Integrating OID with Active Directory and WNA

Hari Muthuswamy

CTO, Eagle Business Solutions

May 10, 2007

Suncoast Oracle User Group
Tampa Convention Center

What is SSO?

- **“Single Sign-On”** (SSO) is a session/user authentication process that permits a user to enter one name and password in order to access multiple applications
- Oracle’s SSO is typically utilizes OID (Oracle Internet Directory)

Why do we need SSO?

- Ease and Convenience for the end user
  - Has to remember only one Password
  - No cheat sheets required
- Reduce Security Threats
  - Enter password only once
  - No cheat sheets of passwords
- Reduced Maintenance
  - Reduced number of forgotten password issues that IT may need to deal with

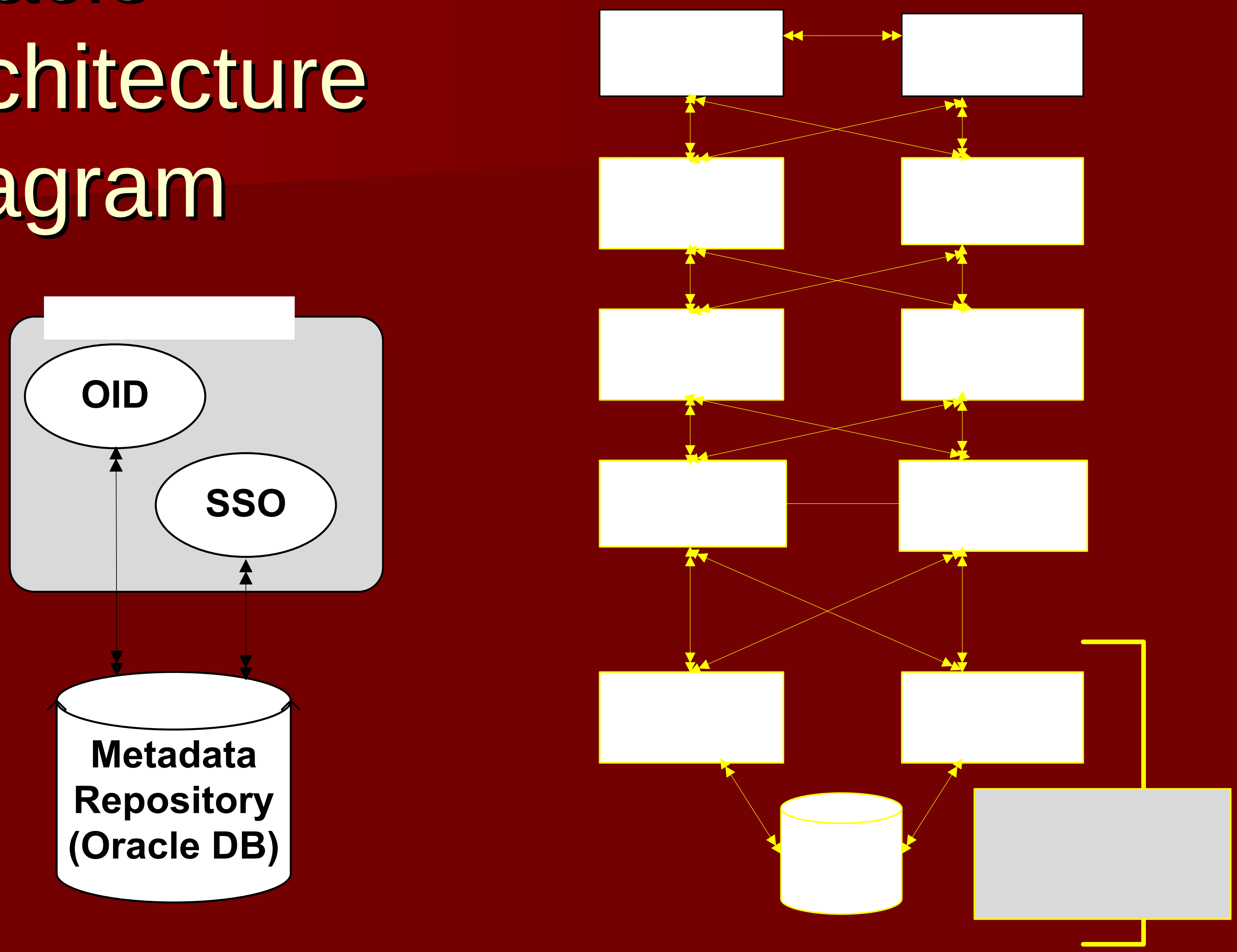
OID

- Oracle Internet Directory
- LDAP V3 Compliant Directory
- Centrally stores all the user, application and resource information in a typical Oracle Enterprise Architecture
- Oracle Database stores all information
- Extremely fast retrieval of data

Active Directory

- Microsoft’s Directory Services Product
- LDAP based Directory used to support Windows Domain authentication and permissions
- Companies often use AD as the primary record of authority

Oracle Architecture Diagram



AD-OID Sync

- Why do we need it?
- Relevance to SSO
- Types of Synchronizations
  - One way or
  - Bi-directional
- External Authentication
- Steps involved

AD-OID Sync - Why do we need it?

- *With multiple Directories in use, one must be the “Record of Authority”, in many cases it is MS Active Directory. To avoid redundant data entry and conflicts between systems, automated synchronization is a MUST*
- HR processes update AD
- You want to use Oracle AS SSO and OID for all you Single Sign-On needs
- You want all the user data to be in Sync

AD-OID Sync - Relevance to SSO

- In a typical organization, Microsoft Windows is the OS on desktops
- AD is the directory behind the network login, in many of these organizations
- Once a user logs into the network, we want to use that authentication to let them access all other resources
- AD-OID Sync is the first step towards it

AD-OID Sync - Types of Synchronization

- One Way
  - – In this type of Synchronization AD usually is the master and all changes are made only in AD and the changes get propagated to OID
- Bi-directional
  - – In this type of Synchronization, User are added, deleted and Updated in both AD and OID and both the directories are Synchronized with each other

What is External Authentication?

- It is the process by which an Oracle Application Server user receives authentication from a remote directory

Why do we need External Authentication

- It lets your Oracle 10g Application Server users to authenticate using their user credentials stored in Active Directory
- AD Import connector component that we use to Migrate Users from AD to OID cannot migrate hashed passwords from AD
- MS uses a proprietary hashing Algorithm called Unicode password encryption that is not supported in OID
- OID supports MD5, MD4, SHA, SSHA, and Crypt and MS does not support any of these

Steps Involved.

- AD-OID Sync
  - Configuring the Import Connector (ActiveChgImp Agent)
- Configuring the External Authentication Plug-in
- Configuring WNA

Configuring the Import connector

- Creating attribute and domain mapping rules
- Uploading mapping file to your Import profile
- Grant access for AD Group synchronization

Creating the attribute and domain mapping rules

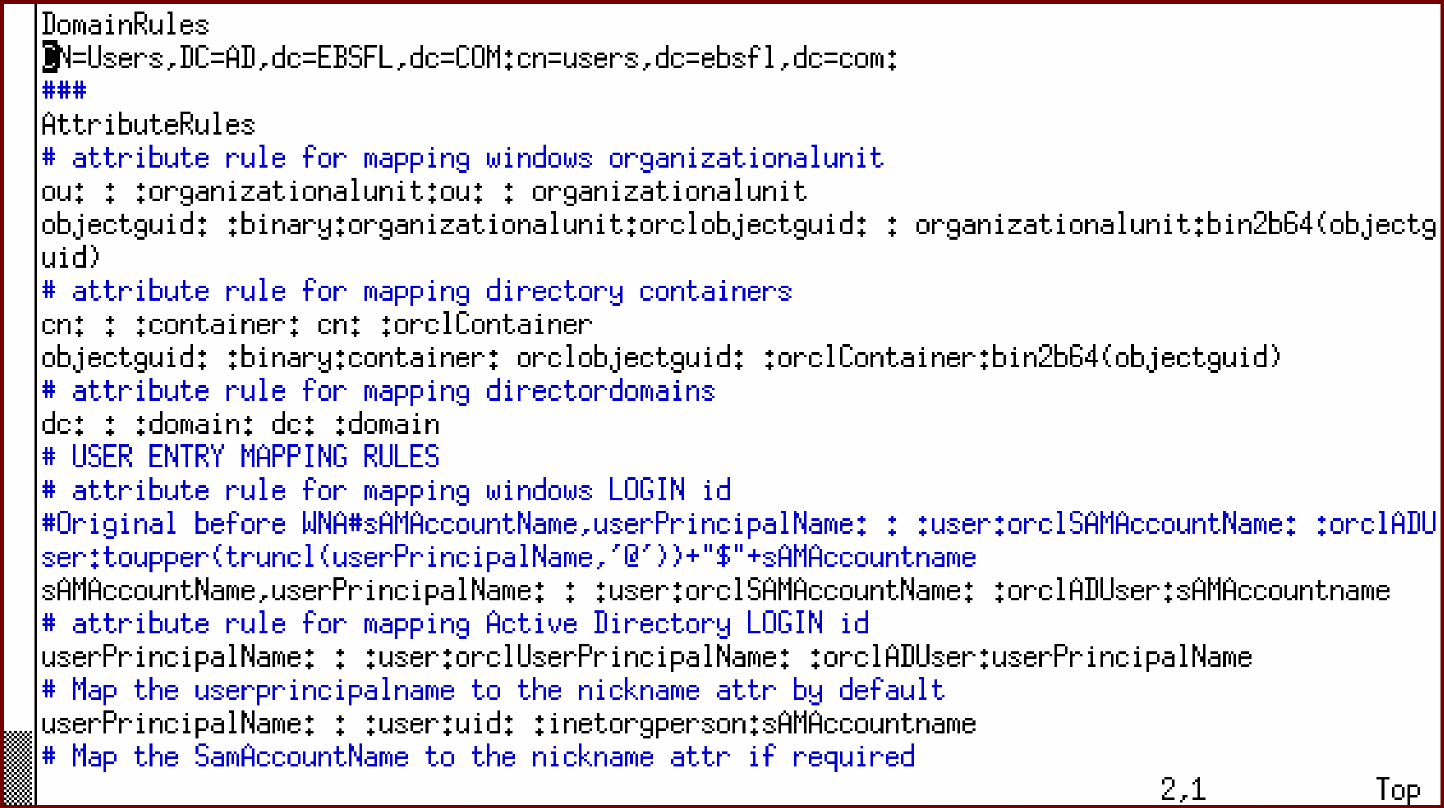
- First step in configuring the Import connection is setting up the attribute and domain mapping rules
- Sample file called activechg.map
- This file has two sections
  - – Domain Rules section
  - – Attribute Rules section

Creating the attribute and domain mapping rules

- The DomainRules tells the DIP server the location of users and groups in the AD server and where to make those changes in the OID server
- The AttributeRules tell the DIP server which attributes on the AD server side will be mapped to which attributes on the OID server side

Sample activechg.map

cd $ORACLE_HOME/ldap/odi/conf
cp activechg.map.master activechg.map



```
DomainRules
CN=Users,DC=AD,dc=EBSFL,dc=COM:cn=users,dc=ebsfl,dc=com:
###
AttributeRules
# attribute rule for mapping windows organizationalunit
ou: : :organizationalunit:ou: : organizationalunit
objectguid: :binary:organizationalunit:orclobjectguid: : organizationalunit:bin2b64(objectg
uid)
# attribute rule for mapping directory containers
cn: : :container: cn: :orclContainer
objectguid: :binary:container: orclobjectguid: :orclContainer:bin2b64(objectguid)
# attribute rule for mapping directordomains
dc: : :domain: dc: :domain
# USER ENTRY MAPPING RULES
# attribute rule for mapping windows LOGIN id
#Original before WNA#sAMAccountName,userPrincipalName: : :user:orclSAMAccountName: :orclADU
ser:toupper(truncl(userPrincipalName,'@'))+"$"+sAMAccountname
sAMAccountName,userPrincipalName: : :user:orclSAMAccountName: :orclADUser:sAMAccountname
# attribute rule for mapping Active Directory LOGIN id
userPrincipalName: : :user:orclUserPrincipalName: :orclADUser:userPrincipalName
# Map the userprincipalname to the nickname attr by default
userPrincipalName: : :user:uid: :inetorgperson:sAMAccountname
# Map the SamAccountName to the nickname attr if required
                                                                      2,1           Top
```

The left DN represents the location of users in the Active directory *source*.
The right represents the where the update will happen in OID.

Uploading the mapping file to your Import profile

- For DIP Server to use the mapping file, it has to be loaded into the import connector profile
- dipassistant mp -host hostname.domain.com -port 389 -passwd welcome1 -profile ActiveChgImp odip.profile.mapfile=/oracle/home/ldap/odi/conf/activechg.map
- Host is FQDN:port number and password for OID

Grant access permission for AD Group synchronization

- AD stores the groups in the users container
- Extra access control policy is needed to allow groups to be created in the users container in OID
- Grantrole.ldif is a sample file that needs small modifications to issue the grant
- ldapmodify -p oid_portnum -h oid_host_name -D "cn=orcladmin" -w orcladmin_pass -f /grantrole.ldif

Migrating AD users and groups to OID

- Migrating users and groups is often referred to as "Bootstrapping" in the OID administrators guide
- A program called the "dipassistant" is used to perform to migrate your Microsoft users and groups to OID
- The dipassistant uses a file called ldp2ldp.properties to migrate the users and groups. This file is located in your $ORACLE_HOME/ldap/odi/samples directory
- Copy the file "ldp2ldp.properties" to a new file and open the new file in a text editor and make the following changes:

Migrating AD users and groups to OID

- Set the "odip.bootstrap.srctype" to "LDAP"
- Set the "odip.bootstrap.srcurl" to the fully qualified domain name and port number where the Active Directory server is running
- Set the "odip.bootstrap.srcdn" to the administrative account on the Active Directory server which has permission to read the directory. Example:

  administrator@ebsfl.com
- Set the "odip.bootstrap.srcpasswd" value to the Active Directory administrators password.
- Set the "odip.bootstrap.desttype" to "LDAP"

Migrating AD users and groups to OID

- Set the "odip.bootstrap.desturl to the fully qualified domain name and port number where the OID server is running. Remember to use a "**:**" to separate the host name and port number

  Example: snake.ebsfl.com:389

- Set the "odip.bootstrap.destdn" to the OID super user account "cn=orcladmin“
- Set the "odip.bootstrap.destpasswd" to the OID super users password
- Set the "odip.bootstrap.mapfile" value to the full path where the AD import mapping file created earlier in this section is located. In our example we used "activechg.map".

Migrating AD users and groups to OID

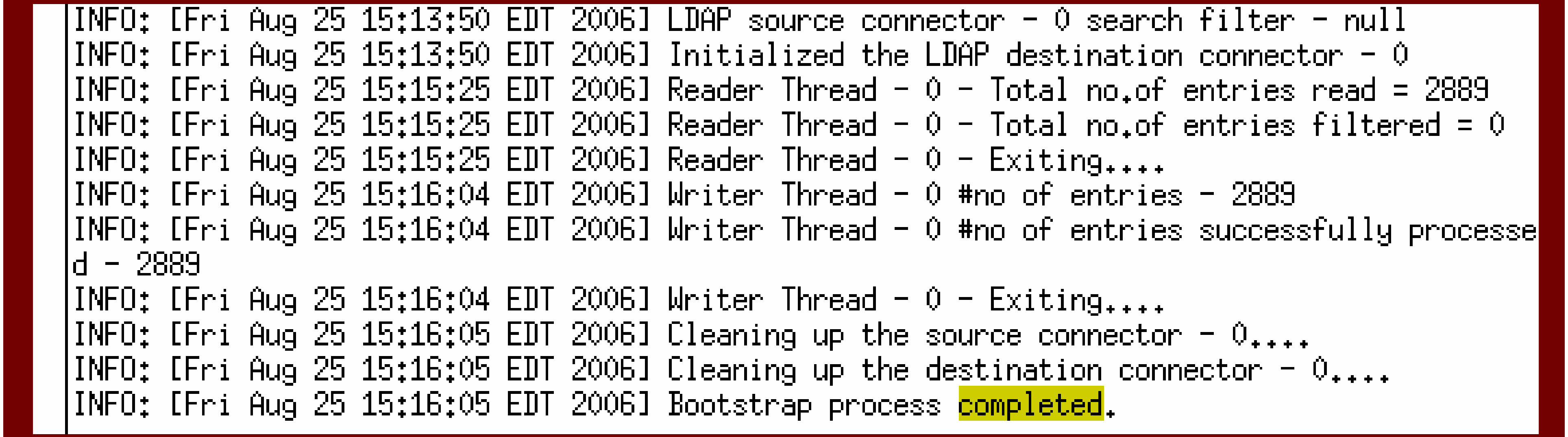
- Set the odip.bootstrap.logfile value to whatever location you want your log files located in your file system
- Set the odip.bootstrap.logseverity level to record your desired level of error capturing. In this example we will set the level to 15 to capture all errors
- Set the odip.bootstrap.trcfile to the location where you want your trace file located

Migrating the AD users and groups to OID

-
  ```
  dipassistant bs -cfg $ORACLE_HOME/ldap/odi/samples/ad2oid.properties
  ```

Bootstrapping result

- The result looks like this after bootstrapping.

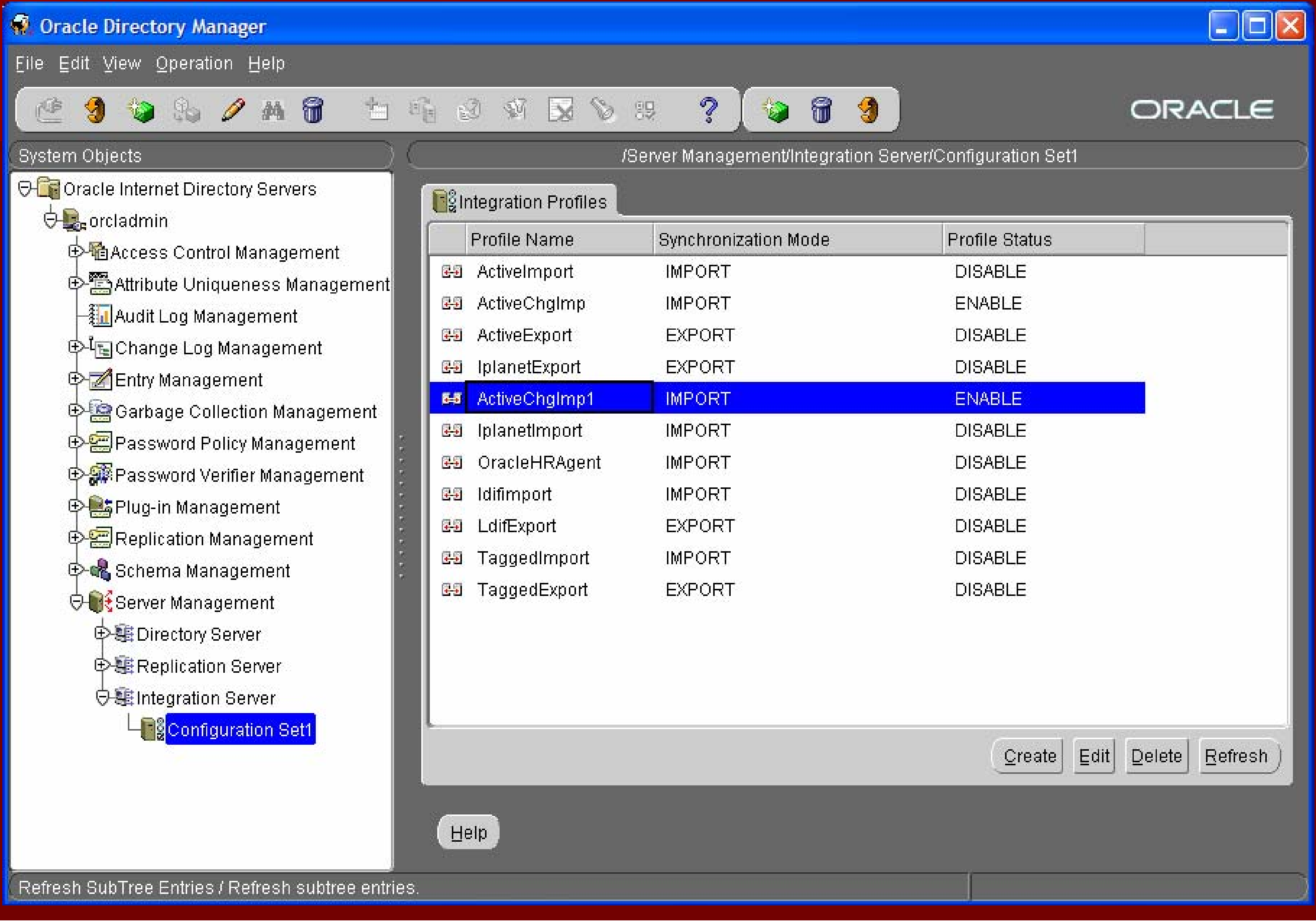


```
INFO: [Fri Aug 25 15:13:50 EDT 2006] LDAP source connector - 0 search filter - null
INFO: [Fri Aug 25 15:13:50 EDT 2006] Initialized the LDAP destination connector - 0
INFO: [Fri Aug 25 15:15:25 EDT 2006] Reader Thread - 0 - Total no.of entries read = 2889
INFO: [Fri Aug 25 15:15:25 EDT 2006] Reader Thread - 0 - Total no.of entries filtered = 0
INFO: [Fri Aug 25 15:15:25 EDT 2006] Reader Thread - 0 - Exiting....
INFO: [Fri Aug 25 15:16:04 EDT 2006] Writer Thread - 0 #no of entries - 2889
INFO: [Fri Aug 25 15:16:04 EDT 2006] Writer Thread - 0 #no of entries successfully processe
d - 2889
INFO: [Fri Aug 25 15:16:04 EDT 2006] Writer Thread - 0 - Exiting....
INFO: [Fri Aug 25 15:16:05 EDT 2006] Cleaning up the source connector - 0....
INFO: [Fri Aug 25 15:16:05 EDT 2006] Cleaning up the destination connector - 0....
INFO: [Fri Aug 25 15:16:05 EDT 2006] Bootstrap process completed.
```

Import Agent Configuration

- **# oidadmin**
- **Once your have successfully logged into ODM, navigate through the DIT to "Server Management -> Integration Servers"**
- **Click on "Configuration Set1". You will see all of the default DIP agent profiles listed on the right**
- **Double click on the agent profile named "ActiveChgImp"**

Import Agent Configuration

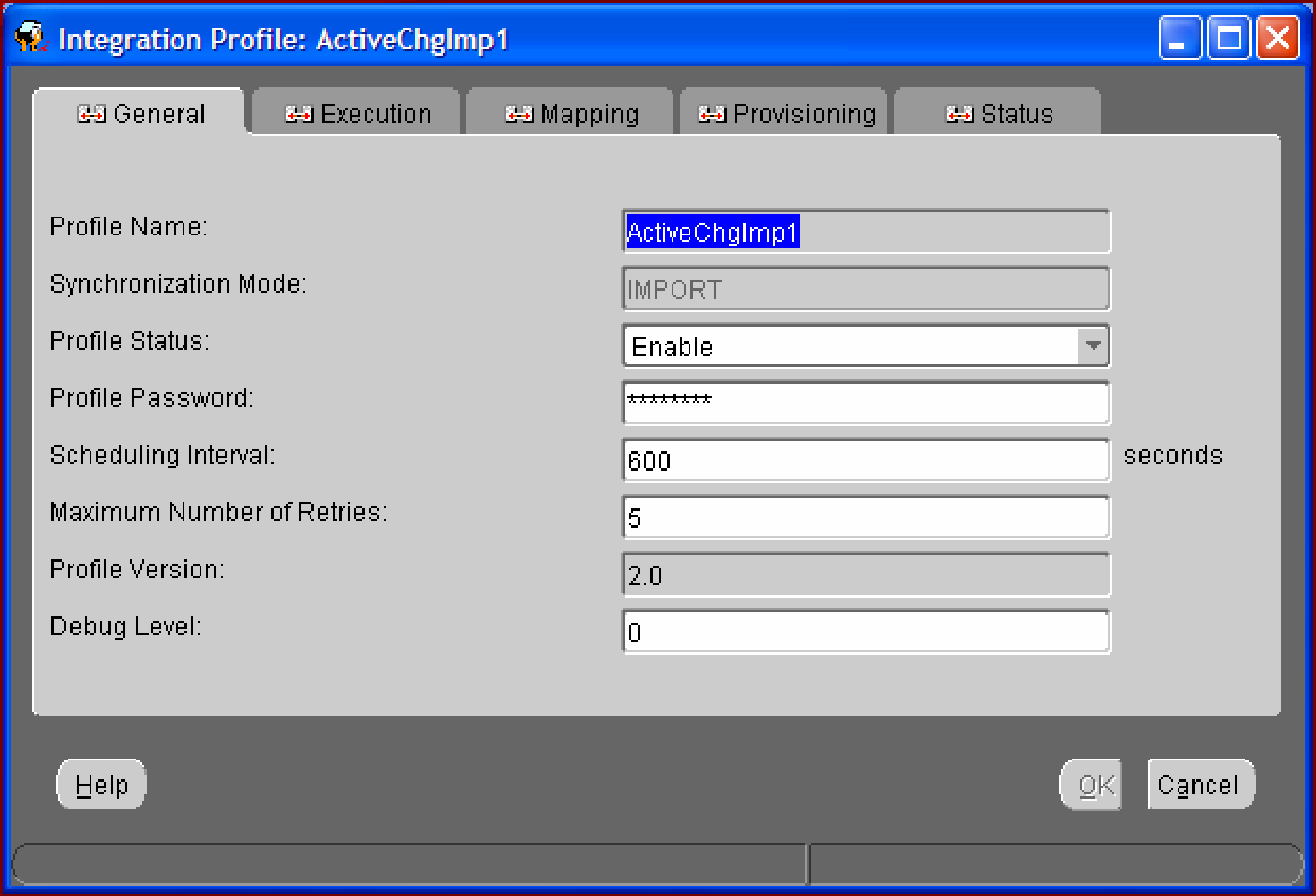


Profile Name	Synchronization Mode	Profile Status
ActiveImport	IMPORT	DISABLE
ActiveChgImp	IMPORT	ENABLE
ActiveExport	EXPORT	DISABLE
IplanetExport	EXPORT	DISABLE
ActiveChgImp1	IMPORT	ENABLE
IplanetImport	IMPORT	DISABLE
OracleHRAgent	IMPORT	DISABLE
Idifimport	IMPORT	DISABLE
LdifExport	EXPORT	DISABLE
TaggedImport	IMPORT	DISABLE
TaggedExport	EXPORT	DISABLE

Refresh SubTree Entries / Refresh subtree entries.

Import Agent Configuration

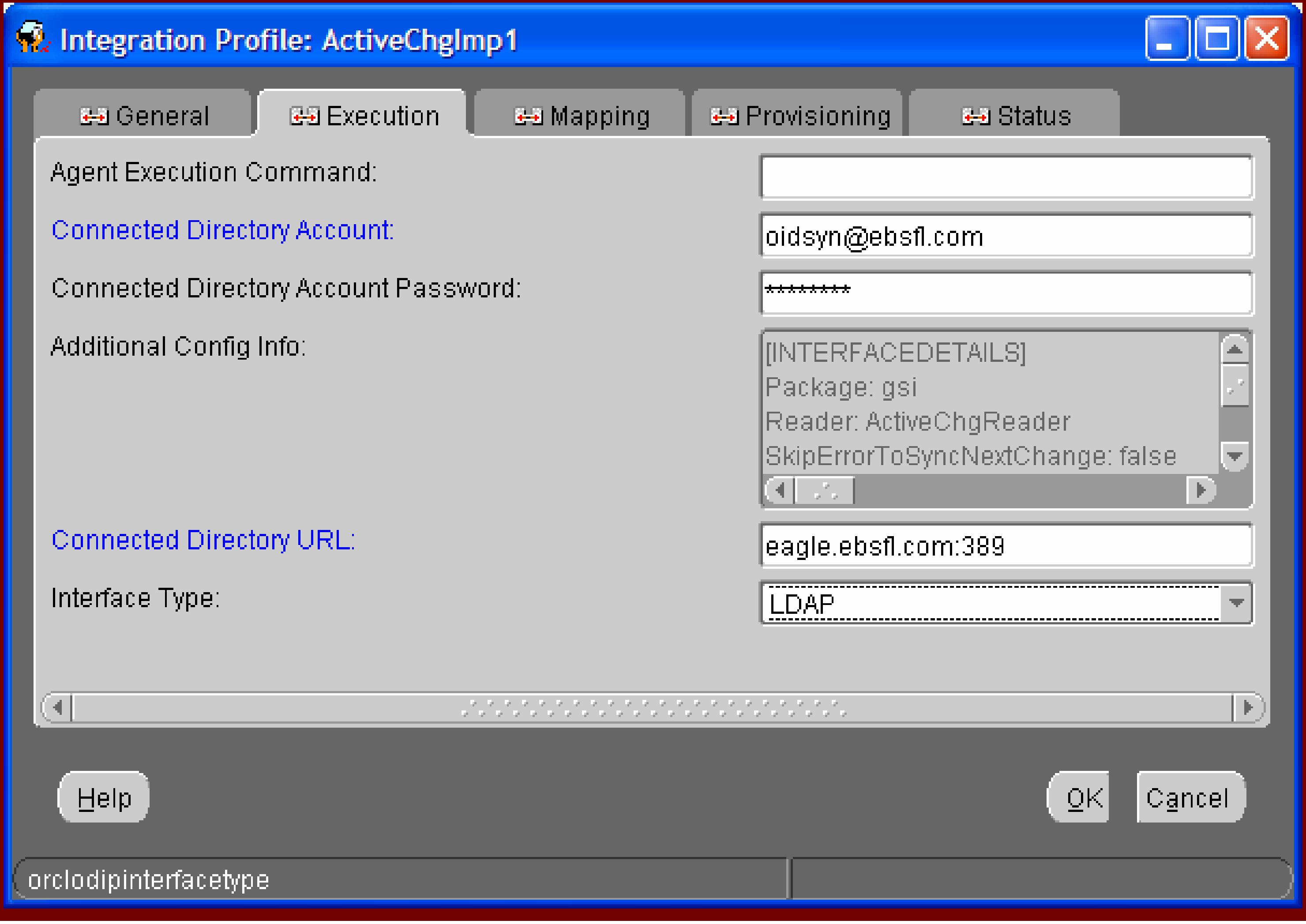
Configuring the General tab



Profile Name:	ActiveChgImp1	
Synchronization Mode:	IMPORT	
Profile Status:	Enable	
Profile Password:	*********	
Scheduling Interval:	600	seconds
Maximum Number of Retries:	5	
Profile Version:	2.0	
Debug Level:	0	

Import Agent Configuration

Configuring the Execution tab



Agent Execution Command:

Connected Directory Account: oidsyn@ebsfl.com

Connected Directory Account Password: *********

Additional Config Info:

```
[INTERFACEDETAILS]
Package: gsi
Reader: ActiveChgReader
SkipErrorToSyncNextChange: false
```

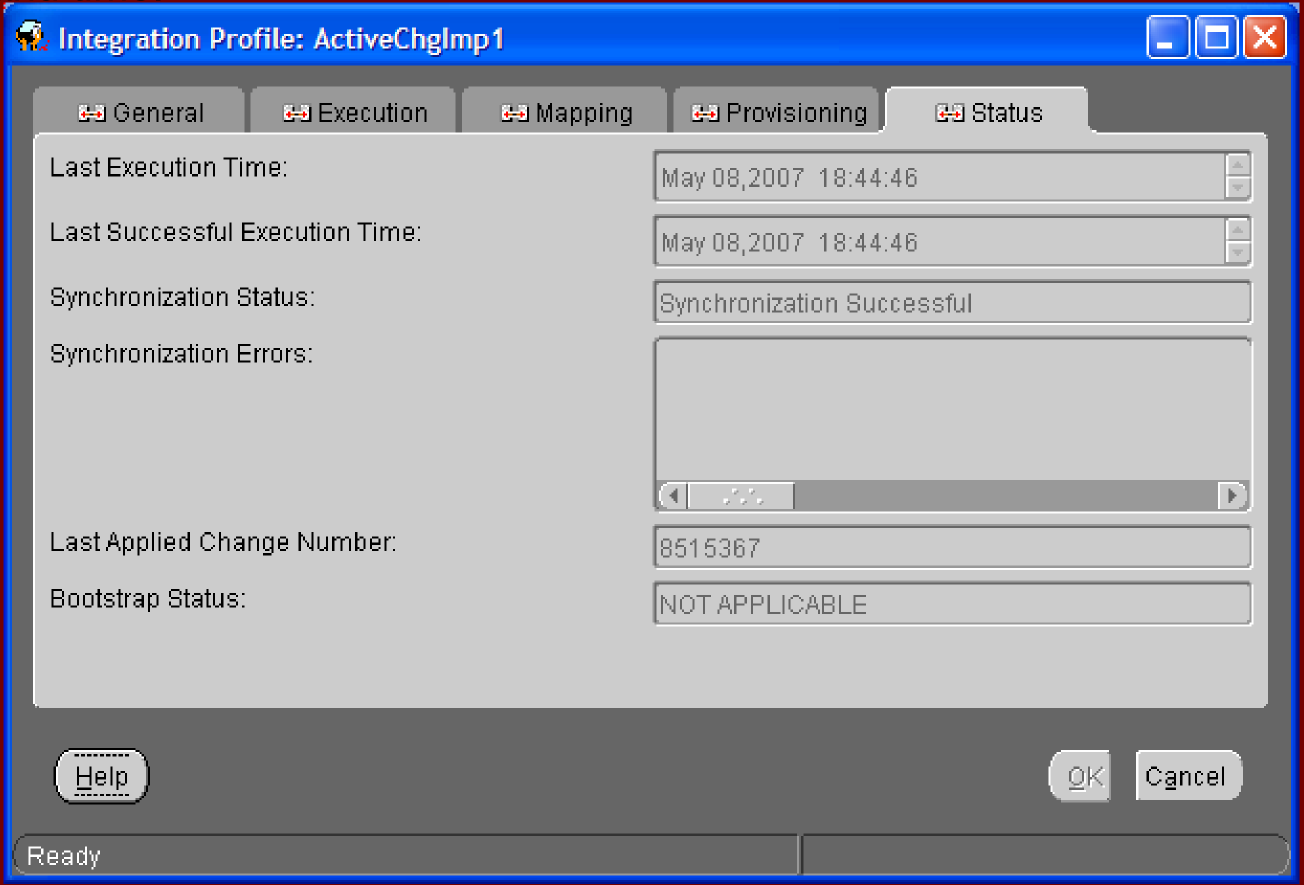
Connected Directory URL: eagle.ebsfl.com:389

Interface Type: LDAP

orclodipinterfacetype

Import Agent Configuration

- **Configuring the "Status" tab**
  - **ldapsearch -p 389 -h AD_Host_name -D "administrator@ebsfl.com" -w admin_password -b "" -s base "objectclass=*" highestCommittedUSN**
  - Enter the number returned in the "Last Applied Change Number" field.
  - You also want to set the "Last Successful Execution Time" to the current date and time.



Last Execution Time: May 08,2007 18:44:46

Last Successful Execution Time: May 08,2007 18:44:46

Synchronization Status: Synchronization Successful

Synchronization Errors:

Last Applied Change Number: 8515367

Bootstrap Status: NOT APPLICABLE

Ready

Import Agent Configuration

- **Configuring the "Status" tab**
  - Now we need to start the DIP server and enable the Agent profile
  - Use the following command to start the DIP server:

    ```
    oidctl connect=iasdb server=odisrv instance=2 config=1 start
    ```

  - **Substitute you OID database connect string where you see "connect"**
  - Now bring up your ActiveChgImp profile again. In the "General" tab, set the "Profile Status" to "Enable"
  - After enabling the ActiveChgImp profile, refresh the profile and open it again. This time click on the "Status" tab and check the synchronization status. It should read "Synchronization Successful"

Configuring External Authentication

- A script called "oidspadi.sh“ needs to be run
- It is located in $ORACLE_HOME/ldap/admin directory
- During execution of this command you will need to provide some basic information about OID and AD

Configuring External Authentication

- AD server FQDN or IP address.
- SSL or Non-SSL
- Port number that the AD server is running on
- Database connect string for the OID database
- “ODS" database schema user password. This is probably set to the same password you use for the cn=orcladmin users
- FQDN or IP address of the server that OID is running on
- Port number that OID server is running on
- Password for the orcladmin user
- Subscriber search base. This is the DN of the users container in OID that you want to authenticate to AD

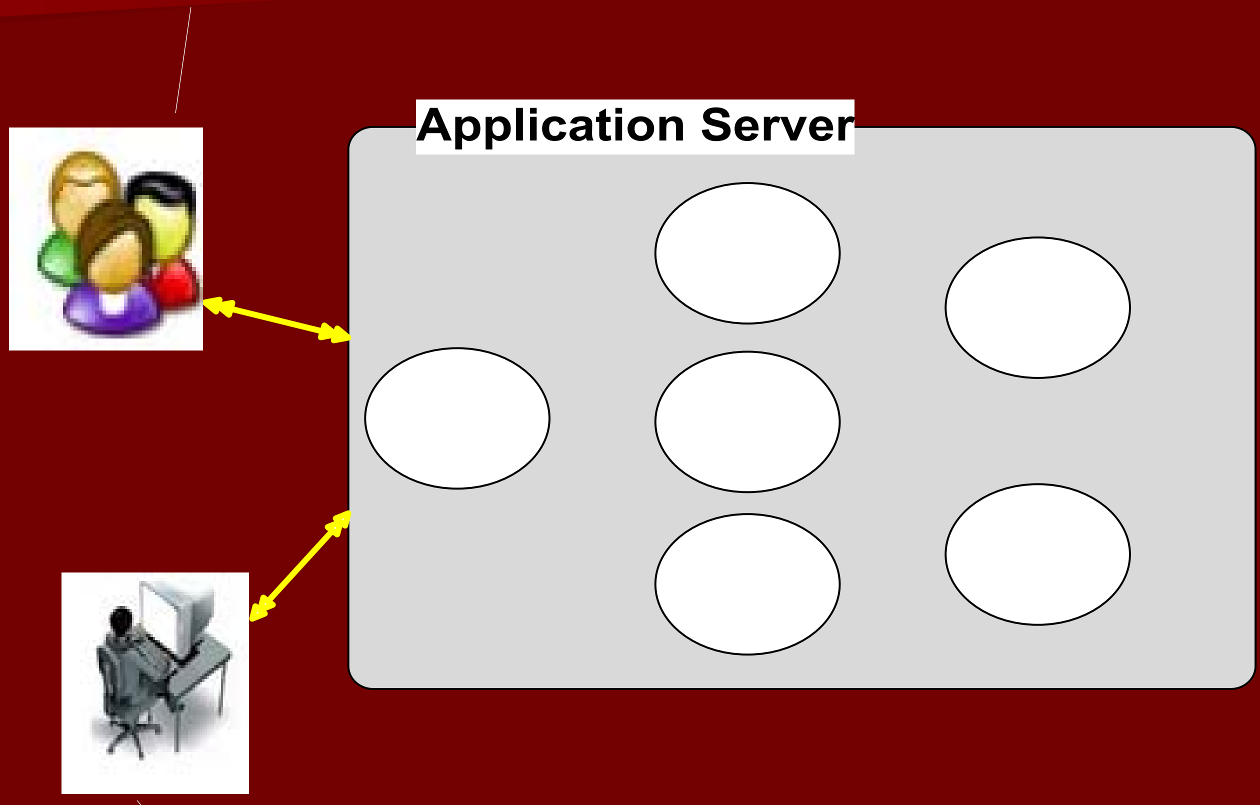
Configuring External Authentication

- "Exception Entry Property"
- This value acts as a filter and determines where users will authenticate. If you leave this value null, all users in your realm will authenticate using their credentials stored in AD. The value you enter here will determine which users will authenticate against OID and which users will authenticate against AD
- Here is an example value for the "Exception entry property":
- (&(objectclass=inetorgperson)(cn=orcladmin))
- This value tells OID that every user except the user "cn=orcladmin" will authenticate using credentials stored in AD

WNA

- WNA stands for Windows Native Authentication.
- It enables users inside a network to login to an SSO protected resource like Portal, without being challenged for a password.
- This is accomplished by passing the Domain credentials on to Portal

Our environment



Application Server

Internal
Users

Our problem

- Internal users do not want to login to access internal applications:
  - Governing Board Agenda Builder
  - EOC Tools
  - Weather Tools
  - Employee Directory
  - Etc.

Our Solution

- WNA (Windows Native Authentication) once configured, utilizes Windows Domain authentication using Kerberos tickets to pass the login credentials to the SSO Server
- SSO Server will validate the credentials against the KDC Server on Windows Domain Server and authenticate the user

WNA – How it works?

- When you log into your Windows desktop, a Kerberos session ticket is generated which contains among other things, your login credentials
- If Windows Native Authentication (WNA) has been configured on the Oracle SSO Server, you will be able to click on your Web application and not be challenged for credentials
- Kerberos session ticket which includes your Windows desktop credentials will be passed through the browser to the Oracle SSO server
- The SSO server will validate the credentials by checking them against the KDC server on the Windows domain server
- If authentication succeeds you will be granted access to your Web applications automatically

Configuring WNA

- Prerequisites
  - Installed Oracle Application Server 10*g* Instance
  - Verified that OID server is up and running.
  - OID must be configured for Active Directory Import
  - OID must be configured for External Authentication

Configuring WNA

- configure your krb5.conf file
- Located in /etc directory
-
  ```
   [libdefaults]
   default_realm = AD.EBSFL.COM
   [realms]
   AD.EBSFL.COM = {
   kdc = ad.ebsfl.com
  #kdc = eagle.ad.ebsfl.com:88
   default_domain = ad.ebsfl.com
   }
   [domain_realm]
   .ebsfl.com = AD.EBSFL.COM
  ```

Configuring WNA

- create a user account in the AD server with the same host name where your SSO server is running

- For example: snake

Configuring WNA

- Generate a keytab file that will be used by the SSO server to map the account name to the service principal name
- ktpass -princ HTTP/snake.ebsfl.com@AD.EBSFL.COM -pass welcome1 -mapuser snake -out snake.keytab
- The -princ value is HTTP/ followed by the FQDN of your SSO server, followed by @YOUR_AD_DEFAULT_REALM. This is case sensitive and you must have the AD default realm in upper case. The FQDN of the SSO server should be in lower case
- The -pass value must be set to the same password you assigned to the SSO hostname user account that you created in the AD server.
- The -mapuser value is the SSO hostname user you created in the AD server
- The -out value is the name you want to give for the file output that is generated, for example
- hostname.keytab
- copy the file to the $ORACLE_HOME/j2ee/OC4J_SECURITY/config directory on the SSO server

Configuring WNA

- **hostname.keytab**
- **copy the file to the $ORACLE_HOME/j2ee/OC4J_SECURITY/config directory on the SSO server**

Configuring WNA

- Test your Kerberos connection between your Linux server and the AD server

-
  ```
  # /usr/kerberos/bin/kinit -k -t $ORACLE_HOME/j2ee/OC4J_SECURITY/config/snake.keytab HTTP/snake.ebsfl.com
  ```

Configuring WNA

- Syntax for configuring WNA

```
$ORACLE_HOME/sso/bin/ssoca wna \
-mode sso -oh $ORACLE_HOME \
-ad_realm AD.EBSFL.COM \
-kdc_host_port eagle.ad.ebsfl.com:88 \
-keytab $ORACLE_HOME/j2ee/OC4J_SECURITY/config/snake.keytab \
-ssohost snake.ebsfl.com \
-oid ldap://snake.domain.com:389 \
-verbose
```

Configuring WNA – List of configuration Files

- **opmn.xml**
- **jaxn.xml**
- **jazn-data.xml**
- **web.xml**
- **orion-application.xml**
- **policy.properties**

Configuring WNA

- **Restart opmn**
- **# opmnctl stopall**
- **Wait about 1 minute before you start the application server**
- **# opmnctl startall**

Review

- Steps to set up AD- OID Sync.
- Steps to set up WNA
- Result

Lessons Learned

- Understand the concept
- Have the design in the paper
- Get the management buy-in
- Contact Oracle Support for the latest scripts before you start the process
- Work closely with the Sysadmins (Unix and AD)
- Implement and test
- Test and test

Lessons Learned

- Have a checklist of files and configuration to check in case of issues
- Set the password not to expire for the service account
- Backup before implementation
- Backup the WNA configuration files and write a script to restore. Very useful during the implementation

Conclusion

- It is a cool concept and works great once the nuances of the implementation are crossed
- Users in our organization love it and it works great

Questions and Answers