定了未列出的使用者目錄類型；例如 Oracle Virtual Directory 您設定了列出之已啟用 LDAP 的使用者目錄 (例如 OID)，但想要使用自訂 ID 屬性。 您設定了 Active Directory 或 ADAM，以便使用自訂 ID 屬性。
	<p>備註：</p> <p>由於 Oracle Virtual Directory 會以單一目錄檢視的方式提供 LDAP 目錄與 RDMBS 資料儲存庫的虛擬抽象概念，因此 Oracle Enterprise Performance Management System 會把該目錄檢視認定為單一外部使用者目錄，不管 Oracle Virtual Directory 支援了多少使用者目錄數量和類型。</p>
名稱	<p>範例： Oracle Internet Directory</p> <p>使用者目錄的描述性名稱。如有設定多個使用者目錄，即可以此識別特定使用者目錄。名稱不應該包含空格和底線以外的特殊字元。</p> <p>範例： Corporate_OID</p>
DNS 查閱	<p>僅限 Active Directory： 選取此選項可啟用 DNS 查閱。請參閱 DNS 查閱與主機名稱查閱。Oracle 建議您，將 DNS 查閱設定為連線到生產環境中 Active Directory 的方法，以便防止連線失敗。</p>
	<p>備註：</p> <p>若要設定全域目錄，請勿選取此選項。</p>
	<p>選取此選項時，即會顯示以下欄位：</p> <ul style="list-style-type: none"> 網域： Active Directory 樹系的網域名稱。 範例： example.com 或 us.example.com AD 網站： Active Directory 網站名稱，通常就是儲存在 Active Directory 組態容器中之網站的相對識別名稱。一般來說，「AD 網站」可識別地理位置，例如城市、州/省、區域或國家/地區。 範例： Santa Clara 或 US_West_region DNS 伺服器： 可支援 DNS 伺服器查閱網域控制器的伺服器 DNS 名稱。
主機名稱	<p>僅限 Active Directory： 選取此選項可啟用靜態主機名稱查閱。請參閱 DNS 查閱與主機名稱查閱。</p>
	<p>備註：</p> <p>如果您正在設定 Active Directory 全域目錄，請選取此選項。</p>

表格 4-1 (續) 連線資訊畫面

標籤	說明
主機名稱	<p>使用者目錄伺服器的 DNS 名稱。如果使用者目錄是用來在 SiteMinder 支援 SSO，請使用完整的網域名稱。Oracle 建議您，只在目的為測試時才使用主機名稱建立 Active Directory 連線。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> 備註：</p> <p>如果您正在設定 Active Directory 全域目錄，請指定全域目錄伺服器的主機名稱。請參閱全域目錄。</p> </div> <p>範例： MyServer</p>
連接埠	<p>執行使用者目錄所在的連接埠號碼。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> 備註：</p> <p>如果您正在設定 Active Directory 全域目錄，請指定全域目錄伺服器所使用的連接埠 (預設值為 3268)。請參閱全域目錄。</p> </div> <p>範例： 389</p>
SSL 已啟用	<p>此核取方塊可讓與此使用者目錄的通訊更安全。使用者目錄必須設定為使用安全通訊。</p>
基礎 DN	<p>使用者與群組搜尋作業之起始節點的識別名稱 (DN)。您也可以使用檢索 DN 按鈕，列出可用的基礎 DN，然後再從清單中選取適當的基礎 DN。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> 備註：</p> <p>若要設定全域目錄，請指定樹系的基礎 DN。</p> </div> <p>如需特殊字元的使用限制，請參閱使用特殊字元。</p> <p>Oracle 建議您選取包含所有 EPM System 產品使用者與群組的最低 DN。</p> <p>範例： dc=example,dc=com</p>
ID 屬性	<p>需選取目錄類型中的 Other，才可以修改屬性值。此屬性必須是存在目錄伺服器之使用者和群組物件中的通用屬性。</p> <p>系統會自動針對 OID orclguid、SunONE (nsuniqueid)、IBM Directory Server (Ibm-entryUuid)、Novell eDirectory (GUID)，以及 Active Directory (ObjectGUID) 設定這個屬性的建議值。</p> <p>範例： orclguid</p> <p>如果您在目錄伺服器中選擇 Other 之後手動設定 ID 屬性值；例如若要設定 Oracle Virtual Directory，則 ID 屬性值應：</p> <ul style="list-style-type: none"> · 指向唯一屬性 · 沒有具體位置 · 不會隨著時間改變

表格 4-1 (續) 連線資訊畫面

標籤	說明
大小上限	<p>搜尋所能傳回的結果數上限。若此值大於使用者目錄設定所支援的值，即會以使用者目錄值覆寫此值。</p> <p>針對非 Active Directory 的使用者目錄，請將此欄位留白，以擷取所有符合搜尋準則的使用者和群組。</p> <p>針對 Active Directory，請將此值設定為 0，以擷取所有符合搜尋準則的使用者和群組。</p> <p>如果您正在使用「委派管理」模式來設定 Oracle Hyperion Shared Services，請將此值設定為 0。</p>
受信任	<p>此核取方塊可用於指出此提供者為信任 SSO 來源。受信任來源所提供的 SSO 憑證不包含使用者的密碼。</p>
匿名繫結	<p>此核取方塊可用於指出 Shared Services 可以匿名方式繫結至使用者目錄，藉以搜尋使用者與群組。僅當使用者目錄允許匿名繫結時，才可加以使用。若未選取此選項，必須在「使用者 DN」中指定具有足夠權限搜尋使用者資訊之儲存所在目錄的帳戶。</p> <p>Oracle 建議不要使用匿名繫結。</p>
 備註： OID 不支援匿名繫結。	
使用者 DN	<p>如有選取匿名繫結，將會停用此選項。</p> <p>Shared Services 用以繫結使用者目錄的使用者識別名稱。這位使用者必須具有在 DN 內搜尋 RDN 屬性的權限。例如，在 <code>dn: cn=John Doe, ou=people, dc=myCompany, dc=com</code> 中，繫結使用者必須具有 <code>cn</code> 屬性的搜尋權限。</p> <p>「使用者網域」中的特殊字元必須使用逸出字元來指定。如需使用限制，請參閱使用特殊字元。</p> <p>範例： <code>cn=admin,dc=myCompany,dc=com</code></p>
附加基礎 DN	<p>此核取方塊可用於將基礎 DN 附加至「使用者 DN」。若要將「目錄管理員」帳戶用為「使用者 DN」，請勿附加「基礎 DN」。</p> <p>如有選取「匿名繫結」選項，將會停用此核取方塊。</p>
密碼	<p>使用者 DN 的密碼</p> <p>如有選取「匿名繫結」選項，將會停用此方塊。</p> <p>範例： <code>UserDNpassword</code></p>
顯示進階選項轉介	<p>此核取方塊可顯示進階選項。</p> <p>僅限 Active Directory：</p> <p>如果您已將 Active Directory 設定為會追蹤轉介，請選取 <code>follow</code> 以自動追蹤 LDAP 轉介。選取 <code>ignore</code> 將不會使用轉介。</p>
解除參照別名	<p>選取 Shared Services 搜尋用以解除參照使用者目錄中之別名，以擷取別名 DN 所指向之物件的方法。請選取：</p> <ul style="list-style-type: none"> · 自動：一律自動解除參照別名。 · 永不：永不解除參照別名。 · 尋找中：僅在解析名稱時解除參照別名。 · 搜尋中：僅在解析名稱後解除參照別名。
連線讀取逾時	<p>若沒有得到回應，LDAP 提供者中止 LDAP 讀取嘗試的間隔(秒)。</p> <p>預設值： 60 秒</p>

表格 4-1 (續) 連線資訊畫面

標籤	說明
連線數上限	連線集區中的連線數上限。LDAP 型目錄 (包括 Active Directory) 的預設值為 100。 預設值：100
逾時	取得集區連線的逾時。在此期間後，將會擲出例外。 預設值：300000 毫秒 (5 分鐘)
收回間隔	選擇性： 執行收回程序以清除集區的時間。收回程序會移除超過允許的閒置連線時間的閒置連線。 預設值：120 分鐘
允許的閒置連線時間	選擇性： 收回程序從集區中移除閒置連線前的經歷時間。 預設值：120 分鐘
增加連線	此選項可指定連線集區的大小，是否可以擴展大於連線上限。預設為已選取。若不允許連線集區擴展，系統將會在無法於 Time Out 設定的時間內取得連線時傳回錯誤。
啟用自訂驗證模組	此核取方塊可讓您使用自訂驗證模組，以驗證此使用者目錄中所定義的使用者。您也必須在「安全性選項」畫面中，輸入完全符合條件之驗證模組的 Java 類別名稱。請參閱 設定安全性選項 。 自訂驗證模組驗證對於精簡型用戶端與豐富型用戶端是透明的，而且不需變用戶端部署。請參閱 <i>Oracle Enterprise Performance Management System Security Configuration Guide (僅英文版)</i> 中的「使用自訂驗證模組」。

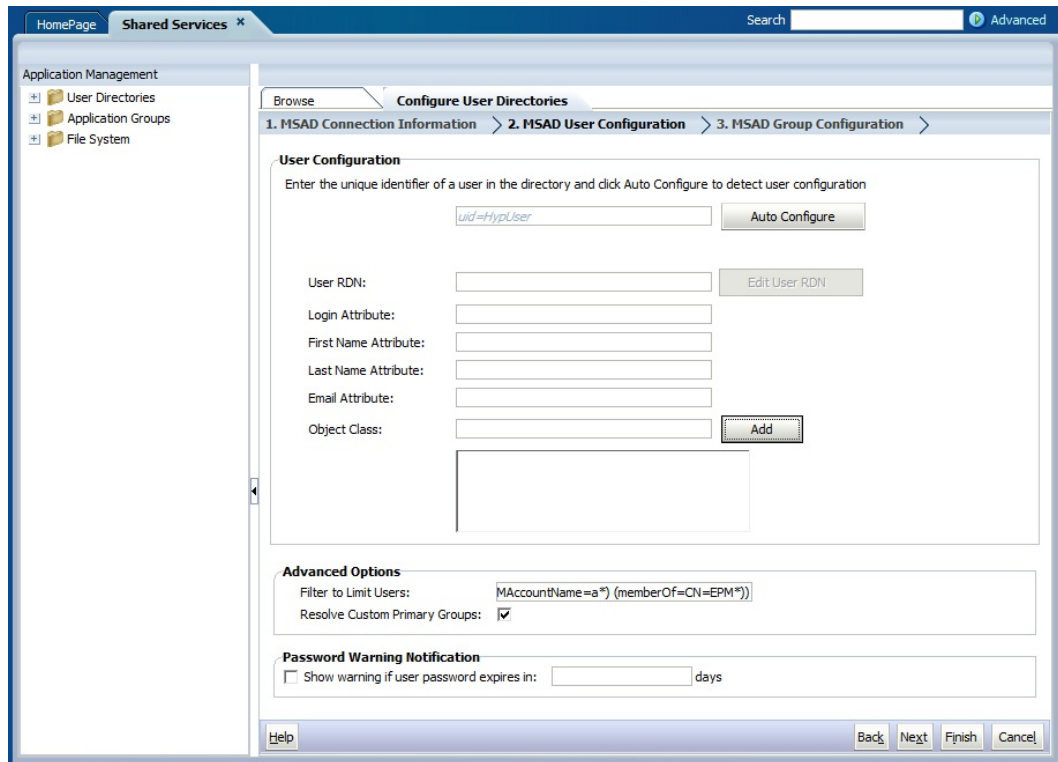
7. 按一下**下一步**。

Shared Services 會使用在「使用者組態」畫面上所設定的特性來建立使用者 URL，以決定搜尋使用者的起始節點。使用此 URL 可加快搜尋速度。

▲ 注意：

使用者 URL 不應指向別名。EPM System 安全性要求使用者 URL 指向實際使用者。

Oracle 建議您利用畫面的「自動設定」區域擷取必要資訊。



備註：

如需可在使用者組態中使用的特殊字元清單，請參閱[使用特殊字元](#)。

- 在**自動設定**中，輸入 `attribute=identifier` 格式的唯一使用者識別碼，例如 `uid=jdoe`。使用者的屬性會顯示在「使用者組態」區域中。

如果您正在設定 OID，就無法自動設定使用者篩選條件，原因是 OID 的根 DSE 並不包含「命名環境定義」屬性中的項目。請參閱 *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* 中的[管理命名環境定義](#)。

備註：

您可以在「使用者組態」區域的文字方塊中，手動輸入必要的使用者屬性。

表格 4-2 使用者組態畫面

標籤	說明 ¹
使用者 RDN	使用者的相對 DN。DN 的各項元件稱為 RDN，代表樹狀目錄中的分支。使用者的 RDN 通常等同於 uid 或 cn。 如需使用限制，請參閱 使用特殊字元 。 範例： ou=People

表格 4-2 (續) 使用者組態畫面

標籤	說明 ¹
登入屬性	<p>唯一屬性 (可以是自訂屬性) 可儲存使用者的登入名稱。使用者在登入 EPM System 產品時，會以此屬性的值作為使用者名稱。使用者 ID (登入屬性值) 必須在所有使用者目錄中是唯一的。例如，您可以分別把 uid 和 sAMAccountName 當做 SunONE 和 Active Directory 組態的登入屬性。這些屬性的值在所有使用者目錄 (包括原生目錄) 中都必須是唯一的。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> 備註： 使用者 ID 不會區分大小寫。</p> </div> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> 備註： 如果您正在把 OID 設定為 Kerberos 環境中的 Oracle Application Server 上所部署 EPM System 產品的外部使用者目錄，就必須把此特性設定為 userPrincipalName。</p> </div> <p>預設值</p> <ul style="list-style-type: none"> · Active Directory： cn · Active Directory 以外的 LDAP 目錄： uid
名字屬性	<p>此屬性可儲存使用者的名字 預設值： givenName</p>
姓氏屬性	<p>此屬性可儲存使用者的姓氏 預設值： sn</p>
電子郵件屬性	<p>選擇性： 此屬性可儲存使用者電子郵件地址 預設值： mail</p>

表格 4-2 (續) 使用者組態畫面

標籤	說明 ¹
物件類別	<p>使用者的物件類別 (可與使用者相關聯的強制性屬性與選擇性屬性)。Shared Services 可在搜尋篩選中使用此畫面中所列的物件類別。Shared Services 應可藉由這些物件類別，找出所有應獲提供的使用者。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin: 10px 0;"> <p> 備註：</p> <p>如果您正在將 Active Directory 或 ADAM 的使用者目錄類型設定為 Other，以便使用自訂 ID 屬性，您就必須將此值設定為 user。</p> </div> <p>如有需要，可以手動新增物件類別。若要新增物件類別，請在物件類別方塊中輸入物件類別名稱，然後按一下新增。</p> <p>若要刪除物件類別，請選取物件類別，然後按一下移除。</p> <p>預設值</p> <ul style="list-style-type: none"> · Active Directory： user · Active Directory 以外的 LDAP 目錄： person, organizationalPerson, inetorgperson
篩選至有限的使用者	<p>只會擷取授予 EPM System 產品角色之使用者的 LDAP 查詢。例如，LDAP 查詢 (uid=Hyp*) 只會擷取名稱開頭為 Hyp 的使用者。</p> <p>「使用者組態」畫面會驗證「使用者 RDN」，並在必要時建議使用使用者篩選。</p> <p>使用者篩選條件會限制查詢期間所傳回的使用者數量。當使用者 RDN 所指向的節點，包含許多無需提供的使用者時，此篩選更形重要。使用者篩選可以設計成排除不需提供的使用者，以提升效能。</p>
多重屬性 RDN 的使用者搜尋屬性	<p>僅限 Active Directory 以外已啟用 LDAP 的使用者目錄：請只在您目錄伺服器已設定為使用多重屬性 RDN 時，才設定此值。您設定的值必須為 RDN 屬性中的其中一個值。您指定的屬性值應該是唯一的值，且屬性應該是可搜尋的屬性。</p> <p>例如，假設您已將 SunONE 目錄伺服器設定為結合 cn (cn=John Doe) 和 uid (uid=jDoe12345) 屬性，以建立和下文相似的多重屬性 RDN：</p> <pre>cn=John Doe+uid=jDoe12345, ou=people, dc=myCompany, dc=com</pre> <p>在這種情況下，當 cn 或 uid 屬性滿足下列條件時，您就可以使用其中一個屬性：</p> <ul style="list-style-type: none"> · 在「連線資訊」頁籤之「使用者 DN」欄位中指出的使用者能搜尋到屬性 · 屬性要求您設定在整個使用者目錄中唯一的值
解析自訂主要群組	<p>僅限 Active Directory：此核取方塊指出是否要識別主要使用者群組以判斷有效角色。依預設，此核取方塊為選取狀態。Oracle 建議您不要變更此設定。</p>
如果使用者密碼在下列天數內到期，則會出現警示：	<p>僅限 Active Directory：此核取方塊指出如果 Active Directory 使用者密碼在指定天數內到期，是否要顯示警告訊息。</p>

¹ 針對可選的組態值，EPM System 安全性會對某些欄位使用預設值。如果您未在欄位中輸入值，會在執行時期中使用預設值。

9. 按一下**下一步**。

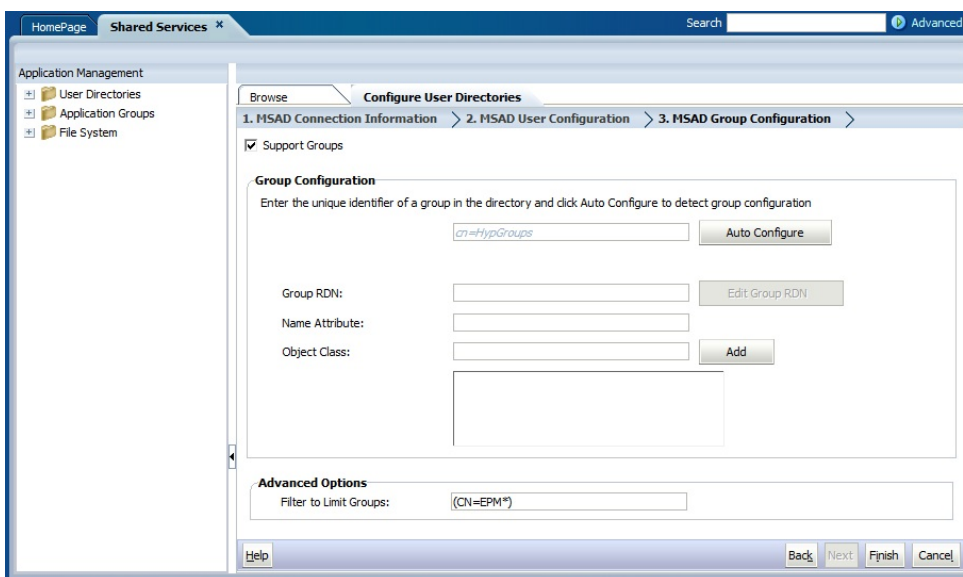
「群組組態」畫面會隨即開啟。Shared Services 會使用此畫面中所設定的特性建立群組 URL，以決定搜尋群組的起始節點。使用此 URL 可加快搜尋速度。

注意：

群組 URL 不應指向別名。EPM System 安全性要求群組 URL 指向實際群組。如果您要設定使用群組別名的 Novell eDirectory，則群組 URL 中的群組別名和群組帳戶必須可用。

備註：

「群組組態」畫面中的資料項目屬於可選。若未輸入群組 URL 設定，Shared Services 即會從基礎 DN 中搜尋群組，對效能造成負面的影響，這在使用者目錄中包含許多群組時尤其明顯。



10. 若您的組織不打算提供群組，或未將使用者分類到使用者目錄的群組中，請清除**支援群組**。清除此選項會停用此畫面中的欄位。

若要支援群組，Oracle 建議您使用自動設定功能擷取必要的資訊。

如果您正在將 OID 設定為使用者目錄，就無法使用自動設定功能，原因是 OID 的根 DSE 並不包含「命名環境定義」屬性中的項目。請參閱 *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* 中的**管理命名環境定義**。

11. 在**自動設定**文字方塊中，輸入唯一群組識別碼，然後按一下**執行**。

群組識別碼必須以 `attribute=identifier` 的格式表示，如 `cn=western_region`。

群組的屬性會顯示在「群組組態」區域中。

 **備註：**

您可以在「群組組態」文字方塊中，輸入必要的群組屬性。

 **注意：**

若未針對節點名稱包含 / (正斜線) 或 \ (反斜線) 的使用者目錄中設定群組 URL，使用者與群組的搜尋作業將會失敗。例如，如果您沒有針對節點 (例如 OU=child\ou, OU=parent/ou 或 OU=child/ou, OU=parent \ ou) 中使用者和群組所在的使用者目錄指定群組 URL，則所有用於列出使用者或群組的作業都會失敗。

表格 4-3 群組組態畫面

標籤	說明 ¹
群組 RDN	<p>群組的相對 DN。此值代表與基礎 DN 對應的路徑，用來作為群組 URL。</p> <p>指定可指出適合用於提供，並包含所有群組之最低使用者目錄節點的群組 RDN。</p> <p>如果您將 Active Directory 主要群組用於提供，請確保該主要群組隸屬於「群組 RDN」。如果位於群組 URL 的範圍之外，則 Shared Services 不會擷取主要群組。</p> <p>群組 RDN 對於登入與搜尋效能有很大的影響。其是所有群組搜尋的起點，因此必須指出包含所有 EPM System 產品群組的最低可能節點。為確保最佳效能，群組 RDN 內的群組數不應超過 10,000 個。若實際的群組超過此數，請使用群組篩選擷取您要提供的群組。</p>
名稱屬性	<p>此屬性可儲存群組的名稱</p> <p>預設值</p> <ul style="list-style-type: none"> · 包含 Active Directory 的 LDAP 目錄： cn · Native Directory： cssDisplayNameDefault

 **備註：**

若群組 URL 內的可用群組數超過 10,000 個，Shared Services 將會顯示警告。

如需使用限制，請參閱[使用特殊字元](#)。

範例： ou=Groups

表格 4-3 (續) 群組組態畫面

標籤	說明 ¹
物件類別	<p>群組的物件類別。Shared Services 可在搜尋篩選中使用此畫面中所列的物件類別。使用這些物件類別應可讓 Shared Services 找到所有與使用者相關聯的群組。</p> <div style="border: 1px solid #0070C0; padding: 10px; margin: 10px 0;"> <p> 備註：</p> <p>如果您正在將 Active Directory 或 ADAM 的使用者目錄類型設定為 Other，以便使用自訂 ID 屬性，您就必須將此值設定為 <code>group?member</code>。</p> </div> <p>如有需要，可以手動新增物件類別。若要新增物件類別，請在「物件類別」文字方塊中輸入物件類別名稱，然後按一下新增。 若要刪除物件類別，請選取物件類別，然後按一下移除。</p> <p>預設值</p> <ul style="list-style-type: none"> · Active Directory：group?member · Active Directory 以外的 LDAP 目錄： groupofuniquenames?uniquemember, groupOfNames?member · Native Directory：groupofuniquenames?uniquemember, cssGroupExtend?cssIsActive
篩選至有限的群組	<p>只會擷取將提供 EPM System 產品角色之群組的 LDAP 查詢。例如，LDAP 查詢 (<code> (cn=Hyp*) (cn=Admin*)</code>) 只會擷取名稱開頭為 Hyp 或 Admin 的群組。</p> <p>群組篩選可用以限制查詢期間傳回的群組數。群組 RDN 所識別的節點包含許多不需提供的群組時，此篩選更形重要。此篩選經設計後可排除不需提供的群組，以提升效能。</p> <p>如果您將 Active Directory 主要群組用於提供，請確保您設定的任何群組篩選條件都可擷取位於該群組 URL 範圍內的主要群組。例如，篩選條件 (<code> (cn=Hyp*) (cn=Domain Users)</code>) 會擷取名稱開頭為 Hyp 的群組，以及名為 Domain Users 的主要群組。</p>

¹ 針對可選的組態值，EPM System 安全性會對某些欄位使用預設值。如果您未在欄位中輸入值，會在執行時期中使用預設值。

12. 按一下完成。

Shared Services 會儲存組態，並返回「已定義的使用者目錄」畫面，而畫面中此時會列出您所設定的使用者目錄。

13. 測試組態。請參閱[測試使用者目錄連線](#)。

14. 視需要變更搜尋順序指派。請參閱[管理使用者目錄搜尋順序](#)以取得詳細資料。

15. 視需要指定安全性選項。請參閱[設定安全性選項](#)以取得詳細資料。

16. 重新啟動 Oracle Hyperion Foundation Services 和其他 EPM System 元件。

將關聯式資料庫設定為使用者目錄

Oracle、SQL Server 及 IBM DB2 關聯式資料庫中的使用者與群組資訊可用於支援提供。若您無法從資料庫的系統架構取得群組資訊，Oracle Hyperion Shared Services 就無法支援從該資料庫提供者提供群組的功能。例如 Shared Services 無法從舊版的 IBM DB2 擷取群組資訊，因為資料庫會使用作業系統上所定義的群組。然而，「佈建管理員」可以將這些使用者加入原生目錄中的群組，然後再提供這些群組。如需受支援的平台資訊，請參閱張貼在 Oracle Technology Network (OTN) 的支援 [Oracle Fusion Middleware 的系統組態](#) 頁面上的 [Oracle Enterprise Performance Management System Certification Matrix \(僅英文版\)](#)。

備註：

如果您使用 DB2 資料庫，使用者名稱至少需包含 8 個字元。使用者名稱不應超過 256 個字元 (Oracle 與 SQL Server 資料庫) 以及 1000 個字元 (DB2)。

設定 Shared Services 才能以資料庫管理員的身分 (如 Oracle SYSTEM 使用者) 連線至資料庫，以擷取使用者與群組的清單。

備註：

Shared Services 只可以擷取作用中資料庫使用者進行提供。非作用中與鎖定的資料庫使用者帳戶會予以忽略。

若要設定資料庫提供者：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理**及**設定使用者目錄**。
3. 按一下**新增**。
4. 在**目錄類型**畫面中，選取**關聯式資料庫 (Oracle、DB2、SQL Server)**。
5. 按一下**下一步**。

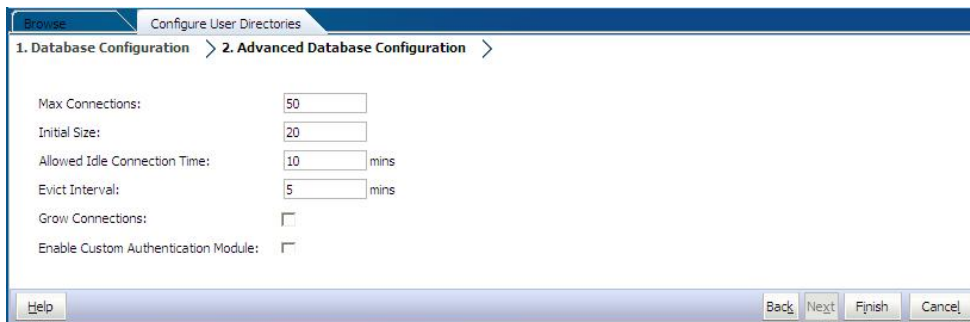
The screenshot shows a software configuration window titled "Configure User Directories" with a sub-tab "Advanced Database Configuration". The "Database Type" is set to "Oracle". There are several input fields: "Name", "Server", "Port" (with "1521" entered), "Service/SID", "User Name", and "Password". A "Trusted" checkbox is checked. At the bottom, there are buttons for "Help", "Back", "Next", "Finish", and "Cancel".

6. 在「資料庫組態」標籤上，輸入組態參數。

表格 4-4 資料庫組態頁籤

標籤	說明
資料庫類型	關聯式資料庫提供者。 Shared Services 只支援把 Oracle 及 SQL Server 資料庫當做資料庫提供者。 範例 ：Oracle
名稱	資料庫提供者的唯一組態名稱。 範例 ：Oracle_DB_FINANCE
伺服器	正在執行資料庫伺服器之電腦的 DNS 名稱。 範例 ：myserver
連接埠	資料庫伺服器連接埠號碼 範例 ：1521
服務/SID (僅限 Oracle)	系統識別碼 (預設值為 orcl)。 範例 ：orcl
資料庫 (僅限 SQL Server 與 DB2)	Shared Services 應連線的資料庫 範例 ：master
使用者名稱	Shared Services 用於存取資料庫的使用者名稱。此資料庫使用者必須具有資料庫系統表格的存取權限。 Oracle 建議您，在 Oracle 資料庫使用 system 帳戶，而在 SQL Server 資料庫則使用資料庫管理員的使用者名稱。 範例 ：SYSTEM
密碼	使用者名稱 中識別之使用者的密碼。 範例 ：system_password
受信任	此核取方塊可用於指定此提供者為受信任的 SSO 來源。受信任來源所提供的 SSO 憑證不包含使用者的密碼。

7. **選擇性**：按**下一步**，以設定連接集區。
「進階資料庫組態」標籤會隨即開啟。



8. 在「進階資料庫組態」中，輸入連線集區參數。

表格 4-5 進階資料庫組態標籤

標籤	說明
連線數上限	集區中的連線數上限。預設值為 50。
初始大小	初始化集區時的可用連線數。預設值為 20。
允許的閒置連線時間	選擇性： 收回程序從集區中移除閒置連線前的經歷時間。預設值為 10 分鐘。
收回間隔	選擇性： 執行收回程序以清除集區的時間。收回程序會移除超過 Allowed Idle Connection Time 的閒置連線。預設值為五分鐘。
增加連線	指定連線集區能否超過 Max Connections。依預設會清除此選項，表示集區無法擴展。若不允許連線集區擴展，系統將會在無法於 Time Out 設定的時間內取得連線時傳回錯誤。
啟用自訂驗證模組	此核取方塊可讓您使用自訂驗證模組，以驗證此使用者目錄中所定義的使用者。您也必須在「安全性選項」畫面中，輸入完全符合條件之驗證模組的 Java 類別名稱。請參閱 設定安全性選項 。 自訂驗證模組驗證對精簡型和豐裕型用戶端是通透的。請參閱 <i>Oracle Enterprise Performance Management System Security Configuration Guide (僅英文版)</i> 中的「使用自訂驗證模組」。

9. 按一下**完成**。
10. 按一下**確定**，以返回「已定義的使用者目錄」畫面。
11. 測試資料庫提供者組態。請參閱[測試使用者目錄連線](#)。
12. 視需要變更搜尋順序指派。請參閱[管理使用者目錄搜尋順序](#)以取得詳細資料。
13. 視需要指定安全性設定。請參閱[設定安全性選項](#)。
14. 重新啟動 Oracle Hyperion Foundation Services 及其他 Oracle Enterprise Performance Management System 元件。

測試使用者目錄連線

當您設定使用者目錄之後，請測試連線，以確保 Oracle Hyperion Shared Services 可使用目前的設定連線至該使用者目錄。

若要測試使用者目錄連線：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理及設定使用者目錄**。
3. 從使用者目錄清單中，選取要測試的外部使用者目錄組態。
4. 按一下**測試**，然後按一下**確定**。

編輯使用者目錄設定

管理員可以修改使用者目錄組態的任何參數 (名稱除外)。Oracle 建議您不要編輯已用於提供之使用者目錄的組態資料。

 **注意：**

編輯使用者目錄組態中的某些設定 (例如 ID 屬性) 會使提供資料無效。因此，修改已提供之使用者目錄的設定時，請格外小心。

若要編輯使用者目錄組態：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理及設定使用者目錄**。
3. 選取要編輯的使用者目錄。
4. 按一下**編輯**。
5. 修改組態設定。

 **備註：**

您無法修改組態名稱。若要修改 LDAP 使用者目錄組態，可以從「目錄伺服器」清單中選擇不同的目錄伺服器或 Other (適用於自訂 LDAP 目錄)。您無法編輯原生目錄參數。

如需您可編輯之參數的說明，請參閱下列表格：

- Active Directory 和其他 LDAP 型使用者目錄，請參閱[設定 OID、Active Directory 及其他 LDAP 型的使用者目錄](#)中的表格。
 - 資料庫：請參閱[將關聯式資料庫設定為使用者目錄](#)中的表格
6. 按一下**確定**，儲存變更。

刪除使用者目錄組態

系統管理員可以隨時刪除外部使用者目錄組態。刪除組態會使衍生自此使用者目錄之使用者與群組的所有提供資訊無效，並從搜尋順序中移除此目錄。

 **提示：**

若不想使用經過設定並已用於提供的使用者目錄，請將此目錄從搜尋順序中移除；如此一來，即不會在其中搜尋使用者與群組。此動作可維護提供資訊的完整性，並讓您之後能夠使用此使用者目錄。

若要刪除使用者目錄組態：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理及設定使用者目錄**。

3. 選取某個目錄。
4. 按一下**刪除**。
5. 按一下**確定**。
6. 按一下**確定**。
7. 重新啟動 Oracle Hyperion Foundation Services 及其他 Oracle Enterprise Performance Management System 元件。

管理使用者目錄搜尋順序

當系統管理員設定外部使用者目錄時，Oracle Hyperion Shared Services 會自動將該使用者目錄新增至搜尋順序，然後把原生目錄的搜尋序列之前的下一個可用搜尋序列指派給該使用者目錄。Oracle Enterprise Performance Management System 在尋找使用者與群組時，會使用搜尋順序來依序搜尋每個已設定使用者目錄中的內容。

系統管理員可以從搜尋順序移除使用者目錄，在此情況下，Shared Services 會自動重新指派其餘目錄的搜尋順序。搜尋順序中不包含的使用者目錄不會用於支援驗證和提供。

備註：

Shared Services 在遇到指定帳戶時會終止對使用者或群組的搜尋。Oracle 建議將包含大多數 EPM System 使用者的公司目錄置於搜尋順序之首。

根據預設，原生目錄被設為搜尋順序中的最後一個目錄。管理員可以執行下列工作，以管理搜尋順序：

- [將使用者目錄加入搜尋順序](#)
- [變更搜尋順序](#)
- [移除搜尋順序指派](#)

將使用者目錄加入搜尋順序

新設定的使用者目錄會自動加入搜尋順序。若從搜尋順序中移除目錄，可以將此目錄加入搜尋順序的結尾。

若要將使用者目錄加入搜尋順序：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱 [啟動 Shared Services Console](#)。
2. 依序選取**管理及設定使用者目錄**。
3. 選取要新增為搜尋順序的已停用使用者目錄。
4. 按一下**包括**。
只有在您選取的使用者目錄不在搜尋順序時，才可以使用此按鈕。
5. 按一下**確定**，以返回「已定義的使用者目錄」畫面。
6. 重新啟動 Oracle Hyperion Foundation Services 和其他 EPM System 元件。

移除搜尋順序指派

從搜尋順序中移除使用者目錄，不會造成目錄組態失效，而會從驗證使用者時所搜尋的目錄清單移除使用者目錄。未包含在搜尋順序中的目錄狀態會設為已停用。當管理員從搜尋順序中移除使用者目錄時，會自動更新指派給其他使用者目錄的搜尋順序。



備註：

您無法移除搜尋順序中的原生目錄。

若要從搜尋順序中移除使用者目錄：

1. 以系統管理員的身分存取 Shared Services Console。請參閱 [啟動 Shared Services Console](#)。
2. 依序選取**管理**及**設定使用者目錄**。
3. 選取要從搜尋順序移除的目錄。
4. 按一下**排除**。
5. 按一下**確定**。
6. 在「目錄組態結果」畫面上，按一下**確定**。
7. 重新啟動 Foundation Services 和其他 EPM System 元件。

變更搜尋順序

指派給每個使用者目錄的預設搜尋順序是根據設定目錄的順序而定。根據預設，原生目錄被設為搜尋順序中的最後一個目錄。

若要變更搜尋順序：

1. 以系統管理員的身分存取 Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理**及**設定使用者目錄**。
3. 選取您想要變更搜尋順序的目錄。
4. 按一下**上移**或**下移**。
5. 按一下**確定**。
6. 重新啟動 Foundation Services、其他 EPM System 元件和使用 Shared Services 安全性 API 的自訂應用程式。

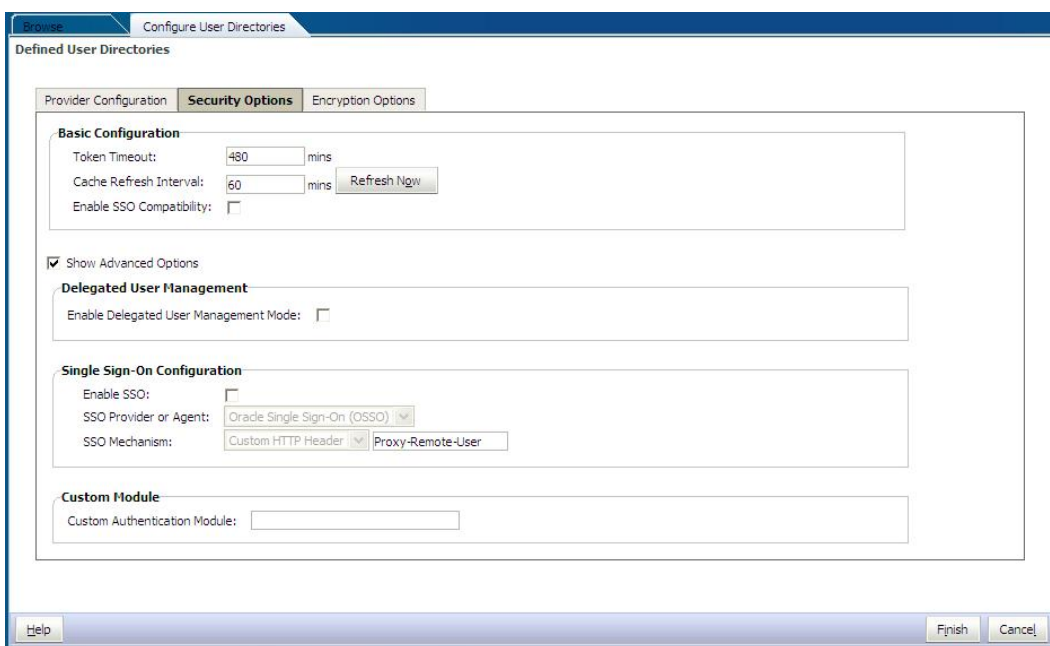
設定安全性選項

安全性選項包含搜尋順序中之所有使用者目錄適用的全域參數。


若要設定安全性選項：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。

2. 依序選取**管理**及**設定使用者目錄**。
3. 選取**安全性選項**。
4. 在**安全性選項**中設定全域參數。



表格 4-6 使用者目錄的安全性選項

參數	說明
憑證逾時	Oracle Enterprise Performance Management System 產品或 Web 識別管理解決方案所發出的 SSO 憑證，會在這個時間 (分鐘) 之後過期。使用者必須在此期間之後重新登入。憑證逾時是根據伺服器的系統時鐘設定。預設值為 480 分鐘。
	<p> 備註：</p> <p>憑證逾時與階段作業逾時不同。</p>
快取重新整理間隔	群組與使用者之間關係資料的 Oracle Hyperion Shared Services 快取之重新整理間隔時間 (分鐘)。預設值為 60 分鐘。 有關新外部使用者目錄群組，和在下一個快取重新整理之後新增至現有群組的新使用者的 Shared Services。透過新建立的外部使用者目錄群組規定的使用者在重新整理快取之前，不會取得其規定的角色。
立即重新整理	按一下此按鈕，以手動初始包含使用者關係資料群組的 Shared Services 快取。您可能希望在外使用者目錄中建立新群組之後，初始快取重新整理，或在新增新使用者至現有群組之後加以規定。僅在 Shared Services 執行在快取中使用資料之後，才重新整理快取。
啟用 SSO 相容性	如果您的部署已與 Oracle Business Intelligence Enterprise Edition 11.1.1.5 版或以前版本整合，請選取此選項。

表格 4-6 (續) 使用者目錄的安全性選項

參數	說明
啟用委派使用者管理模式	此選項可啟用 EPM 系統產品的委派使用者管理，以支援提供活動的分散式管理。請參閱 <i>Oracle Enterprise Performance Management System User Security 管理手冊</i> 中的「委派使用者管理」。
啟用 SSO	此選項可支援安全性代理程式的 SSO，例如 Oracle Access Manager
SSO 提供者或代理程式	<p>從應接受 SSO 的 EPM 系統產品中選取來源 Web 識別管理解決方案。若沒列出您的 Web 識別管理解決方案 (例如 Kerberos)，請選取其他。</p> <p>選取 SSO 提供者時，會自動選取偏好的 SSO 機制和名稱。您可以視需要變更 SSO 機制名稱 (HTTP 標頭或自訂登入類別)。</p> <p>如果您選取 Other 作為 SSO 提供者或代理程式，則必須確定其支援 EPM System 支援的 SSO 機制。請參閱 <i>Oracle Enterprise Performance Management System Security Configuration Guide (僅英文版)</i> 中的「受支援的 SSO 方法」。</p>
SSO 機制	<p>選取的 Web 識別管理解決方案用於將使用者登入名稱提供給 EPM System 產品的方法。如需可接受之 SSO 方法的說明，請參閱 <i>Oracle Enterprise Performance Management System Security Configuration Guide (僅英文版)</i> 中的「受支援的 SSO 方法」。</p> <ul style="list-style-type: none"> · Custom HTTP Header：設定安全性代理程式傳遞至 EPM System 的標頭名稱。 · Custom Login Class：指定處理 HTTP 驗證要求的自訂 Java 類別。請參閱 <i>Oracle Enterprise Performance Management System Security Configuration Guide (僅英文版)</i> 中的「自訂登入類別」。

 **備註：**

自訂登入類別不同於自訂驗證

- HTTP Authorization Header：此為標準的 HTTP 機制。
- Get Remote User from HTTP Request：如果安全性代理程式植入 HTTP 要求的遠端使用者，請選取此選項。

表格 4-6 (續) 使用者目錄的安全性選項

參數	說明
自訂驗證模組	<p>完全符合條件的自訂驗證模組 Java 類別名稱 (例如 <code>com.mycompany.epm.CustomAuthenticationImpl</code>)，應用於驗證所有選取了自訂驗證模組之使用者目錄的使用者。只有在目錄組態已啟用驗證模組 (預設值) 時，才可將其用於使用者目錄。</p> <p>Oracle Hyperion Foundation Services 需要自訂的驗證 JAR 檔案名為 <code>CustomAuth.jar</code>。<code>CustomAuth.jar</code> 必須在 <code>MIDDLEWARE_HOME\user_projects\domains\WEBLOGIC_DOMAIN\lib</code> 中，通常是在 <code>C:\Oracle\Middleware\user_projects\domains\EPMSys\lib</code> 中。</p> <p>在所有用戶端安裝上，<code>CustomAuth.jar</code> 必須在 <code>EPM_ORACLE_HOME\common\jlib\11.1.2.0</code> 中，通常會是在 <code>C:\Oracle\Middleware\EPMSys11R1\common\jlib\11.1.2.0</code> 中。</p> <p>您可以在此 JAR 檔案中使用任何套件結構與類別名稱。如需詳細資訊，請參閱 <i>Oracle Enterprise Performance Management System Security Configuration Guide (僅英文版)</i> 中的「使用自訂驗證模組」。</p>

5. 按一下 **確定**。
6. 重新啟動 Foundation Services 和其他 EPM System 元件。

重新產生加密金鑰

Oracle Enterprise Performance Management System 使用下列金鑰來確保安全性：

- 「單一登入憑證」加密金鑰，用來加密和解密 EPM System SSO 憑證。此金鑰儲存於 Oracle Hyperion Shared Services Registry 中
- 「信任服務」金鑰，EPM System 元件用來驗證要求 SSO 憑證的服務授權
- 「提供者組態」加密金鑰，用於加密 EPM System 安全性用來繫結已設定外部使用者目錄的密碼 (啟用 LDAP 之使用者目錄的使用者 DN 密碼)。設定外部使用者目錄時，會設定此密碼。

定期變更這些金鑰以強化 EPM System 安全性。Oracle Hyperion Shared Services 和 EPM System 的子系統使用有 128 位元金鑰強度的 AES 加密方式。

▲ 注意：

當您重新產生「單一登入加密」金鑰時，Oracle Hyperion Financial Management 和 Oracle Hyperion Profitability and Cost Management 所用的工作流程就會失效。重新產生金鑰後，請開啟並儲存工作流程，使其重新生效。

若要重新產生「單一登入加密」金鑰、「提供者組態」金鑰或「信任服務」金鑰，請執行以下動作：

1. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。請參閱[啟動 Shared Services Console](#)。
2. 依序選取**管理及設定使用者目錄**。
3. 選取**加密選項**。
4. 在**加密選項**中，選取您要重新產生的金鑰。

表格 4-7 EPM System 加密選項

選項	說明
單一登入憑證	<p>選取此選項以重新產生要用來加密和解密 EPM System SSO 憑證的加密金鑰。</p> <p>如果安全性選項中的啟用 SSO 相容性已勾選，請選取以下其中一個按鈕：</p> <ul style="list-style-type: none"> • 產生新金鑰可建立新的 SSO 憑證加密金鑰 • 重設為預設可還原預設的 SSO 憑證加密金鑰
信任的服務金鑰	選取此選項以重新產生信任驗證金鑰，供 EPM System 元件用來驗證要求 SSO 憑證的服務授權。
提供者組態金鑰	選取此選項以重新產生用來加密密碼 (啟用 LDAP 之使用者目錄的使用者 DN 密碼) 的金鑰，該密碼是 EPM System 安全性用來繫結已設定外部使用者目錄的密碼。設定外部使用者目錄時，會設定此密碼。

 **備註：**

如果您回復至預設加密金鑰，則必須從所有 EPM System 主機中，刪除現有 keystore 檔案 (`EPM_ORACLE_HOME/common/CSS/ssHandlerTK`)。

5. 按一下**確定**。
6. 如果您選擇產生新的 SSO 加密金鑰，請完成此步驟。
 - a. 按一下**下載**。
 - b. 按一下**確定**，以便將 ssHandlerTK (支援新 SSO 加密金鑰的金鑰存放區檔案) 儲存至裝載 Oracle Hyperion Foundation Services 之伺服器中的資料夾。
 - c. 將 ssHandlerTK 複製至所有 EPM System 主機上的 `EPM_ORACLE_HOME/common/CSS`。
7. 重新啟動 Foundation Services 和其他 EPM System 元件。

使用特殊字元

Active Directory 與其他 LDAP 型使用者目錄允許在實體 (例如 DN、使用者名稱、角色及群組名稱) 中使用特殊字元。Oracle Hyperion Shared Services 則需經過特殊處理後，才可辨識這類字元。

一般而言，在使用者目錄設定中指定特殊字元時必須使用逸出字元；例如，基本 DN 與群組 URL。下表列出可用在使用者名稱、群組名稱、使用者 URL、群組 URL，以及使用者 DN 之 OU 值中的特殊字元。

表格 4-8 支援的特殊字元

字元	名稱或意義	字元	名稱或意義
(左括弧	\$	貨幣符號
)	右括弧	+	加號
"	雙引號	&	& 符號
'	單引號	\	反斜線
,	逗號	^	插入號
=	等於	;	分號
<	小於	#	井字號
>	大於	@	於

 **備註：**

請勿在基礎 DN 內的組織單位名稱中使用 / (正斜線)

- 「登入使用者」屬性值不可使用特殊字元。
- 使用者名稱、群組名稱、使用者與群組 URL，以及使用者 DN 的 OU 名稱不可使用星號 (*)。
- 不支援包含特殊字元組合的屬性值。
- & 符號可不搭配逸出字元使用。在 Active Directory 設定中，& 符號必須指定為 & 。
- 使用者與群組名稱不可包含反斜線 (\) 與正斜線 (/)。例如不可使用 test/\user 與 new\test/user 一類的名稱。

表格 4-9 無需逸出的字元

字元	名稱或意義	字元	名稱或意義
(左括弧	'	單引號
)	右括弧	^	插入號
\$	貨幣符號	@	於
&	& 符號		

 **備註：**

& 必須指定為 & 。

在使用者目錄設定 (使用者名稱、群組名稱、使用者 URL、群組 URL 及使用者 DN) 中使用這些字元時，必須加以逸出。

表格 4-10 在使用者目錄組態設定中逸出特殊字元

特殊字元	逸出	設定範例	逸出範例
逗號 (,)	反斜線 (\)	ou=test,ou	ou=test\,ou
加號 (+)	反斜線 (\)	ou=test+ou	ou=test\+ou
等於 (=)	反斜線 (\)	ou=test=ou	ou=test\=ou
井字號 (#)	反斜線 (\)	ou=test#ou	ou=test\#ou
分號 (;)	反斜線 (\)	ou=test;ou	ou=test\;ou
小於 (<)	反斜線 (\)	ou=test<ou	ou=test\<>ou
大於 (>)	反斜線 (\)	ou=test>ou	ou=test\>ou
引號 (")	兩個反斜線 (\)	ou=test"ou	ou=test\\"ou
反斜線 (\)	三個反斜線 (\)	ou=test\ou	ou=test\\\ou

 **備註：**

- 在「使用者 DN」中，雙引號 (") 必須使用一個反斜線加以逸出。例如 ou=test"ou 必須指定為 ou=test\"ou。
- 在「使用者 DN」中，反斜線 (\) 必須使用一道反斜線加以逸出。例如，ou=test\ou 必須指定為 ou=test\\ou。

 **注意：**

若未指定使用者 URL，則在 RDN 根目錄內所建立的使用者，即不可含有 / (正斜線) 或 \ (反斜線)。同理，若未指定群組 URL，則在 RDN 根目錄內所建立的群組，亦不可在名稱中使用這些字元。例如，像是 OU=child\ou,OU=parent/ou 或 OU=child/ou,OU=parent\ou 的群組名稱並不受支援。然而，您若是在使用者目錄組態中使用唯一的屬性作為 ID Attribute，即不在此限。

Native Directory 中的特殊字元

Native Directory 中的使用者與群組名稱支援特殊字元。

表格 4-11 支援的特殊字元：Native Directory

字元	名稱或意義	字元	名稱或意義
@	於	,	逗號
#	井字號	=	等於
\$	貨幣符號	+	加號
^	插入號	;	分號
(左括弧	!	驚嘆號

表格 4-11 (續) 支援的特殊字元：Native Directory

字元	名稱或意義	字元	名稱或意義
)	右括弧	%	百分比
'	單引號		

5

使用自訂驗證模組

另請參閱：

- [簡介](#)
- [使用案例的範例和限制](#)
- [先決條件](#)
- [設計和編碼的考量](#)
- [部署自訂驗證模組](#)

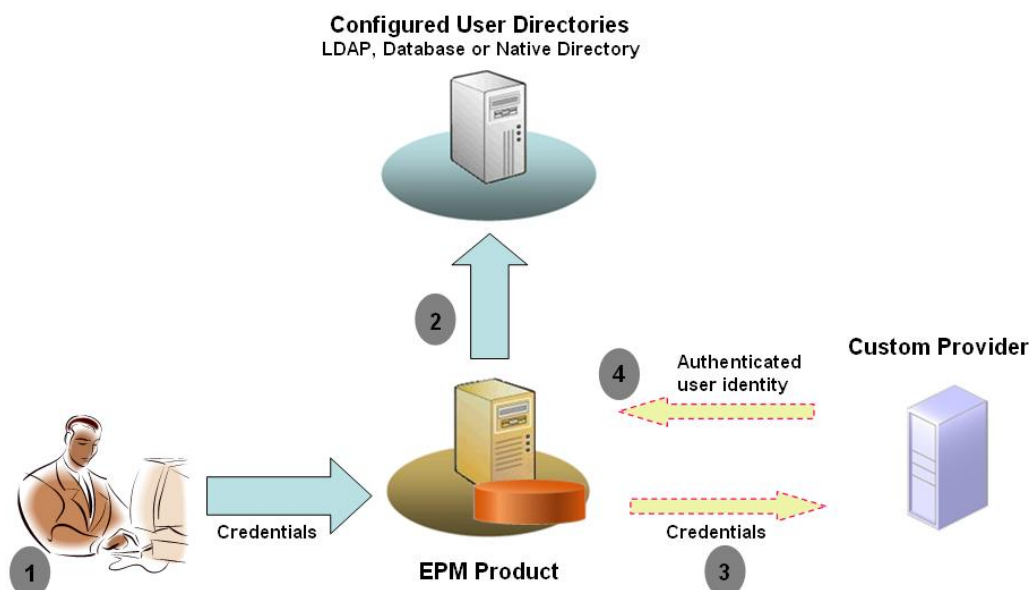
簡介

自訂驗證模組就是客戶開發並實作來驗證 Oracle Enterprise Performance Management System 使用者的 Java 模組。一般來說，EPM System 產品會使用登入畫面，擷取用來驗證使用者的使用者名稱和密碼。除了使用 EPM System 驗證機制之外，您也可以使用自訂的驗證模組來驗證使用者，並將已通過驗證的使用者認證傳遞到 EPM System 來進一步處理。實作自訂驗證模組並不會牽涉到修改 EPM System 產品。

且無論是豐富型用戶端 (例如 Oracle Smart View for Office 和 Oracle Essbase Studio) 還是精簡型用戶端 (例如 Oracle Hyperion Enterprise Performance Management Workspace)，都能使用自訂驗證模組。

自訂驗證模組會利用使用者在登入某個 EPM System 產品時所輸入的資訊。如果您已針對某個使用者目錄啟用自訂驗證模組，該模組會驗證使用者。自訂驗證模組在成功驗證使用者之後，會把使用者名稱傳回 EPM System。

下列圖例代表自訂驗證案例的範例：



例如，您可以把 RSA SecurID 基礎架構當作自訂提供者，以確保 EPM System 會使用透明的強式驗證機制。以下是這個程序的概觀：

1. 使用者輸入認證 (通常是使用者名稱和密碼) 以存取 EPM System 產品。這些認證應該會向自訂驗證模組使用的提供者，唯一識別該使用者。例如，如果您使用 RSA SecurID 基礎架構來驗證使用者，使用者會輸入 RSA 使用者 ID 和 PIN 碼 (而非 EPM System 使用者 ID 和密碼)。
2. EPM System 會使用搜尋順序 (請參閱 [搜尋順序](#))，依序在不同的已設定使用者目錄中尋找該使用者。
 - 如果您沒有針對自訂驗證來設定目前的使用者目錄，EPM System 會嘗試透過 EPM System 驗證機制來尋找並驗證該使用者。
 - 如果您已針對自訂驗證來設定該使用者目錄，EPM System 會把驗證程序委派給自訂模組。
3. 如果 EPM System 將驗證委派給自訂模組，自訂驗證模組會接受該認證，並使用自己的邏輯，根據某個自訂提供者 (例如 RSA SecurID 基礎架構) 來引導使用者驗證。
4. 如果自訂驗證模組成功地根據自己的提供者來驗證使用者，該模組會將使用者名稱傳回 EPM System，或是傳回 Java 例外。

自訂驗證模組傳回的使用者名稱，必須與已針對自訂驗證啟用的某個使用者目錄中的某個使用者名稱完全相符。

- 如果自訂驗證模組傳回使用者名稱，EPM System 會在已針對自訂驗證啟用的某個使用者目錄中尋找該使用者。在這個階段，EPM System 不會搜尋沒有針對自訂驗證設定的使用者目錄。
- 如果自訂驗證模組擲回例外，或是傳回 null 使用者，EPM System 會繼續依照搜尋順序，在剩下沒有針對自訂驗證啟用的使用者目錄中搜尋該使用者。如果 EPM System 沒有找到與認證相符的使用者，就會顯示錯誤。

使用案例的範例和限制

自訂驗證實作的案例包括下列項目：

- 新增一次性密碼支援
- 根據 [Resource Access Control Facility \(RACF\)](#) 來執行驗證
- 新增對於已啟用 LDAP 之使用者目錄的「簡單驗證及安全性階層」(SASL) 繫結，而非新增簡單 LDAP 繫結

如果您實作自訂驗證模組，擁有挑戰/回應機制的驗證方式可能無法正常運作。系統不會把自訂驗證模組擱回的自訂訊息傳播至用戶端。由於用戶端 (例如 **Oracle Hyperion Enterprise Performance Management Workspace**) 會覆寫該錯誤訊息以顯示一般訊息，因此下列案例是無效的：

- 兩個連續的 RSA SecurID PIN 碼
- 有挑戰機制的密碼變體，例如輸入密碼的第一個、最後一個和第三個字母

先決條件

- 已經過完全測試，且名為 CustomAuth.jar 的 Java 封存，其中包含自訂驗證模組程式庫。CustomAuth.jar 必須實作在 com.hyperion.css 套件中定義的公用介面 CSSCustomAuthenticationIF，作為標準 Oracle Hyperion Shared Services API 的一部分。請參閱 http://download.oracle.com/docs/cd/E12825_01/epm.111/epm_security_api_11111/client/com/hyperion/css/CSSCustomAuthenticationIF.html。
- 能夠以 Shared Services 管理員的身分存取 Shared Services

設計和編碼的考量

搜尋順序

您除了可以在 Oracle Hyperion Shared Services 中設定原生目錄之外，還可以設定多個使用者目錄。系統會把預設的搜尋順序位置指派給所有的已設定使用者目錄。您可以在 **Oracle Hyperion Shared Services Console** 修改搜尋順序。您可以移除搜尋順序中的已設定使用者目錄，但無法移除原生目錄。Oracle Enterprise Performance Management System 並不會使用搜尋順序中沒有的使用者目錄。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊*。

搜尋順序決定了 EPM System 在驗證使用者時，要用什麼樣的順序搜尋使用者目錄。如果使用者在某個使用者目錄中通過驗證，EPM System 就會停止搜尋，並傳回該使用者。如果 EPM System 無法根據搜尋順序中的使用者目錄成功驗證該使用者，就會拒絕驗證並傳回錯誤。

自訂驗證對於搜尋順序的影響

自訂驗證會影響 EPM System 安全性解譯搜尋順序的方式。

如果自訂驗證模組傳回使用者名稱，EPM System 只會在已針對自訂驗證啟用的某個使用者目錄中尋找該使用者。在這個階段，EPM System 會忽略沒有針對自訂驗證設定的使用者目錄。

瞭解自訂驗證的流程

我們將使用下列的使用案例來探討自訂驗證的流程：

- 使用案例 1
- 使用案例 2
- 使用案例 3

使用案例 1

下列表格詳細說明，在本案例中使用的 EPM System 使用者目錄組態及搜尋順序。本案例假設，自訂驗證模組使用 RSA 基礎架構來驗證使用者。

表格 5-1 案例 1 的設定

使用者目錄的類型和名稱	搜尋順序	自訂驗證	使用者名稱範例	密碼 ¹
原生目錄	1	已停用	test_user_1 test_user_2 test_user_3	password
已啟用 LDAP SunONE_West	2	已停用	test_ldap1 test_ldap_2 test_user_3 test_ldap_4	ldappassword
已啟用 LDAP SunONE_East	3	已啟用	test_ldap1 test_ldap_2 test_user_3	在 SunONE 用的為 ldappassword， 自訂模組中的為 RSA PIN。

¹ 為簡單起見，我們假設所有使用者都使用相同的使用者目錄密碼。

如要啟動驗證程序，使用者要在某個 EPM System 產品的登入畫面中輸入使用者名稱和密碼。在這個案例中，自訂驗證模組會執行下列動作：

- 接受使用者名稱和 RSA PIN，作為使用者認證
- 將採用 `username@providername` 格式的使用者名稱 (例如 `test_ldap_2@SunONE_East`) 傳回 EPM System 安全性

表格 5-2 使用者互動和結果

使用者名稱和密碼	驗證結果	登入使用者目錄
test_user_1/password	成功	原生目錄
test_user_3/password	成功	原生目錄
test_user_3/ ldappassword	成功	SunONE_West (搜尋順序 2) ¹
test_user_3/RSA PIN	成功	SunONE_East (搜尋順序 3) ²

表格 5-2 (續) 使用者互動和結果

使用者名稱和密碼	驗證結果	登入使用者目錄
test_ldap_2/ ldappassword	成功	SunONE_West (搜尋順序 2)
test_ldap_4/RSA PIN	失敗 EPM System 顯示驗證錯誤。 ³	

- 1 自訂驗證無法驗證該使用者，因為該使用者輸入了 EPM System 認證。EPM System 只能在一個沒有針對自訂驗證設定的使用者目錄中識別該使用者。該使用者不在原生目錄 (搜尋順序 1) 中，但在 SunONE West (搜尋順序 2) 中被識別出來。
- 2 EPM System 沒有在原生目錄 (搜尋順序 1) 或 SunONE West (搜尋順序 2) 中找到該使用者。自訂驗證模組會根據 RSA 伺服器來驗證使用者，並將 test_user_3@SunONE_EAST 傳回 EPM System。EPM System 在 SunONE East (搜尋順序 3) 中找到該使用者，而這個目錄是已啟用自訂驗證的使用者目錄。
- 3 Oracle 建議您，讓自訂模組驗證成功的所有使用者出現在搜尋順序內某個已啟用自訂驗證的使用者目錄中。如果自訂驗證模組傳回的使用者名稱沒有出現在搜尋順序內某個已啟用自訂驗證的使用者目錄中，登入就會失敗。

使用案例 2

下列表格詳細說明，在本案例中使用的 EPM System 使用者目錄組態及搜尋順序。本案例假設，自訂驗證模組使用 RSA 基礎架構來驗證使用者。

在這個案例中，自訂驗證模組會執行下列動作：

- 接受使用者名稱和 RSA PIN，作為使用者認證
- 將使用者名稱 (例如 test_ldap_2) 傳回 EPM System 安全性

表格 5-3 搜尋順序範例

使用者目錄	搜尋順序	自訂驗證	使用者名稱範例	密碼 ¹
原生目錄	1	已停用	test_user_1 test_user_2 test_user_3	password
已啟用 LDAP，例如 SunONE	2	已啟用	test_ldap1 test_ldap2 test_user_3	在 SunONE 用的為 ldappassword，自訂模組中的為 RSA PIN。

¹ 為簡單起見，我們假設所有使用者都使用相同的使用者目錄密碼。

如要啟動驗證程序，使用者要在某個 EPM System 產品的登入畫面中輸入使用者名稱和密碼。

表格 5-4 使用者互動和結果

使用者名稱和密碼	登入結果	登入使用者目錄
test_user_1/password	成功	原生目錄
test_user_3/password	成功	原生目錄
test_user_3/ldappassword	失敗	SunONE ¹

表格 5-4 (續) 使用者互動和結果

使用者名稱和密碼	登入結果	登入使用者目錄
test_user_3/RSA PIN	成功	SunONE ²

- 根據原生目錄進行的使用者驗證失敗，原因是密碼不相符。利用自訂驗證模組進行的使用者驗證失敗，原因是所用的密碼並不是有效的 RSA PIN 碼。EPM System 不會嘗試在 SunONE (搜尋順序 2) 中驗證該使用者，原因是自訂驗證設定會覆寫這個目錄中的 EPM System 驗證。
- 根據原生目錄進行的使用者驗證失敗，原因是密碼不相符。自訂驗證模組會驗證使用者，並將使用者名稱 test_user_3 傳回 EPM System。

使用案例 3

下列表格詳細說明，在本案例中使用的 EPM System 使用者目錄組態及搜尋順序。本案例假設，自訂驗證模組使用 RSA 基礎架構來驗證使用者。

為了清楚起見，Oracle 建議您在這些案例中，讓自訂驗證模組傳回採用 username@providername 格式的使用者名稱，例如 test_ldap_4@SunONE。

表格 5-5 搜尋順序範例

使用者目錄	搜尋順序	自訂驗證	使用者名稱範例	密碼 ¹
原生目錄	1	已啟用	test_user_1 test_user_2 test_user_3	RSA_PIN
已啟用 LDAP，例 如 MSAD	2	已停用	test_ldap1 test_ldap4 test_user_3	ldappassword
已啟用 LDAP，例 如 SunONE	3	已啟用	test_ldap1 test_ldap4 test_user_3	在 SunONE 用的為 ldappassword，自訂 模組中的為 RSA PIN。

- 為簡單起見，我們假設所有使用者都使用相同的使用者目錄密碼。

如要啟動驗證程序，使用者要在某個 EPM System 產品的登入畫面中輸入使用者名稱和密碼。

表格 5-6 使用者互動和結果

使用者名稱和密碼	驗證結果	登入使用者目錄
test_user_1/password	成功	原生目錄
test_user_3/RSA_PIN	成功	原生目錄
test_user_3/ldappassword	成功	MSAD (搜尋順序 2)
test_ldap_4/ldappassword	成功	MSAD (搜尋順序 2)
test_ldap_4/RSA PIN	成功	SunONE (搜尋順序 3)

使用者目錄及自訂驗證模組

如要使用自訂驗證模組，您可以分別把包含 EPM System 使用者和群組資訊的每個使用者目錄，設定成會將驗證作業委派給自訂模組。

使用自訂模組來驗證的 EPM System 使用者，都必須出現在搜尋順序內某個使用者目錄中 (請參閱[搜尋順序](#))。此外，您必須將該使用者目錄設定成會將驗證作業委派給自訂模組。

自訂提供者中的使用者識別 (例如 RSA SecurID 基礎架構中的 1357642)，可能會與在 Shared Services 設定之使用者目錄中的使用者名稱 (例如某個 Oracle Internet Directory 中的 jDoe) 不相同。自訂驗證模組在驗證使用者之後，必須將使用者名稱 jDoe 傳回 EPM System。

備註：

Oracle 建議您的最佳作法是，讓在 EPM System 設定之使用者目錄中的使用者名稱，與自訂驗證模組所用的使用者目錄中的使用者名稱完全相同。

CSSCustomAuthenticationIF Java 介面

自訂驗證模組必須使用 CSSCustomAuthenticationIF Java 介面，才能夠與 EPM System 安全性架構整合。該模組會在自訂驗證成功時傳回使用者名稱字串，並在驗證失敗時傳回錯誤訊息。若要完成驗證程序，自訂驗證模組傳回的使用者名稱必須出現在 Shared Services 搜尋順序內的某個使用者目錄中。EPM System 安全性架構支援 `username@providerName` 格式。

備註：

請確保自訂驗證模組傳回的使用者名稱中沒有 * (星號)，因為 EPM System 安全性在搜尋使用者時，會把星號解譯成萬用字元。

如需 CSSCustomAuthenticationIF 介面簽章，請參閱[程式碼範例 1](#)。

您的自訂驗證模組 (可以是類別檔案) 必須包含在 CustomAuth.jar 中，而該套件的結構並不重要。

如需 CSSCustomAuthenticationIF 介面的詳細資訊，請參閱[安全性 API 文件](#)。

CSSCustomAuthenticationIF 的 `authenticate` 方法支援自訂驗證。`authenticate` 方法接受使用者在嘗試存取 EPM System 時所輸入的認證 (使用者名稱和密碼) 以作為輸入參數。如果自訂驗證成功，該方法就會傳回字串 (使用者名稱)。如果驗證失敗，就會擲回 `java.lang.Exception`。該方法傳回的使用者名稱，必須唯一識別 Shared Services 搜尋順序內某個使用者目錄中的使用者。EPM System 安全性架構支援 `username@providerName` 格式。

 **備註：**

如要啟動資源 (例如 JDBC 連線集區)，請使用類別建構函式。這可讓系統不會在每次驗證時都載入資源，進而提高效能。

部署自訂驗證模組

單一 Oracle Enterprise Performance Management System 部署只能支援一個自訂模組。您可以啟用搜尋順序中一或多個使用者目錄的自訂驗證設定。

自訂驗證模組必須實作在 `com.hyperion.css` 套件中定義的公用介面 `CSSCustomAuthenticationIF`。本文件假設，您擁有功能完整的自訂模組，且該模組會定義系統用於根據您所選使用者提供者來驗證使用者的邏輯。您在開發並測試自訂驗證模組之後，必須在 **EPM System** 環境中實作該模組。

步驟簡介

請勿在自訂驗證程式碼中使用 `log4j` 來記錄錯誤。如果您在上一個版本所用的程式碼中使用 `log4j`，您必須先移除該 `log4j`，才能夠在這個版本中使用該程式碼。

如要實作自訂驗證模組，請完成以下步驟：

- 停止執行 **EPM System** 產品，包括 **Oracle Hyperion Shared Services**，以及所有使用 **Shared Services API** 的系統。
- 將自訂驗證模組的 **Java** 封存 `CustomAuth.jar` 複製到部署中：

- **WebLogic**：將 `CustomAuth.jar` 複製到 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` 中，這通常會是 `C:/Oracle/Middleware/user_projects/domains/EPMSysstem/lib`。

如果您是升級自擁有自訂驗證模組實作的版本 **11.1.2.0** 或 **11.1.2.1**，請將 `CustomAuth.jar` 從 `EPM_ORACLE_HOME/common/jlib/11.1.2.0` 移動到 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` 中。

- **所有用戶端部署**：將 `CustomAuth.jar` 複製到所有 **EPM System** 用戶端部署中的下列位置：

`EPM_ORACLE_HOME/common/jlib/11.1.2.0`，這通常會是 `Oracle/Middleware/common/jlib/11.1.2.0`。確定 `CustomAuth.jar` 檔案一律放在 `EPM_ORACLE_HOME/common/jlib/11.1.2.0` 目錄中。

針對與自訂驗證搭配使用的所有伺服器 and 用戶端，`CustomAuth.jar` 檔案必須存在於以下兩個位置：

- * `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib`
- * `EPM_ORACLE_HOME/common/jlib/11.1.2.0`

- 更新 **Shared Services** 中的使用者目錄設定。請參閱[更新 Shared Services 中的設定](#)。
- 啟動 **Shared Services**，然後啟動其他的 **EPM System** 產品。
- 測試您的部署。請參閱[測試您的部署](#)。

更新 Shared Services 中的設定

根據預設，所有使用者目錄的自訂驗證設定是停用的。您可以覆寫該預設行為，啟用特定外部使用者目錄或原生目錄的自訂驗證設定。

更新使用者目錄組態

對於必須啟用自訂驗證設定的使用者目錄，您必須更新這些目錄的組態。

若要更新使用者目錄組態，請：

1. 啟動 Oracle Hyperion Foundation Services。
2. 以系統管理員的身分存取 Oracle Hyperion Shared Services Console。
3. 依序選取**管理及設定使用者目錄**。
4. 在「已定義的使用者目錄」畫面上，選取您要變更自訂驗證設定的使用者目錄。

備註：

EPM System 僅使用包含在搜尋順序中的使用者目錄。

5. 按一下**編輯**。
6. 選取**顯示進階選項**。
7. 在**自訂模組**中，選取**驗證模組**以啟用目前使用者目錄的自訂模組。
8. 按一下**完成**。
9. 重複此程序，以更新搜尋順序中其他使用者目錄的組態。

更新安全性選項

請在開始執行下列程序之前，確保 CustomAuth.jar 在 `EPM_ORACLE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` 中。

若要更新安全性選項，請：

1. 以系統管理員的身分存取 Shared Services Console。
2. 依序選取**管理及設定使用者目錄**。
3. 選取**安全性選項**。
4. 選取**顯示進階選項**。
5. 在**驗證模組**中，輸入自訂驗證模組的完整類別名稱；而這個自訂驗證模組將用來驗證已選取該自訂驗證模組之所有使用者目錄中的使用者。例如 `com.mycompany.epm.CustomAuthenticationImpl`。
6. 按一下**確定**。

測試您的部署

如果您沒有針對自訂驗證來設定原生目錄，請勿使用原生目錄使用者來測試自訂驗證。

 **備註：**

您必須負起責任，找出並修正與自訂驗證模組有關的任何問題。Oracle 假設，您的自訂模組可以完美地將自訂模組所用之使用者目錄中的某個使用者，對映到 EPM System 搜尋順序內已啟用自訂驗證之使用者目錄中的某位使用者。

如要測試部署，請利用自訂模組所用之使用者目錄 (例如 RSA SecurID 基礎架構) 中的使用者認證登入 EPM System。這些認證與 EPM System 認證可能會不一樣。

如果 EPM System 產品讓您存取其資源，您的實作就算是成功。而表示系統沒有找到使用者的錯誤，並不一定代表實作失敗。如果您碰到這種情況，請確保您輸入的認證確實出現在自訂使用者儲存空間中，且在 EPM System 搜尋順序內已啟用自訂驗證的某個使用者目錄中，有個完全相符的使用者。

如何測試自訂驗證：

1. 確保 EPM System 產品正在執行中。
2. 存取某個 EPM System 元件，例如 Oracle Hyperion Enterprise Performance Management Workspace。
3. 以在已啟用自訂驗證的某個使用者目錄中定義之使用者的身分登入。
 - a. 在**使用者名稱**中，輸入您的使用者識別碼，例如某個 RSA User ID。
 - b. 在**密碼**中輸入密碼，例如某個 RSA PIN 碼。
 - c. 按一下**登入**。
4. 確認您可以存取 EPM System 產品的資源。

6

EPM System 的安全準則

另請參閱：

- [實作 SSL](#)
- [變更管理密碼](#)
- [重新產生加密金鑰](#)
- [變更資料庫密碼](#)
- [保護 Cookie 的安全](#)
- [縮短 SSO 憑證逾時時間](#)
- [複查安全性報表](#)
- [自訂嚴密驗證的驗證系統](#)
- [停用 EPM Workspace 除錯公用程式](#)
- [變更預設的 Web 伺服器錯誤頁面](#)
- [對於第三方軟體的支援](#)

實作 SSL

SSL 會使用加密系統加密資料。SSL 會在用戶端與伺服器之間建立安全連線，而資料在此連線上均可透過 SSL 安全地進行傳送。

如要保護您的 Oracle Enterprise Performance Management System 環境，請保護您的 Web 應用程式所用的所有通訊管道，以及使用 SSL 的使用者目錄連線。請參閱[已啟用 SSL 的 EPM System 元件](#)。

此外，請使用防火牆來保護所有代理程式通訊埠，例如連接埠 6861，也就是 Oracle Hyperion Reporting and Analysis 代理程式的連接埠。最終使用者並不需要存取 EPM System 代理程式連接埠。

變更管理密碼

預設的原生目錄管理使用者帳戶提供所有 Oracle Hyperion Shared Services 功能的存取權。這個密碼是您在部署 Oracle Hyperion Foundation Services 時所設定的。您必須定期變更這個帳戶的密碼。

編輯 *admin* 使用者帳戶，以變更其密碼。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「修改使用者帳戶」。

重新產生加密金鑰

請定期使用 Oracle Hyperion Shared Services Console 來重新產生下列項目：

- 單一登入憑證

 **注意：**

當您產生新的金鑰存放區時，Oracle Hyperion Financial Management 和 Oracle Hyperion Profitability and Cost Management 所用的工作流程就會失效。重新產生金鑰存放區後，請開啟並儲存工作流程，使其重新生效。

- 信任的服務金鑰
- 提供者組態金鑰

請參閱[重新產生加密金鑰](#)。

 **備註：**

Oracle Hyperion Shared Services 和 Oracle Enterprise Performance Management System 的子系統使用有 128 位元金鑰強度的 AES 加密方式。

變更資料庫密碼

請定期變更所有 Oracle Enterprise Performance Management System 產品資料庫的密碼。本節詳細說明變更 Oracle Hyperion Shared Services Registry 中資料庫密碼的程序。

如需變更 EPM System 產品資料庫密碼的詳細程序，請參閱 *Oracle Enterprise Performance Management System 安裝與組態手冊*。

若要變更 Shared Services 登錄中 EPM System 產品資料庫的密碼：

1. 使用資料庫管理主控台，變更為設定 EPM System 產品資料庫的使用者帳戶的密碼。
2. 停止執行 EPM System 產品 (Web 應用程式、服務及程序)。
3. 使用 EPM 系統組態程式，透過下列程序之一重新設定資料庫。

僅限 Oracle Hyperion Shared Services：

 **備註：**

在分散式環境中 (EPM System 產品與 Shared Services 位在不同的機器上)，您必須在所有伺服器上執行此程序。

- a. 在 EPM 系統組態程式的 Foundation 工作中，選取**設定資料庫**。
- b. 在「Shared Services 和登錄資料庫組態」頁面上，選取**連線到之前設定的 Shared Services 資料庫**。

- c. 為用於設定 Shared Services 資料庫的使用者帳戶指定新密碼。請勿變更任何其他的設定。
- d. 繼續設定，並在完成時按一下**完成**。

Shared Services 之外的其他 EPM System 產品：

備註：

請只針對部署在目前伺服器上的 EPM System 產品執行這些步驟。

如需詳細指示，請參閱 *Oracle Enterprise Performance Management System 安裝與組態手冊*。

4. 啟動 EPM System 產品和服務。

保護 Cookie 的安全

Oracle Enterprise Performance Management System Web 應用程式會設定 Cookie 來追蹤階段作業。設定 Cookie 時 (特別是階段作業 Cookie)，伺服器可以設定安全旗標，強制瀏覽器透過安全頻道傳送 Cookie。這種行為可降低階段作業遭到劫持的風險。

備註：

只有在 EPM System 產品是部署在已啟用 SSL 的環境中時，您才需要保護 Cookie 的安全。

請修改 Oracle WebLogic Server 階段作業描述元，以保護 WebLogic Server Cookie 的安全。將 session-param 元素中的 cookieSecure 屬性值設為 true。請參閱 [Oracle Fusion Middleware Programming Security for Oracle WebLogic Server 11g](#) 中的「Securing Web Applications」。

縮短 SSO 憑證逾時時間

預設的 SSO 憑證逾時時間為 480 分鐘。Oracle 建議您，縮短 SSO 憑證逾時時間 (例如縮短到 60 分鐘)，以便將該憑證在曝光後遭到重新使用的風險降到最低。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「設定安全性選項」。

複查安全性報表

「安全性報表」包含設定了稽核的安全性工作相關的稽核資訊。請定期在 Oracle Hyperion Shared Services Console 產生並複查該報表，尤其是要找出嘗試登入各個 Oracle Enterprise Performance Management System 產品失敗的情況，以及提供的變更。選取 [詳細資訊檢視](#) 做為報表產生選項，以根據修改過的屬性和新屬性值分組報表資料。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「產生報表」。

自訂嚴密驗證的驗證系統

您可以使用自訂驗證模組，將嚴密驗證加入 EPM System 中。例如，您可以在非挑戰回應模式中使用 RSA SecurID 雙重要素驗證。對於精簡型用戶端與豐富型用戶端來說，自訂驗證模組驗證是透明的，而且不需變更用戶端部署。請參閱[使用自訂驗證模組](#)。

停用 EPM Workspace 除錯公用程式

- 為協助您排解疑難，Oracle Hyperion Enterprise Performance Management Workspace 隨附未處理過的 JavaScript 檔案。為確保安全無虞，請移除生產環境中未處理過的 JavaScript 檔案：
 - 請建立 `EPM_ORACLE_HOME/common/epmstatic/wspace/js/` 目錄的備份。
 - 請移除 `EPM_ORACLE_HOME/common/epmstatic/wspace/js` 的每個子目錄中的 `.js` 檔案，但要保留 `DIRECTORY_NAME.js` 檔案。每個子目錄皆包含一個與目錄名稱相同的 `.js` 檔案。例如，`EPM_ORACLE_HOME/common/epmstatic/wspace/js/com/hyperion/bpm/web/common` 就包含 `Common.js`。請移除所有 `.js` 檔案，但要保留名稱與目錄名稱相同的檔案，在本案例中為 `Common.js`。
- EPM Workspace 提供了幾個除錯公用程式和測試應用程式；如果您是以除錯模式部署 EPM Workspace，就能夠使用這些公用程式和應用程式。為確保安全無虞，管理員務必要關閉 EPM Workspace 中的用戶端除錯功能。

如何關閉除錯模式：

- 以管理員身分登入 EPM Workspace。
- 依序選取**導覽**、**管理**，然後選取 **Workspace 伺服器設定**。
- 在「Workspace 伺服器設定」的 **ClientDebugEnabled** 中，選取**否**。
- 按一下**確定**。

變更預設的 Web 伺服器錯誤頁面

當應用程式伺服器無法接受要求時，後端應用程式伺服器的 Web 伺服器外掛程式 (例如 Oracle WebLogic Server 的 Oracle HTTP Server 外掛程式) 會傳回預設的錯誤頁面，其中包含外掛程式的建置資訊。在其他情況下 Web 伺服器也會顯示其預設錯誤頁面。攻擊者可透過此資訊找出公開網站的已知弱點。

請自訂 (Web 應用程式伺服器外掛程式和 Web 伺服器的) 該錯誤頁面，以便讓該頁面不會包含生產系統元件的資訊，例如同伺服器版本、伺服器類型、外掛程式建置日期，以及外掛程式類型。如需詳細資訊，請參閱應用程式伺服器及 Web 伺服器的廠商說明文件。

對於第三方軟體的支援

Oracle 感謝並支援第三方廠商提供的回溯相容性。因此，只要廠商表示提供回溯相容性，我們就會使用廠商提供的後續維護版本和 Service Pack。若發現任何不相容的情

況，Oracle 將會指定產品所應部署的修補程式版本 (並從支援對照表中移除不相容的版本)，或提供 Oracle 產品的維護版本或服務修正程式。

伺服器端更新：是否支援升級至第三方伺服器端元件受後續維護版本政策管理。一般而言，Oracle 支援將第三方伺服器端元件升級至目前受支援版本之 **Service Pack** 的下一個維護版本，但不支援升級至下一個主要版本。

用戶端更新：Oracle 支援用戶端元件的自動更新，包括更新至第三方用戶端元件的下一個主要版本。例如，您可以將瀏覽器 **JRE** 版本升級至目前受支援的 **JRE** 版本。

A

自訂驗證的程式碼範例

程式碼範例 1

備註：

請勿在自訂驗證程式碼中使用 `log4j` 來記錄錯誤。如果您在上一個版本所用的自訂驗證程式碼中使用 `log4j`，您必須先移該 `log4j`，才能夠在這個版本中使用該程式碼。

以下程式碼片段是空白的自訂模組實作：

```
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;
    }
}
```

輸入參數：

- 環境定義：包含地區設定資訊鍵值對的對映

- 使用者名稱：能唯一識別自訂模組用來驗證使用者之使用者目錄中使用者的識別碼。使用者會在登入某個 **Oracle Enterprise Performance Management System** 元件時，輸入此參數的值。
- 密碼：針對自訂模組用來驗證使用者之使用者目錄中使用者所設定的密碼。使用者會在登入某個 **EPM System** 元件時，輸入此參數的值。

程式碼範例 2

以下程式碼範例將示範，如何利用文字檔中的使用者名稱和密碼，使用自訂驗證功能來驗證使用者。您必須在類別建構函式中將使用者和密碼清單初始化，才能讓自訂驗證功能運作正常。

```
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");
    }
}

```

程式碼範例 2 的資料檔案

請確保資料檔案的名稱為 `datafile.txt`；這就是我們在範例程式碼中使用的名稱，而該檔案已包含在您建立的 **Java** 封存中。

請在作為自訂使用者目錄的文字檔中使用以下內容，以便支援程式碼範例 2 所實作的自訂驗證模組 (請參閱[程式碼範例 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

```

如果您計畫傳回採用 `username@providername` 格式的使用者名稱，請在作為自訂使用者目錄的文字檔中使用以下內容：

```

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

實作自訂登入類別

Oracle Enterprise Performance Management System 提供

`com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl`，以擷取 x509 憑證中的使用者識別 (DN)。

如果您必須從非 DN 憑證中的屬性擷取使用者識別，您必須開發並實作類似

`com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl` 的自訂登入類別，如本附錄中所示。

自訂登入類別的程式碼範例

本程式碼範例說明預設 `com.hyperion.css.sso.agent.X509CertificateSecurityAgentImpl` 的實作。一般來說，您必須自訂這個實作的 `parseCertificate(String sCertificate)` 方法，才能從非 DN 的憑證屬性取得使用者名稱。

```
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

```

部署自訂登入類別

如要實作自訂登入類別，請完成以下步驟：

1. 建立並測試自訂登入類別。確保程式碼中沒有任何對 log4j 的參照。請參閱[自訂登入類別的程式碼範例](#)。

您可以為自訂類別設定任何名稱。

2. 將自訂登入類別封裝到 CustomAuth.jar 中
3. 將 CustomAuth.jar 複製到部署中：

- **WebLogic**：將 CustomAuth.jar 複製到 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` 中，這通常會是 `Oracle/Middleware/user_projects/domains/EPMSysstem/lib`。

備註：

如果您是升級自擁有自訂登入類別實作的版本 **11.1.2.0** 或 **11.1.2.1**，請將 CustomAuth.jar 從 `EPM_ORACLE_HOME/common/jlib/11.1.2.0` 移動到 `MIDDLEWARE_HOME/user_projects/domains/WEBLOGIC_DOMAIN/lib` 中。

- **用戶端部署**：將 CustomAuth.jar 複製到所有 Oracle Enterprise Performance Management System 用戶端部署中的下列位置：

`EPM_ORACLE_HOME/common/jlib/11.1.2.0`，這通常會是 `Oracle/Middleware/common/jlib/11.1.2.0`

Oracle 建議您，如果您要使用自訂登入類別，請啟用「用戶端憑證驗證」

C

在不同的使用者目錄之間移轉使用者和群組

簡介

有許多情況可能會造成已提供的 Oracle Enterprise Performance Management System 使用者的使用者和群組識別過時。當可用的提供資訊過時的時候，EPM System 元件就會變得無法使用。以下是可能會建立過時之提供資料的情況：

- 淘汰使用者目錄：組織可能會在將使用者移動到另一個使用者目錄之後，淘汰先前的使用者目錄。
- 版本升級：使用者目錄的版本升級可能會牽涉到主機名稱或作業系統環境要求的變更。
- 廠商變更：組織可能會停止使用某個使用者目錄，改用另一個廠商的使用者目錄。例如，組織可能會將自己的 Oracle Internet Directory 替換成 SunONE Directory Server。

備註：

- 在本附錄中，我們會把您要逐步淘汰的使用者目錄稱為來源使用者目錄，並把您要移入使用者帳戶的使用者目錄稱為目標使用者目錄。
- 此移轉程序不支援將使用者帳戶從來源使用者目錄移轉至目標使用者目錄，而僅支援它們在 EPM 應用程式中的關聯。必須在目標使用者目錄中手動建立使用者。此程序適用於任何來源使用者目錄 (包含原生目錄) 的使用者。

如果使用 Hyperion Shared Services 設定的來源使用者目錄具有原生目錄群組除外的群組，則也應在目標使用者目錄中建立這些群組。

先決條件

- 其提供資料要在不同使用者目錄之間移轉的 Oracle Enterprise Performance Management System 使用者和群組，必須在目標使用者目錄中。
存在於來源使用者目錄中的群組關係，在目標使用者目錄中也必須存在。
- EPM System 使用者的使用者名稱在來源和目標使用者目錄中都必須是相同的。

移轉程序

匯出原生目錄的資料

在來源環境中依照下列步驟：

請使用 Oracle Hyperion Enterprise Performance Management System Lifecycle Management 來僅從原生目錄匯出下列共用服務物件：

- 原生目錄群組
- 指派的角色
- 委派清單

生命週期管理會建立多個匯出檔案，通常會建立在 `EPM_ORACLE_INSTANCE/import_export/USER_NAME/EXPORT_DIR/resource/Native Directory`，其中 `USER_NAME` 是使用者的識別 (例如執行匯出作業的 `admin`)，而 `EXPORT_DIR` 是匯出目錄的名稱。一般來說，系統會建立下列檔案：

- `Groups.csv`
- `Assigned Roles.csv`
- `Delegated Lists.csv`
- 每個已部署應用程式的 `Assigned Roles/PROD_NAME.csv`，其中 `PROD_NAME` 是某個 Oracle Enterprise Performance Management System 元件的名稱，例如 `Shared Services`。

備註：

- 如需如何使用生命週期管理 匯出資料的詳細指示，請參閱 *Oracle Enterprise Performance Management System Lifecycle Management Guide (僅英文版)*。
- 請確保未匯出 `Users.csv` 檔案。

當您匯出人工因素之後，請確認「移轉狀態報表」上最後一個匯出作業的狀態為 `Completed`。

若要匯出原生目錄資料，請執行下列動作：

1. 在 Oracle Hyperion Shared Services Console 的「檢視」窗格中，選取 **Foundation** 應用程式群組中的 **Shared Services** 應用程式。
2. 若要移轉，請僅從以下清單中選取所需的物件：
 - 原生目錄群組
 - 指派的角色
 - 委派清單
3. 按一下 **匯出**。
4. 輸入匯出封存的名稱，預設名稱是 `admin DATE`，例如 `admin 13-03-18`。
5. 按一下 **匯出**。

匯入原生目錄資料

在目標環境中依照下列步驟：

1. 手動建立：
 - a. 目標外部使用者目錄中的使用者，類似於來源使用者目錄。

- b. 目標外部使用者目錄中的群組，類似於來源使用者目錄，但原生目錄群組除外。
2. 設定目標使用者目錄。
如果您已經把使用者帳戶從來源使用者目錄移動到另一個使用者目錄，請在 EPM System 中將目標使用者目錄新增為外部使用者目錄。例如，如果您已經將使用者帳戶從 Oracle Internet Directory 移動到 SunONE Directory Server，請將 SunONE Directory Server 新增為外部使用者目錄。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「第三章：設定使用者目錄」。

 **備註：**

請確認目標使用者目錄包含其資料已移轉出來源使用者目錄之所有 EPM System 使用者的使用者帳戶和群組。

如果您已經將使用者移動到某個已經定義為外部使用者目錄的使用者目錄，請確認您可以在 Oracle Hyperion Shared Services 中看到這些使用者帳戶，方法是在 Shared Services Console 中搜尋使用者。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「搜尋使用者、群組、角色與委派清單」。

當您將目標使用者目錄設定成外部使用者目錄時，請確認「Login Attribute」特性所指向之屬性的值原先是作為來源使用者目錄中的使用者名稱。請參閱 [先決條件](#)。

3. 將目標使用者目錄移至搜尋順序頂端。

 **備註：**

如果目標使用者目錄的名稱與來源使用者目錄的名稱完全相同，您必須刪除 EPM System 組態中的來源使用者目錄。

Shared Services 指派給剛新增之使用者目錄的搜尋優先順序，會低於指派給現有目錄的搜尋順序。請變更搜尋順序，以便讓目標使用者目錄的搜尋優先順序高於來源使用者目錄。這個順序能讓 Shared Services 在搜尋來源使用者目錄之前，就先在目標使用者目錄中找到使用者。請參閱 *Oracle Enterprise Performance Management System User Security 管理手冊* 中的「管理使用者目錄搜尋順序」。

4. 請重新啟動 Oracle Hyperion Foundation Services 及其他 EPM System 元件，以便強制執行您所做的變更。
5. 匯入原生目錄資料 (從來源環境匯出)：
請搭配建立/更新選項來執行生命週期管理，以便匯入您先前從原生目錄匯出的資料 (如下所列)。
 - Groups.csv
 - Assigned Roles.csv
 - Delegated Lists.csv

 **備註：**

- 如需如何使用生命週期管理匯入資料的詳細指示，請參閱 *Oracle Enterprise Performance Management System Lifecycle Management Guide (僅英文版)*。
- 請確保未匯入 Users.csv 檔案。

當您匯入資料之後，請確認「移轉狀態報表」上最後一個匯入作業的狀態為 Completed。

若要匯入原生目錄資料，請執行下列動作：

- a. 在 Shared Services Console 的「檢視」窗格中，展開**檔案系統**。
- b. 選取匯入檔案的檔案系統位置。
- c. 選取您要匯入提供資訊的人工因素類型。
- d. 按一下**匯入**。
- e. 按一下**確定**。

產品專有的更新

 **注意：**

Oracle 建議您，先將使用者和群組資料備份在 Oracle Enterprise Performance Management System 元件所用的儲存庫中，再開始進行產品專有的更新。當您更新本機產品儲存庫中的資訊之後，您就只能利用備份來回復至本機產品儲存庫中的舊使用者和群組資料。

Planning

Oracle Hyperion Planning 會將已提供的使用者和群組資訊存放在 Planning 儲存庫中。如果您在不同使用者目錄之間移轉使用者和群組的動作，導致原生目錄中的使用者識別遭到變更，您就必須選取「移轉使用者/群組」，讓 Planning 儲存庫中的資訊能夠與原生目錄中的資訊同步。當您在 Planning 中指派資料表單、成員和工作清單的存取權時，就會看到這個按鈕。

Financial Management

Oracle Hyperion Financial Management 會將已提供來存取物件的使用者和群組資訊儲存在本機 Financial Management 儲存庫中。如果您在不同使用者目錄之間移轉使用者和群組的動作，導致原生目錄中的使用者和群組資訊遭到變更，您就必須讓 Financial Management 儲存庫中的資訊與原生目錄中的資訊同步。