Metadata Management Tutorial

User External Authentication Using erwin Metadata Management (EMM)

TABLE OF CONTENTS

TABLE OF FIGURES

1 Overview

Disclaimer

Some of the features detailed in this document may not apply and/or be available for the particular erwin Metadata Management (EMM) edition you may have.

erwin Metadata Management (EMM) use a role-based authorization (permissions) system. The rolename associated with a session determines what permissions are available (read, update, administrate, etc.), and to some extent what user interface will be presented (business UI, e.g.). In order to determine what rolename to use for a given session, erwin Metadata Management (EMM) allows for two modes of authentication:

- **Native Authentication** Mode where one uses the provided rolename/password based authentication to assign a rolename to a session.

- **3rd-Party Authentication Mode** where another single sign-on (SSO) application provides a rolename and some identifying information about the user for a session.

The Native Authentication Mode may be use for demonstrations and proof-of-concept type exercises (evaluations, etc.). However, for a production environment, most customers will wish to use their own SSO type technology, and will use the 3rd-Party Authentication Mode.

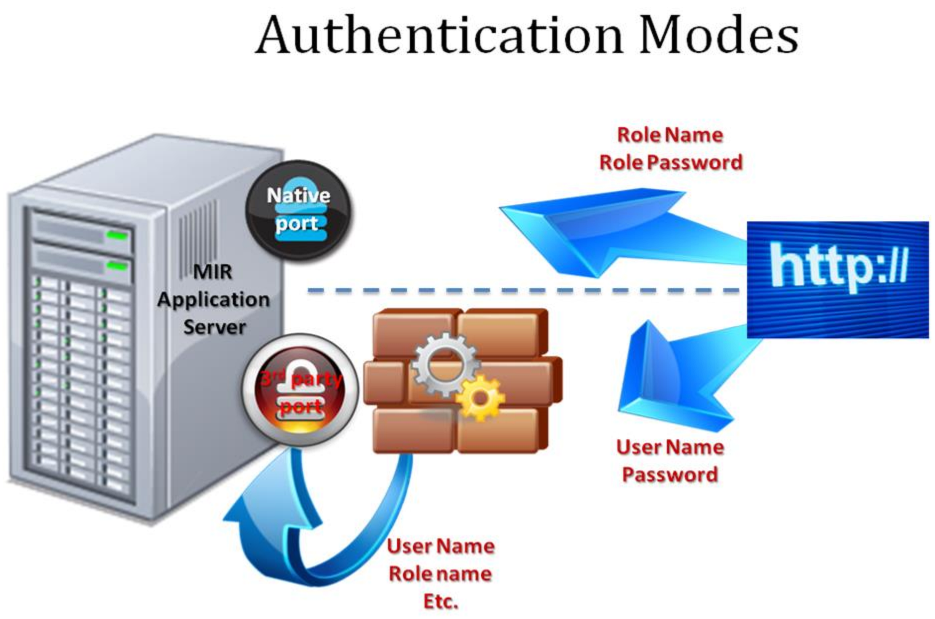


Figure 1 - erwin Metadata Management (EMM) Authentication Modes

This document describes the authentication part of the security and we present different modes of authentication that erwin Metadata Management (EMM) would support.

1.1 Role Based Authorization

For both Native and 3rd-Party Authentication, rolenames are created in erwin Metadata Management (EMM) using the Role Administration feature in the web UI. This document is not intended to describe role-based authorization in detail, but provides this section for context and illustration.

1.2 Native/LDAP Authentication Mode

When first installing and configuration erwin Metadata Management (EMM), or for proof-of-concept and evaluation scenarios, it is likely unnecessary to integrate the metadata management environment with the organization's SSO system. In these cases, one may wish to use the native mode of authentication.

For native mode a password must be associated with each rolename. Once a rolename is defined, a user may log in to the repository using that rolename and the associated password. This mode is accessible through the standard URL configured during the setup process. This is the default behavior after installation of erwin Metadata Management (EMM).

In addition, one may connect to an LDAP service and accept authentication from that to allow users to sign in. Please see the on-line help documentation in erwin Metadata Management (EMM) for details on how to set up LDAP authentication and role assignment.

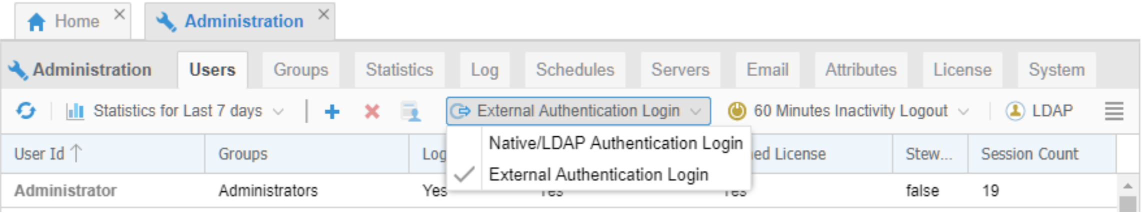
1.3 External Authentication Mode

Some organizations will setup an enterprise wide authentication and SSO based environment. These systems are designed to "screen" all web applications running in the enterprise, requiring a user to be logged in (authenticated) before accessing the web application. The login typically persists across sessions and different web applications, until the user logs out. Also, these authentication systems generally have a way to provide information about the user's identity to the downstream applications so that further security policies (like authorization, etc.) may be applied.

2 Configurating External Authentication

2.1 Enabling External Authentication in the Metadata Manager UI

By default, erwin Metadata Management (EMM) only supports the native/LDAP authentication mode. In-order to enable the external authentication mode, one should go to Tools > Administration > Users and select External Authentication Login:


Home
Administration
Administration
Users
Groups
Statistics
Log
Schedules
Servers
Email
Attributes
License
System
Statistics for Last 7 days
External Authentication Login
60 Minutes Inactivity Logout
LDAP
Native/LDAP Authentication Login
External Authentication Login

User Id ↑	Groups	Log	ed License	Stew...	Session Count
Administrator	Administrators	Yes	Yes	false	19

Figure 2 - Select External Authentication Login model

Once this is accomplished, erwin Metadata Management (EMM) is open to access without authentication. It is up to your organization to restrict access (e.g., via a firewall) to erwin Metadata Management (EMM) directly, only allowing the External Authentication system to redirect authenticated connection requests. This HTTP request must contain the information required to identify the authenticated user (username) along with any other identifying information or group assignments to be made. It is assumed by erwin Metadata Management (EMM) that this has been authenticated and thus will be acted upon without complaint.

2.2 Identification Parameters

In external authentication authentication mode, erwin Metadata Management (EMM) does not provide a user interface for entering a rolename (nor password). Instead, when accessing erwin Metadata Management (EMM) using the external authentication mode URL, this information must be “passed to” the erwin Metadata Management (EMM) web application by way of HTTP-based header variables.

The **HTTP-based header variables** are as follows:

- MIRRoleName – Role Name mapped to the MIR configured role names.
- MIRUserId – User unique identifier (likely to be the same as the username or userid used for login to the SSO environment)
- MIRUserName – User DisplayName (e.g., full name including first and last name). If this header variable is not provided, the UserID will be presented for all user name displays (e.g., “Hello <name>”).
- MIRUserEmail – User E-mail address used for notification purposes.
- MIRUserDescription – Additional unformatted descriptive information
- <Other properties as defined> – Additional identifying and descriptive information (e.g., phone, department, etc.) as defined in erwin Metadata Management (EMM) using the Custom Attributes tab in the Administration pane.

2.3 Authentication Javascript

When a request is handled by erwin Metadata Management (EMM), a specific Javascript file is used to handle any special considerations, such as:

- Translation of parameters
- External Authentication lookup for group assignments, identifying information, etc.
- Rule-based group assignment

To ensure that a javascript file is being used, it must be placed in \conf\MIRScripts\MetaIntegration\Authentication\.

An example of this file may be found in the installation at \conf\template\authentication.

The name is not fixed, but it has to have a section as follows:

```
function defineScriptMetadata(scriptDefinition) {
 scriptDefinition.setName("Multi-user testing authentication");
 scriptDefinition.setDescription("Designed to skip the authentication dialog for all new
sessions.");

 scriptDefinition.addObjectType(MIRElementType.REPOSITORY);

 scriptDefinition.addEventType(ScriptDefinition.EVENT_LOGIN);
}
```

3 Examples

3.1 Oracle Access Manager (OAM) Formerly Oblix

As a means of illustrating how to configure an SSO environment to integrate with erwin Metadata Management (EMM), this section describes how to setup Oracle Access Manager to provide authentication for erwin Metadata Management (EMM). It is assumed that the user is familiar with Oracle Access Manager and this document does not intend to elaborate on the setup process of the Oracle Access Manager software itself.

Note: these steps may be used as a guide as to how to configure other external authentication solutions which support setting header variables in a similar fashion.

The following diagram shows the main components of the Oracle Access manager:

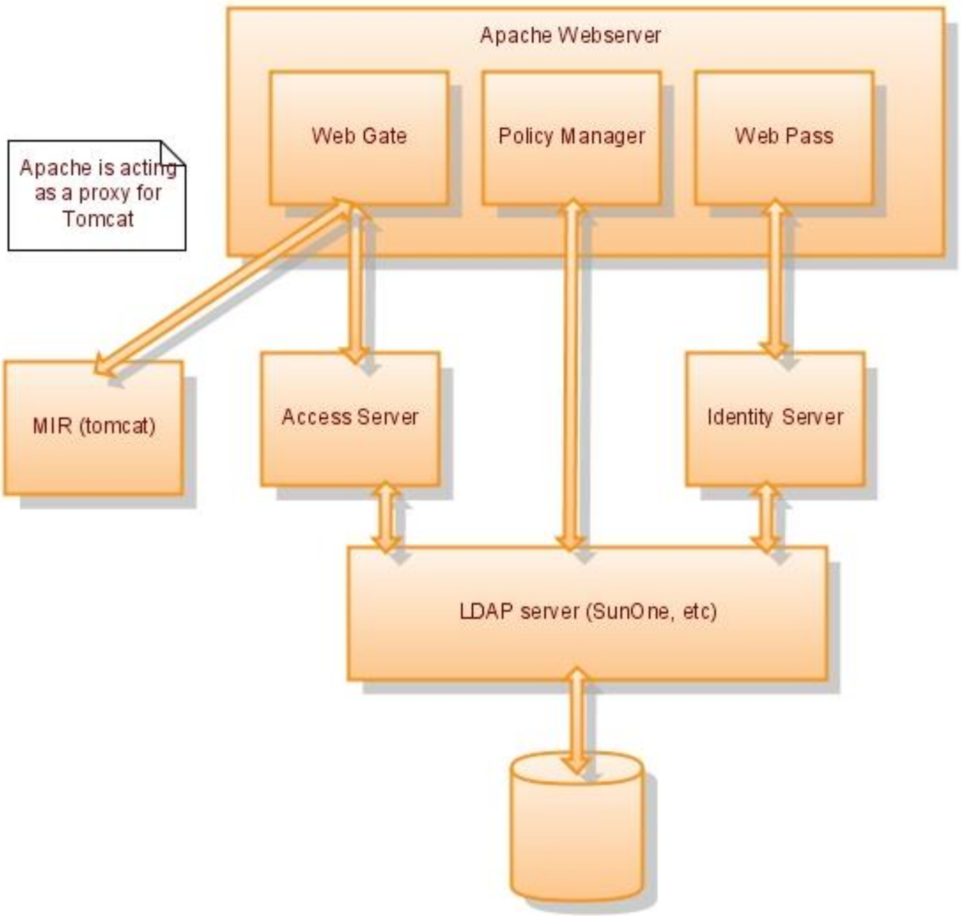


Figure 3 - High-Level Architecture of Oracle Access Manager

3.1.1 OAM Setup

The overall setup process in OAM is as follows. It is provided here for illustrative purposes, as any customer with OAM will already have accomplished these tasks.

- Install an LDAP OAM that is compatible with OAM (e.g., SunOne LDAP directory OAM)

- Install the Apache web OAM. Required for webpass and other web components of OAM.
- Install the identity OAM. This OAM manages all the user and group information. One should use the install options to configure the LDAP OAM and also to make any LDAP schema updates.
- Install WebPass. This component will interact with the identity OAM and act as a front end for any configuration for the Identity OAM. Be sure to also setup the users, groups and other identities.
- Install the Access OAM. During the setup process, give access to the same LDAP configuration. Also update the LDAP schema during the process.
- Install the Policy OAM, which acts as the front end to the Access OAM.
- Install WebGate, which acts as a gateway for protecting all the web resources including Access OAM and Policy OAM. WebGate will intercept requests for any web resource and communicate with the Policy OAM to check the access and also to carry out any further actions.
- Policy Manager is used to setup web gates and also create policies for web resources.

3.1.2 erwin Metadata Management (EMM) Specific Configuration

After completing the above steps (already accomplished for any customer using OAM), one must configure OAM to transfer to erwin Metadata Management (EMM) and provide the correct HTTP header variables.

1. Use the following link to setup a proxy Apache webOAM for the Tomcat OAM where erwin Metadata Management (EMM)is running:

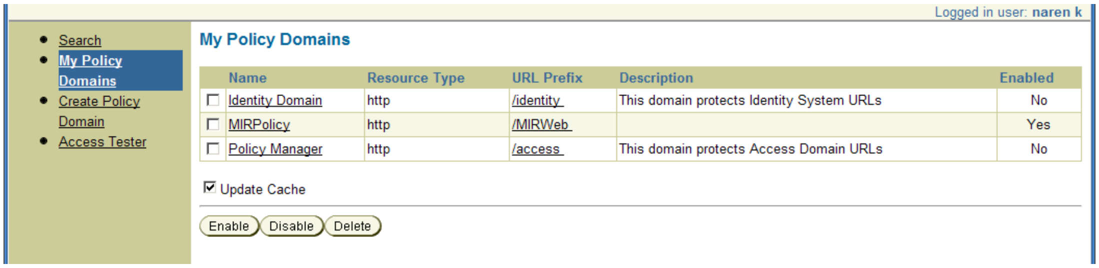
   http://tomcat.apache.org/tomcat-5.5-doc/proxy-howto.html

2. Once this is setup, erwin Metadata Management (EMM) should be accessible through the Apache port. Something like: http://OAM/MM (which will redirect to http://OAM:19980/MM).

3. Create (or reuse) a Web Gate corresponding to the proxy apache web OAM.

4. Create a policy in OAM to allow/deny access to erwin Metadata Management (EMM) and also to map the users to roles.

5. Login to Policy Manager (http://server/access/oblix) to create a new Policy, called MIRPolicy.



My Policy Domains

	Name	Resource Type	URL Prefix	Description	Enabled
□	Identity Domain	http	/identity	This domain protects Identity System URLs	No
□	MIRPolicy	http	/MIRWeb		Yes
□	Policy Manager	http	/access	This domain protects Access Domain URLs	No

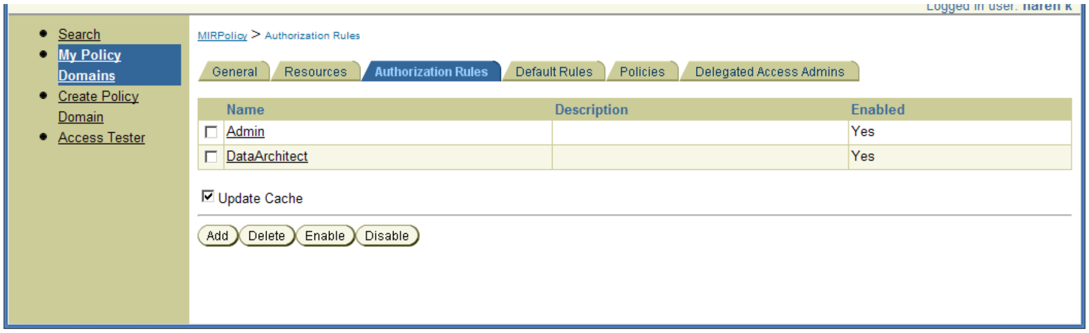
☑ Update Cache

Figure 4 - Creating MIRPolicy for OAM

6. Add a resource type to protect as http and provide the prefix **"/MM".**

7. Setup Authentication: In the default rules, add the appropriate authentication scheme. The "Oracle Access and Identity Basic over LDAP" would work.

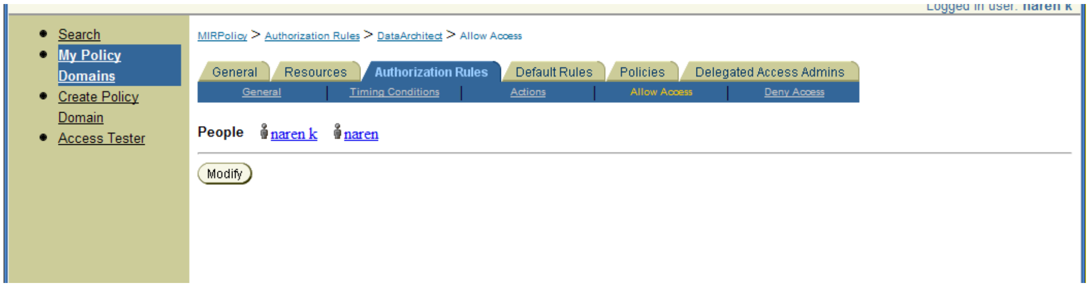
8. Setup Authorization rules: In the authorization rules tab, create a rule for each MIR rolename and allow/deny access to the LDAP users/groups accordingly. Example below:



	Name	Description	Enabled
□	Admin		Yes
□	DataArchitect		Yes

☑ Update Cache

Figure 5 - Defining erwin Metadata Management (EMM) Rolenames for OAM



People naren k naren

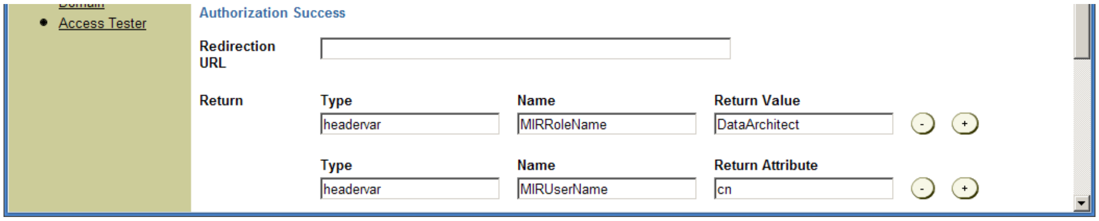
Figure 6 - Allow/Deny Access for OAM

9. In the actions tab, create the following header variables on authorization success for each action.

   a. Create "MIRRoleName" as headerVar with Return Value representing the actual role name in MIR. MIR will parse this information to figure out how to associate a logged in user with MIR role name.

   b. Create MIRUserName as a headerVar and assign the corresponding LDAP attribute. This will be used to personalize things like greeting, etc.

   c. Create other MIR attributes appropriately.



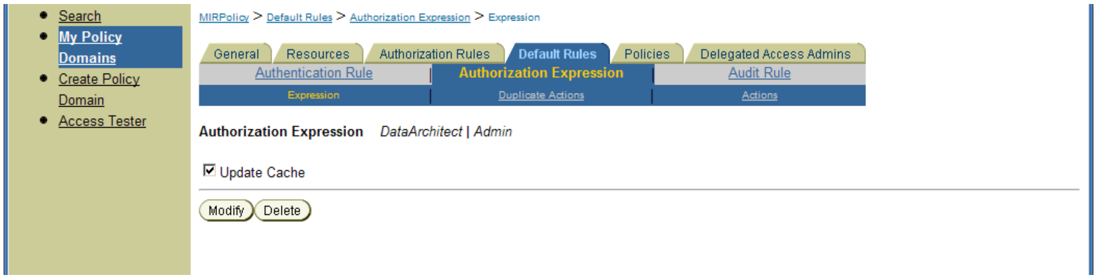
Authorization Success

Redirection URL

Return	Type	Name	Return Value
	headervar	MIRRoleName	DataArchitect
	Type	Name	Return Attribute
	headervar	MIRUserName	cn

Figure 7 - Defining erwin Metadata Management (EMM)/MIR HTTP Header Variable Attributes for OAM

10. Once the rules have been created for authorization, go back to the Default Rules → Authorization Expression and add an expression defining how the authorization rules should be evaluated.

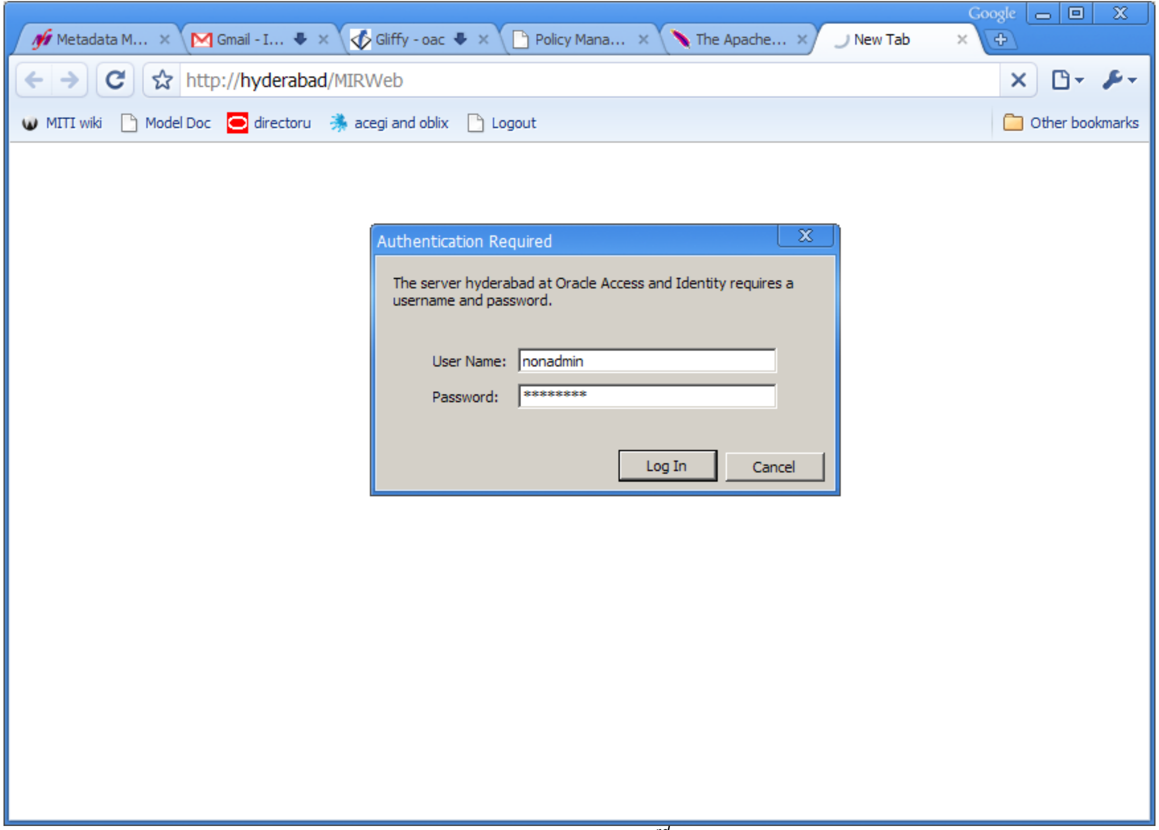


Authorization Expression DataArchitect | Admin

☑ Update Cache

Figure 8 - Defining HTTP Header Variable Expressions for OAM

When a user tries to access erwin Metadata Management (EMM) through the apache proxy OAM (3rd-party mode), if the user has not logged in one should see the following screen (This could be different based on the configuration setup by the OAM administrator):



Authentication Required

The server hyderabad at Oracle Access and Identity requires a username and password.

User Name: nonadmin

Password: *********

Figure 9 - Authentication Presented Using OAM for 3rd-Party Authentication Mode

After providing a valid LDAP username and password, one will be taken directly to the erwin Metadata Management (EMM) repository screen.