Portal – TAM Integration

For TAM – Portal Integration, the first thing is to configure Portal and TAM with the same LDAP Server. To configure Portal with the LDAP Server, run the wizard **configWizard.bat** located at "C:\IBM\WebSphere\wp_profile\PortalServer\wizard" folder and give the appropriate values.

After Portal and TAM are connected with same LDAP Server, some configurations are needed at Portal and TAM side independently.

Configurations which are required at TAM Side

1. Edit the **pd.conf** file located at "C:\IBM\tivoli\Policy Director\etc"

   Change the value **ssl-enabled-fips = false**

2. Edit the **PD.properties** located at "C:\IBM\WebSphere\AppServer\java\jre\PolicyDirector"

   Change the value **ssl-enabled-fips = false**

3. Male sure that in WebSphere Application Server, under Security > SSL certificate and key management, the field "Use the United States Federal Information Processing Standard (FIPS) algorithms" is **not** checked.

4. WebSEAL is installed and configured at TAM side.

Note: If any of the first three steps is missing, then you receive error at Portal Machine when you try to configure it, that is, **"TAM Runtime for Java cannot run with FIPS mode enabled".**

Configurations which are required at Portal Side

Creating the AMJRTE properties file

You must create the AMJRTE properties files before configuring Tivoli Access Manager for authentication, authorization, Credential vault, and/or user provisioning.

Perform the following steps to create the AMJRTE properties file:

1. Use a text editor to open the **wkplc_comp.properties** file:

   Windows located in the *wp_profile_root*\ConfigEngine\properties directory

2. Enter only the following parameters in the **wkplc_comp.properties** file under the AMJRTE connection parameters heading:

   a. For ***wp.ac.impl.PDAdminId***, type the user ID for the administrative Tivoli Access Manager user. (sec_master)

b. For ***wp.ac.impl.PDAdminPwd***, type the password for the administrative Tivoli Access Manager user. (cyber2003)

c. For ***wp.ac.impl.PDPermPath***, type the location of the Tivoli Access Manager AMJRTE properties file. (C:/IBM/WebSphere/AppServer/java/jre/PdPerm.properties)

   Note: When you define path, use **forward slash '/'** instead of back slash '\'

d. For ***wp.ac.impl.PDServerName***, type the unique application name used to create a new Tivoli server in the Access Manager Policy server. (amwp61)

e. For ***wp.ac.impl.SvrSslCfgPort***, type the configuration port for the application name. (7223)

f. For ***wp.ac.impl.SvrSslCfgMode***, type the configuration mode of the SvrSslCfg command. (remote)

g. For ***wp.ac.impl.TamHost***, type the name of the Tivoli Access Manager Policy server used when running PDJrteCfg. (tampoc.rc.com)

h. For ***wp.ac.impl.PDPolicyServerList***, type the host name, port, and priority combinations for your Tivoli Access Manager Policy servers used when running SvrSslCfg. (tampoc.rc.com:7135:1)

i. For ***wp.ac.impl.PDAuthzServerList***, type the host name, port, and priority combination for your Tivoli Access Manager authorization servers. (tampoc.rc.com:7136:1)

j. For the ***wp.ac.impl.PDKeyPath***, type the encryption keys used for the SSL communication between AMJRTE and Tivoli Access Manager. (C:/IBM/WebSphere/AppServer/java/jre/lib/pdperm.ks)

   Note: When you define path, use **forward slash '/'** instead of back slash '\'

3. Save your changes to the **wkplc_comp.properties** file.

4. Run the following task to create the AMJRTE properties file:

   Windows **ConfigEngine.bat run-svrssl-config -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory.

5. **Note:** If the configuration task fails, validate the values in the **wkplc_comp.properties** file.

6. The following files are created:

   Windows C:\ IBM\WebSphere\AppServer\java\jre\ PdPerm.properties

C: \IBM\WebSphere\AppServer\java\jre\lib\ PdPerm.ks

Configuring Tivoli Access Manager for authentication, authorization, and the Credential Vault

Perform the following steps to configure authentication, authorization, and the vault adapter:

1. Run the following validation task to validate that the AMJRTE properties exists:

   Windows **ConfigEngine.bat validate-pdadmin-connection -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory

2. **Note:** If this task fails, run the run-svrssl-config task to create the properties file; see "Creating the AMJRTE properties file" for information about running this task. Please attempt the validate-pdadmin-connection task again. If this task still fails, do not proceed any further. It indicates that portal cannot connect to the TAM server and subsequent tasks will fail.

3. Use a text editor to open the **wkplc_comp.properties** file:

   Windows located in the *wp_profile_root*\ConfigEngine\properties directory

4. Enter only the following parameters in the wkplc_comp.properties file under the Namespace management parameters heading:

   a. For ***wp.ac.impl.EACserverName***, type the Namespace context information to further distinguish externalized portal role names from other role names in the namespace. (WebSphere_Portal)

   b. For ***wp.ac.impl.EACcellName***, type the Namespace context information to further distinguish externalized portal role names from other role names in the namespace. (Leave blank this entry)

   c. For ***wp.ac.impl.EACappname***, type the Namespace context information to further distinguish externalized portal role names from other role names in the namespace. (wps)

   d. For ***wp.ac.impl.reorderRoles***, type false to keep the role order or true to reorder the roles by resource type first. (false)

5. For ***wp.ac.impl.TamHost*** under the SvrSslCfg command parameter heading in the wkplc_comp.properties file, type the Tivoli Access Manager Policy Server used when running PDJrteCfg. (tampoc.rc.com)

6. Enter only the following parameters in the **wkplc_comp.properties** file under the WebSEAL junction parameters heading:

a. For ***wp.ac.impl.JunctionType***, type tcp or ssl to define the type of junction to be created in Tivoli Access Manager. (tcp)

b. For ***wp.ac.impl.JunctionPoint***, type the WebSEAL junction point to the WebSphere Portal installation. (/wpsv61)

Note: This parameter must begin with the / character. For Example /wpsv61

c. For ***wp.ac.impl.WebSealInstance***, type the WebSEAL installation used to create the junction. (default-webseald-tampoc.rc.com)

d. For ***wp.ac.impl.TAICreds***, type the headers inserted by WebSEAL that the TAI uses to identify the request as originating from WebSEAL. (iv-user,iv-creds)

e. For ***wp.ac.impl.JunctionHost***, type the backend server host name to supply to the junction create command. (wpstimtampoc.rc.com)

f. For ***wp.ac.impl.JunctionPort***, type the backend server port to supply to the junction create command. (10050)

7. Enter only the following parameters in the **wkplc_comp.properties** file under the WAS WebSEAL TAI parameters heading:

a. For ***wp.ac.impl.loginId***, type the reverse proxy identity used when you create a TCP junction. (sec_master)

Specifying a user ID: The user ID you specify must be an existing user in the LDAP directory that WebSphere Application Server security can authenticate. The user ID must also be registered and validated in Tivoli Access Manager. WebSEAL requires this user ID to authenticate with WebSphere Application Server security.

b. For ***wp.ac.impl.BaUserName***, type the reverse proxy identity used when you create an SSL junction. (sec_master)

Specifying a user ID: The user ID you specify must be an existing user in the LDAP directory that WebSphere Application Server security can authenticate. The user ID must also be registered and validated in Tivoli Access Manager. WebSEAL requires this user ID to authenticate with WebSphere Application Server security.

c. For ***wp.ac.impl.BaPassword***, type the password for the ***wp.ac.impl.BaUserName***. (cyber2003)

8. Enter only the following parameters in the **wkplc_comp.properties** file under the Portal authorization parameters heading:

a. For ***wp.ac.impl.PDRoot***, type the root object space entry in the Tivoli Access Manager namespace. All Portal roles will be installed under this objectspace

entry. If you will be using Tivoli Access Manager for multiple profiles, choose a unique name for each root objectspace entry to easily distinguish one entry from another profile entry. (/WPv61)

b. For ***wp.ac.impl.PDAction***, type the Custom Action created by the Tivoli Access Manager external authorization plug-in. The combination of the action group and the action determines the Tivoli Access Manager permission string required to assign membership to externalized portal roles. (m)

c. For ***wp.ac.impl.PDActionGroup***, type the Custom Action group created by the Tivoli Access Manager external authorization plug-in. The combination of the action group and the action determines the Tivoli Access Manager permission string required to assign membership to externalized portal roles. ([WP61])

d. For ***wp.ac.impl.PDCreateAcl***, type true to automatically create and attach a Tivoli Access Manager ACL when WebSphere Portal externalizes a role or false to not create and attach a Tivoli Access Manager ACL when WebSphere Portal externalizes a role. (true)

9. Enter only the following parameters in the **wkplc_comp.properties** file under the Portal vault parameters heading:

   a. For ***wp.ac.impl.vaultType***, type the new vault type identifier representing the Tivoli GSO lockbox vault. (AccessManager)

   b. For ***wp.ac.impl.vaultProperties***, type the file to used to configure the vault with Tivoli Access Manager specific user and SSL connection information. (accessmanagervault.properties)

   c. For ***wp.ac.impl.manageResources***, type true if the credential vault or any custom portlets are allowed to create new resource objects in Tivoli Access Manager or type false to allow only the Tivoli Access Manager administrator to define the accessible resources to associate users with from the command line or graphical user interface. (true)

   d. For ***wp.ac.impl.readOnly***, type true to allow credential vault or any custom portlets to modify the secrets stored in Tivoli Access Manager or false to allow only the Tivoli Access Manager administrator to modify the secrets from the command line or graphical user interface. (false)

10. Save your changes to the **wkplc_comp.properties** file.

11. Run the following validation task:

    Windows **ConfigEngine.bat enable-tam-all -DWasPassword=*password*** from the *wp_profile_root*\ConfigEngine directory.

12. **Note:** If the configuration task fails, validate the values in the **wkplc_comp.properties** file.

13. Perform the following steps to **stop and restart the server1 and *WebSphere_Portal* servers**, where server1 is the name of the WebSphere Application Server and *WebSphere_Portal* is the name of the WebSphere Portal server:

    a. Open a command prompt and change to the following directory:

       Windows *wp_profile_root*\bin

    b. Enter the following command to stop the WebSphere Application Server:

       Windows stopServer.bat server1 -username *admin_userid* -password *admin_password*

    c. Enter the following command to stop the *WebSphere_Portal* server, where *WebSphere_Portal* is the name of the WebSphere Portal server:

       Windows stopServer.bat *WebSphere_Portal* -username *admin_userid* -password *admin_password*

    d. Enter the following command to start the WebSphere Application Server:

       Windows startServer.bat server1

    e. Enter the following command to start the *WebSphere_Portal* server, where *WebSphere_Portal* is the name of the WebSphere Portal server:

       Windows startServer.bat *WebSphere_Portal*

Configuring Tivoli Access Manager to perform authentication only

WebSphere Portal and WebSphere Application Server support a TAI (Trust Association Interceptor) that is provided by Tivoli. If you use Tivoli Access Manager to perform authorization for WebSphere Portal, you must also use Tivoli Access Manager to perform the authentication. Using Tivoli Access Manager to perform only authorization is not supported.

This configuration is done at TAM Machine where WebSEAL is installed and configured

1. Enter the following tasks on the **pdadmin** command line to create the trusted user account:

   Note: To prevent potential vulnerabilities, do not use the sec_master or *wpsadmin* users for the trusted user account. The trusted user account should be for the TAI only.

a. pdadmin> user create *webseal_userid webseal_userid_DN firstname surname password*

   For Example

   pdadmin> user create *webseal cn=webseal,ou=Austin,o=ibm,c=us web seal cyber2003*

b. pdadmin> user modify *webseal_userid* account-valid yes

   For Example

   pdadmin> user modify *webseal* account-valid yes

2. Run the following validation task to validate that the AMJRTE properties exists:

   Windows **ConfigEngine.bat validate-pdadmin-connection -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory

3. **Note:** If this task completes successfully then skip task 4 to 7 and perform task 8. If this task fails, run the run-svrssl-config task to create the properties file; see "Creating the AMJRTE properties file" for information about running this task. Please attempt the validate-pdadmin-connection task again. If this task still fails, do not proceed any further. It indicates that portal cannot connect to the TAM server and subsequent tasks will fail.

4. Use a text editor to open the **wkplc_comp.properties** file:

   Windows located in the *wp_profile_root*\ConfigEngine\properties directory

5. Enter only the following parameters in the **wkplc_comp.properties** file under the WebSEAL junction parameters heading:

   a. For ***wp.ac.impl.JunctionType***, type tcp or ssl to define the type of junction to be created in Tivoli Access Manager. (tcp)

   b. For ***wp.ac.impl.JunctionPoint***, type the WebSEAL junction point to the WebSphere Portal installation. (/wpsv61)

      Note: This parameter must begin with the '/' character.

   c. ***wp.ac.impl.WebSealInstance***, type the WebSEAL installation used to create the junction. (default-webseald-tampoc.rc.com)

   d. For ***wp.ac.impl.TAICreds***, type the headers inserted by WebSEAL that the TAI uses to identify the request as originating from WebSEAL. (iv-user,iv-creds)

e. For ***wp.ac.impl.JunctionHost***, type the backend server host name to supply to the junction create command. (wpstimtampoc.rc.com)

f. For ***wp.ac.impl.JunctionPort***, type the backend server port to supply to the junction create command. (10050)

6. Enter only the following parameters in the **wkplc_comp.properties** file under the WAS WebSEAL TAI parameters heading:

   a. For ***wp.ac.impl.loginId***, type the reverse proxy identity used when you create a TCP junction. (sec_master)

      Specifying a user ID: The user ID you specify must be an existing user in the LDAP directory that WebSphere Application Server security can authenticate. The user ID must also be registered and validated in Tivoli Access Manager. WebSEAL requires this user ID to authenticate with WebSphere Application Server security.

   b. For ***wp.ac.impl.BaUserName***, type the reverse proxy identity used when you create an SSL junction. (sec_master)

      Specifying a user ID: The user ID you specify must be an existing user in the LDAP directory that WebSphere Application Server security can authenticate. The user ID must also be registered and validated in Tivoli Access Manager. WebSEAL requires this user ID to authenticate with WebSphere Application Server security.

   c. For ***wp.ac.impl.BaPassword***, type the password for the ***wp.ac.impl.BaUserName***. (cyber2003)

7. Save your changes to the **wkplc_comp.properties** file.

8. Run the following task to configure TAI for Tivoli Access Manager:

   Windows **ConfigEngine.bat enable-tam-tai -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory.

9. Restart all required servers to propagate your changes.

10. If you created a TCP junction in the previous step, go to the WebSEAL machine and edit the **webseald-*instance*.conf** file for the appropriate WebSEAL instance. An example is **webseald-default.conf**. This sets the **basicauth-dummy-passwd** value to the password for the ID that WebSEAL uses to identify itself to WebSphere Application Server. This user ID and password were created in an earlier step (cyber2003). Stop and start the WebSEAL server before continuing.

11. Edit the **webseald-*instance*.conf** file and change the ***process-root-requests*** property value to **filter** to avoid problems with WebSEAL processing.

Configuring Tivoli Access Manager to perform authorization

You can configure IBM® Tivoli® Access Manager to perform authorization as an independent task from configuring Tivoli Access Manager to perform authentication, but you must configure both tasks. Using Tivoli Access Manager to perform only authorization is not supported.

Perform the steps in **Configuring Tivoli Access Manager to perform authentication only** before configuring Tivoli Access Manager to perform authorization.

Perform the following steps to configure Tivoli Access Manager to perform authorization:

1. Run the following validation task to validate that the AMJRTE properties exists:

   Windows **ConfigEngine.bat validate-pdadmin-connection -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory

2. **Note:** If this task completes successfully then skip task 3 to 5 and perform task 6. If this task fails, run the run-svrssl-config task to create the properties file; see "Creating the AMJRTE properties file" for information about running this task. Please attempt the validate-pdadmin-connection task again. If this task still fails, do not proceed any further. It indicates that portal cannot connect to the TAM server and subsequent tasks will fail.

3. Enter only the following parameters in the **wkplc_comp.properties** file under the Namespace management parameters heading:

   a. For ***wp.ac.impl.EACserverName***, type the Namespace context information to further distinguish externalized portal role names from other role names in the namespace. (WebSphere_Portal)

   b. For ***wp.ac.impl.EACcellName***, type the Namespace context information to further distinguish externalized portal role names from other role names in the namespace. (Leave blank this entry)

   c. For ***wp.ac.impl.EACappname***, type the Namespace context information to further distinguish externalized portal role names from other role names in the namespace. (wps)

   d. For ***wp.ac.impl.reorderRoles***, type false to keep the role order or true to reorder the roles by resource type first. (false)

4. Enter only the following parameters in the wkplc_comp.properties file under the Portal authorization parameters heading:

   a. For ***wp.ac.impl.PDRoot***, type the root objectspace entry in the Tivoli Access Manager namespace. All Portal roles will be installed under this objectspace entry. If you will be using Tivoli Access Manager for multiple profiles, choose

a unique name for each root objectspace entry to easily distinguish one entry from another profile entry. (/WPv61)

b. For ***wp.ac.impl.PDAction***, type the Custom Action created by the Tivoli Access Manager external authorization plug-in. The combination of the action group and the action determines the Tivoli Access Manager permission string required to assign membership to externalized portal roles. (m)

c. For ***wp.ac.impl.PDActionGroup***, type the Custom Action group created by the Tivoli Access Manager external authorization plug-in. The combination of the action group and the action determines the Tivoli Access Manager permission string required to assign membership to externalized portal roles. ([WP61])

d. For ***wp.ac.impl.PDCreateAcl***, type true to automatically create and attach a Tivoli Access Manager ACL when WebSphere Portal externalizes a role or false to not create and attach a Tivoli Access Manager ACL when WebSphere Portal externalizes a role. (true)

5. Save your changes to the **wkplc_comp.properties** file.

6. Run the following validation task:

   Windows **ConfigEngine.bat enable-tam-authorization -DWasPassword=*password*** from the *wp_profile_root*\ConfigEngine directory.

7. **Note:** If the configuration task fails, validate the values in the **wkplc_comp.properties** file.

8. Stop and restart the server1 and WebSphere_Portal servers.

Configuring the Credential Vault adapter for Tivoli Access Manager

Note: Users who are storing credentials in the accessmanagervault.properties file must be defined in Tivoli Access Manager as global sign-on (GSO) users.

Perform the following steps to configure the Tivoli Access Manager vault adapter that is packaged with WebSphere Portal:

1. Run the following validation task to validate that the AMJRTE properties exists:

   Windows **ConfigEngine.bat validate-pdadmin-connection -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory

   Note: If this task fails, run the run-svrssl-config task to create the properties file; see "Creating the AMJRTE properties file" for information about running

this task. Please attempt the validate-pdadmin-connection task again. If this task still fails, do not proceed any further. It indicates that portal can not connect to the TAM server and subsequent tasks will fail.

2. Run the following task to create and populate the *wp_profile_root*/PortalServer/config/config/accessmanagervault.properties file:

   Windows **ConfigEngine.bat enable-tam-vault -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory.

3. Stop and restart the server1 and WebSphere_Portal servers.

Enabling user provisioning

When users are created in WebSphere Portal, they are not automatically imported into Tivoli Access Manager. Enabling automatic user provisioning to Tivoli Access Manager changes this behavior. Once this feature is enabled, users are automatically imported into Tivoli Access Manager whenever they are created in WebSphere Portal. When user provisioning to Tivoli Access Manager, anyone with access to the public URL can become an active user in Tivoli Access Manager as long as the self-registration feature remains enabled.

Note: There are two ways to create users in WebSphere Portal:

- **Self-registration:** This feature is enabled by default.
- **Manage Users and Groups portlet:** Administrators can use this portlet to create WebSphere Portal users.

Perform the following steps to enable user provisioning within Tivoli Access Manager:

1. Run the following validation task to validate that the AMJRTE properties exists:

   Windows **ConfigEngine.bat validate-pdadmin-connection -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory

2. **Note:** If this task fails, run the run-svrssl-config task to create the properties file; see "Creating the AMJRTE properties file" for information about running this task. Please attempt the validate-pdadmin-connection task again. If this task still fails, do not proceed any further. It indicates that portal can not connect to the TAM server and subsequent tasks will fail.

3. Run the following task to enable user provisioning:

Windows **ConfigEngine.bat enable-tam-userprov -DWasPassword=*password* -Dwp.ac.impl.PDAdminPwd=*password*** from the *wp_profile_root*\ConfigEngine directory.

4. Stop and restart the server1 and WebSphere_Portal servers.

Verifying Tivoli Access Manager is working

After configuring IBM® WebSphere® Portal to use Tivoli Access Manager for externalized authorization, you should verify that it is working properly before continuing with any additional configuration tasks.

Perform the following steps to verify that Tivoli Access Manager is working properly:

1. Perform the following steps to verify that at least one user, typically the administrator, has the **Administrator@VIRTUAL/EXTERNAL ACCESS CONTROL_1** role:

    a. Enter the

    pdadmin> acl show WPS_Administrator-VIRTUAL_wps-EXTERNAL_ACCESS_CONTROL_1

    command on the pdadmin command line to verify that the administrator and administrator group have the Administrator@VIRTUAL/EXTERNAL ACCESS CONTROL_1 role.

    For example

    pdadmin> acl show WPv61_Administrator-VIRTUAL_wps-EXTERNAL_ACCESS_CONTROL_1

    b. **Optional:** Enter the following commands to add the administrator to the Administrator@VIRTUAL/EXTERNAL ACCESS CONTROL_1 role if no entry is found:

    - **pdadmin> acl modify WPS_Administrator-VIRTUAL_wps-EXTERNAL_ACCESS_CONTROL_1 set user *wpsadmin* T[WPS]m**

      For Example

      pdadmin> acl modify WPv61_Administrator-VIRTUAL_wps-EXTERNAL_ACCESS_CONTROL_1 set user *wpsadmin* T[WPv61]m

    - **pdadmin> acl modify WPS_Administrator-VIRTUAL_wps-EXTERNAL_ACCESS_CONTROL_1 set group *wpsadmins* T[WPS]m**

      For example

pdadmin> acl modify WPv61_Administrator-VIRTUAL_wps-EXTERNAL_ACCESS_CONTROL_1 set group *wpsadmins* T[WPv61]m

where *wpsadmin* is the administrator user ID and *wpsadmins* is the administrator group.

2. Perform the following steps from the Resource Permissions portlet:
    a. Select a resource type.
    b. Click the Assign Access icon for the specific resource.
    c. Click the Edit Role icon for a role that you want to externalize.
    d. Click Add to explicitly assign at least one user or group to your chosen role for the resource.
    e. Click Search for Users or User Groups or click the pull down for the Search by option where the default is set to All available to select specific users or user groups. Then click OK. An informational message box should display the message that members were successfully added to the role.
    f. Optional: Explicitly assign additional roles. If you do not assign at least one user or group to each role type for the resource, you must use the external security manager interface to create this role type later. For example, if you do not assign any users or groups to the Editor role type for the resource, then you must use the external security manager interface to create the Editor role type later.
    g. Click the Externalize icon for the resource. These steps move every role that is defined for each resource you assigned to the Tivoli Access Manager protected object space. One ACL is created for each externalized role.
3. Add users to the ACLs that are attached to the role types on that resource by using either the Tivoli Access Manager GUI or the pdadmin command line.

    Note: If you log on to WebSphere Portal for administration purposes and you intend to externalize resources to Tivoli Access Manager, remember the following:

    - You must be a member of the wpsadmins group
    - The wpsadmins group must appear in the VIRTUAL/EXTERNAL_ACCESS_CONTROL_1 ACL

Note: If TAM groups are not appeared in the Portal Admin Console, then do the following steps at Portal Machine.

Edit the wimconfig.xml located at C:\IBM\WebSphere\wp_profile\config\cells\wpstimtampoc\wim\config the Group entry looks like this:

<config:ldapEntityTypes name="Group" searchFilter="">
<config:objectClasses>**groupOfUniqueNames**</config:objectClasses>
</config:ldapEntityTypes>

The group object class should be adjusted. Please do the following:

1) Stop the Portal
2) Take a backup of wimconfig.xml

3) Edit wimconfig.xml so the section above looks like this:

<config:ldapEntityTypes name="Group" searchFilter="">
<config:objectClasses>**groupOfNames**</config:objectClasses>
</config:ldapEntityTypes>

4) Restart the portal and try to find wasadmins in the Users and Groups portlet

Reference link:

- http://publib.boulder.ibm.com/infocenter/wpexpdoc/v6r1/index.jsp?topic=/com.ibm.wp.exp.doc_v6101/security/run_svrssl_config.html

IBM Global Services
Route 100
Somers, NY 10589
U.S.A.
Produced in the United States of America
08-10