

## How to Configure Centrally Managed Users For Database Release 18c or Later Releases (Doc ID 2462012.1)

---

### In this Document

#### [Goal](#)

#### [Solution](#)

- [I Download the latest version of 'opwdintg.exe' for Password Authentication Integration](#)
- [II Configure the integration between Microsoft Active Directory and the Oracle Database](#)
- [III Configure Password Authentication for Centrally Managed Users](#)
- [IV Configure Authorization for Centrally Managed Users](#)
- [V Configure Kerberos Authentication](#)
- [VI Configure SSL authentication](#)
- [VII Known Issues and Troubleshooting Steps](#)
- [VIII How to investigate connection issues](#)
- [Additional Information:](#)

#### [References](#)

---

### APPLIES TO:

---

Oracle Database - Enterprise Edition - Version 18.1.0.0.0 and later  
Advanced Networking Option - Version 18.3.0.0.0 and later  
Information in this document applies to any platform.

### GOAL

---

Starting with 18c database users can be directly authenticated and authorized against Active Directory without using Oracle Enterprise User Security (EUS) or another intermediary directory service. Users can authenticate to the Oracle Database using credentials stored in Active Directory and also be associated with database schemas and roles using Active Directory groups. Microsoft Active Directory users can be mapped to exclusive or shared Oracle Database schemas and associated with database roles in the directory.

This note is intended to provide a quick overview on the steps needed to quickly setup authentication for CMU users and a few troubleshooting steps for the known issues.

You should have read and be familiar with the following documents to understand the concepts of Centrally Managed Users.

[Oracle Database Release 18 Security Guide, Chapter 5 "Configuring Centrally Managed Users with Microsoft Active Directory"](#).

[Oracle Database Release 19 Security Guide, Chapter 6 "Configuring Centrally Managed Users with Microsoft Active Directory"](#).

#### **Important !!!**

1. The minimum version requirement for Active Directory server is Windows 2008.

2. CMU is not available as a feature in Standard Edition, see [Licensing Information](#).

3. Apply the Mandatory Patches for CMU in 18C / 19C Database as explained in [Note 2716598.1](#)

[Patch 31404487](#) replaces [patch 28994890](#).

If patch 28994890 was applied on top of a 18c database DBRU (where DBRU version is lower than 18.11), then roll back patch 28994890, and only apply patch 31404487 to database 18c.

If the 18c database version is equal to or higher than DBRU 18.11, where bug 28994890 has been included in the base line DBRU, then apply patch 31404487 on top of the 18c DBRU directly.

Note that the patches are only applicable to on-premise databases. The content of the patches have been included in Autonomous Databases (ADBS) for CMU, if you [use Microsoft Active Directory with Autonomous Database](#).

## SOLUTION

The Oracle Database users can be authenticated against Active Directory using one of the following methods:

1. Password authentication
2. Kerberos authentication
3. SSL authentication

### I Download the latest version of 'opwdintg.exe' for Password Authentication Integration

You should use the latest version of 'opwdintg.exe' as it solves several known issues on Active Directory side.

To download the latest version of opwdintg.exe, please follow the directions below to agree to the Oracle License Agreement.

You must accept the 'Oracle License Agreement' to download this software 'opwdintg.exe'

By clicking the I AGREE link below, you confirm the following:

I reviewed and accept the [Oracle License Agreement](#)

[I AGREE](#)

After download, you should verify the integrity of the software by checking the checksum(hash) of the file matches the following:

```
C:\Users\Administrator\Downloads>certutil -hashfile opwdintg.exe md5
```

MD5 hash of file opwdintg.exe:

64 bc 38 92 69 73 61 be b8 df 6c f8 d9 d3 2a a6

CertUtil: -hashfile command completed successfully.

```
C:\Users\Administrator\Downloads>certutil -hashfile opwdintg.exe
```

SHA1 hash of file opwdintg.exe:

a3 5e cf 9d 3b db ad df d5 f8 03 40 b9 c1 d3 34 f0 4a 07 ad  
CertUtil: -hashfile command completed successfully.

## II Configure the integration between Microsoft Active Directory and the Oracle Database

### Steps to be performed on the Active Directory Server (for password authentication, download the latest version of opwdintg.exe)

For Password Authentication, follow the instructions of "Install the Password Filter and Extend the Microsoft Active Directory Schema" in the "Configuring Centrally Managed Users with Microsoft Active Directory" chapter, in the Oracle online public document "Oracle Database Security Guide".

To summarize, use 'opwdintg.exe' to extend the AD schema and install password filter; restart the Windows DC; Then add AD users to appropriate ORA\_VFR groups, and reset passwords for AD users.

**AD administrator should make sure that AD schema has been extended so that the 'orclCommonAttribute' is present as one of the AD users' attributes, and that Oracle password hash has been populated in this attribute for any AD users who need to connect to Oracle databases, and also make sure this attribute is visible to Oracle service user (described below).**

Next, perform the following steps:

1. Login to the Active Directory with a privileged account and create the Oracle service user:

New Object - User ×

Create in: myad.example.com/Users

First name:  Initials:

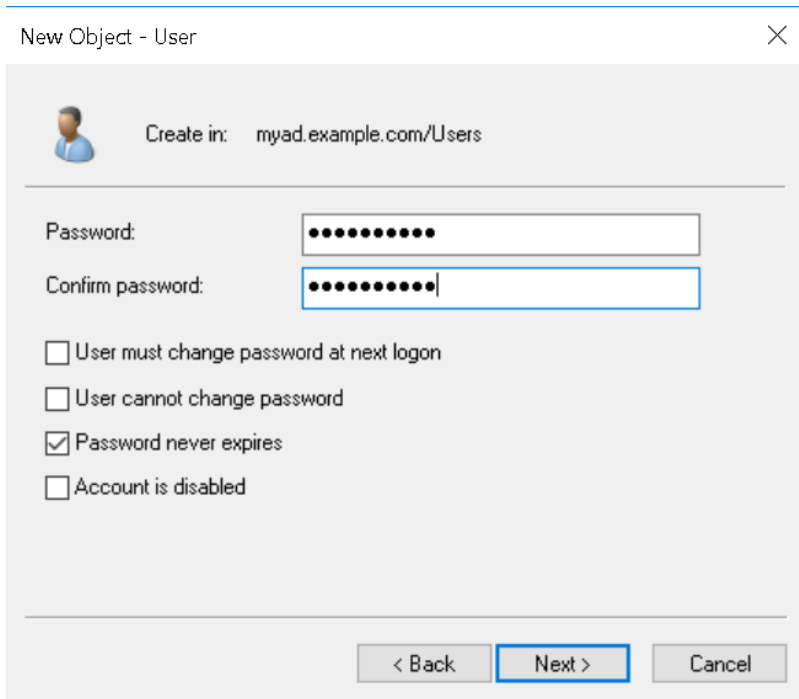
Last name:

Full name:

User logon name:

User logon name (pre-Windows 2000):

< Back **Next >** Cancel



New Object - User

Create in: myad.example.com/Users

Password: .....

Confirm password: .....

☐ User must change password at next logon

☐ User cannot change password

☒ Password never expires

☐ Account is disabled

< Back   Next >   Cancel

## 2. Grant the Oracle service directory user account the Read properties and Write lockoutTime

Grant the Oracle service directory user account the permissions "Read Properties" and "Write lockoutTime" of the AD users who need to login to Oracle database; You also need to grant the Oracle service user "Control Access" permission on 'orclCommonAttribute' of the AD users who need to login to Oracle database.

To simplify the task on Windows domain controller, you can use the following commands to delegate the control (i.e., granting the required permissions) on the container (MyOU1, in the example below) of target AD users to the Oracle service user (EXAMPLE\oracleservice) in the Windows domain "EXAMPLE":

```
dsacl "ou=MyOU1,dc=example,dc=com" /I:S /G "EXAMPLE\oracleservice:RPWP;lockoutTime"
dsacl "ou=MyOU1,dc=example,dc=com" /I:S /G "EXAMPLE\oracleservice:CA;orclCommonAttribute"
```

Delegation of Control Wizard

Select Users, Computers, or Groups

Select this object type:  
Users, Groups, or Built-in security principals

From this location:  
myad.example.com

Enter the object names to select (examples):  
oracleservice@oracleservice@myad.example.com

Advanced... OK Cancel

< Back Next > Cancel Help

Delegation of Control Wizard

Tasks to Delegate

You can select common tasks or customize your own.

☐ Delegate the following common tasks:

- ☐ Create, delete, and manage user accounts
- ☐ Reset user passwords and force password change at next logon
- ☐ Read all user information
- ☐ Modify the membership of a group
- ☐ Join a computer to the domain
- ☐ Manage Group Policy links
- ☐ Generate Resultant Set of Policy (Planning)

☒ Create a custom task to delegate

< Back Next > Cancel Help

Delegation of Control Wizard

Permissions

Select the permissions you want to delegate.

Show these permissions:

- ☒ General
- ☒ Property-specific
- ☐ Creation/deletion of specific child objects

Permissions:

- ☐ Full Control
- ☒ Read
- ☐ Write
- ☐ Create All Child Objects
- ☐ Delete All Child Objects
- ☐ Read All Properties

< Back Next > Cancel Help

Delegation of Control Wizard

Permissions

Select the permissions you want to delegate.

Show these permissions:

- ☒ General
- ☒ Property-specific
- ☐ Creation/deletion of specific child objects

Permissions:

- ☒ Write lockoutTime
- ☐ Read loginShell
- ☐ Write loginShell
- ☐ Read Logon Name
- ☐ Write Logon Name
- ☐ Read Logon Name (pre-Windows 2000)

< Back Next > Cancel Help

3. Export the trusted certificate from Active Directory server:

```

C:\Windows\System32>certutil -ca.cert root.crt

CA cert[0]: 3 -- Valid

CA cert[0]:

-----BEGIN CERTIFICATE-----

MIIDlTCCAn2gAwIBAgIQPvTQKsXf26VDm6xnziB1VzANBgkqhkiG9w0BAQsFADBd
MRMwEQYKCZImiZPyLGBGRYDY29tMRcwFQYKCZImiZPyLGBGRYHZXhhbXBsZTEU
MBIGCgmSJomT8ixkARkWBGl5YWQxZmFzAVBgNVBAMTDm15YWQtRlJPU1RZLUNBMB4X

.....

unRZRxoiczbv9fBliM13/JTm31dz8GBElDkxGEOcsH8pt1aj/V+IeGQJGSRaPIPO
xgmzWLzJjmAvT6SBXPutG0pygIuaqx9/3vk+zQDF0xxBLnd37B9YHcxfFIVbQIcV
5BzhOAtcSP9m

-----END CERTIFICATE-----

CertUtil: -ca.cert command completed successfully.

C:\Windows\System32>

```

## Steps to be performed on the Database Server

1. Create the wallet to store the oracle service user credentials and the trusted certificate from the Active Directory Server.

You can specify the CMU wallet location by using CMU\_WALLET database property.

For example, you can specify a CMU wallet location, e.g., '/u01/app/cmu\_wallet' to be a common wallet location for every database or PDB/CDB root, by executing the following steps:

Step #1. Apply patch 31404487 to your database 19c or 18c.

Step #2. Prepare the CMU wallet from command line:

```
mkdir /u01/app/cmu_wallet
```

Config or move the dsi.ora and wallet files from its current location to this common wallet location:

```
mv dsi.ora /u01/app/cmu_wallet/
mv cwallet.sso /u01/app/cmu_wallet/
mv ewallet.p12 /u01/app/cmu_wallet/
```

Step #3. Execute the following DDL statements on each database or PDB/CDB root:

```
CREATE OR REPLACE DIRECTORY example_dir AS '/u01/app/cmu_wallet';
ALTER DATABASE PROPERTY SET CMU_WALLET='EXAMPLE_DIR';
```

Then this common wallet location '/u01/app/cmu\_wallet' will be effective for each database or CDB/PDB of 19c/18c where the patch 31404487 has been applied.

If you do not want to use CMU\_WALLET database property, you can remove the CMU\_WALLET setting by executing:

```
ALTER DATABASE PROPERTY REMOVE CMU_WALLET;
```

If you do not use CMU\_WALLET database property, then the CMU wallet can be created in the default location or it can be placed in a custom location specified by the parameter WALLET\_LOCATION in sqlnet.ora, as illustrated below.

If using CDB/PDB, for each PDB, the wallet location is specific to the PDB's GUID.

If WALLET\_LOCATION is not set in sqlnet.ora, the PDB specific wallet location is \$ORACLE\_BASE/admin/<db\_unique\_name>/<pdb\_guid>/wallet/.

If WALLET\_LOCATION is specified in sqlnet.ora, the PDB specific wallet location is <WALLET\_LOCATION\_specified\_in\_sqlnet.ora>/<pdb\_guid>/. (See Oracle public online document "[Oracle Database Security Guide](#)" for more details).

You may need to create these file folders for these wallet locations if they have not been created yet. Then you can create wallet in one of these folders (wallet locations) and after you execute the command 'orapki wallet create -wallet' command, you will see the wallet embodied in the files 'ewallet.p12' and cwallet.sso' in the file folder.

You should place dsi.ora (described below) file in the same file folder (wallet location) where ewallet.p12 and cwallet.sso files are.

```
$ mkdir -p /u01/app/cmu_wallet
$ cd /u01/app/cmu_wallet
$ orapki wallet create -wallet . -pwd <wallet password> -auto_login
Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Operation is successfully completed.
$ mkstore -wrl . -createEntry ORACLE.SECURITY.USERNAME oracleservice
Oracle Secret Store Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Enter wallet password:
$ mkstore -wrl . -createEntry ORACLE.SECURITY.DN
cn=oracleservice,cn=users,dc=myad,dc=example,dc=com
Oracle Secret Store Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Enter wallet password:
$ mkstore -wrl . -createEntry ORACLE.SECURITY.PASSWORD <password>
Oracle Secret Store Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Enter wallet password:
$ orapki wallet add -wallet . -trusted_cert -cert root.crt
Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
```

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Cannot modify auto-login (sso) wallet

Enter wallet password:

Operation is successfully completed.

\$orapki wallet display -wallet .

Oracle PKI Tool Release 18.0.0.0.0 - Production

Version 18.1.0.0.0

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Requested Certificates:

User Certificates:

Oracle Secret Store entries:

ORACLE.SECURITY.USERNAME

ORACLE.SECURITY.DN

ORACLE.SECURITY.PASSWORD

Trusted Certificates:

Subject: CN=myad-FROSTY-CA,DC=myad,DC=example,DC=com

If the wallet location is specified by CMU\_WALLET database property, then CMU will always use that wallet location. If the database property CMU\_WALLET is not set, and the sqlnet.ora WALLET\_LOCATION parameter is also not specified in the sqlnet.ora file, then the database will search the following default locations in this order for the wallet. The directory location may need to be created.

For a non-multitenant database, or for the CDB root container of a multitenant database:

- a. \$ORACLE\_BASE/admin/<db\_unique\_name>/wallet/
- b. \$ORACLE\_HOME/admin/<db\_unique\_name>/wallet/

For a PDB in a multitenant database:

- a. \$ORACLE\_BASE/admin/<db\_unique\_name>/<pdb\_guid>/wallet/
- b. \$ORACLE\_HOME/admin/<db\_unique\_name>/<pdb\_guid>/wallet/

2. Create the dsi.ora file (For more information on the location of dsi.ora, refer to the [documentation page](#)) with the Active Directory server information:

```
$ cat /u01/app/cmu_wallet/dsi.ora
DSI_DIRECTORY_SERVERS = (ADSERVER.EXAMPLE.COM:389:636)
DSI_DEFAULT_ADMIN_CONTEXT = "ou=MyOU1,dc=example,dc=com"
DSI_DIRECTORY_SERVER_TYPE = AD
```

Note:



### 1. On the location of dsi.ora:

The recommended location for dsi.ora is the database CMU wallet location. (For more information on the location of dsi.ora, refer to the [documentation](#) page)

### 2. On the content of dsi.ora:

The DSI\_DIRECTORY\_SERVERS should use fully qualified hostname, e.g.: ADSERVER.EXAMPLE.COM. or the host IP address, but not the standalone hostname. You should make sure that the network connectivity works between your database and the AD servers, e.g., by "ping <AD\_hostname>".

The DSI\_DIRECTORY\_SERVERS must follow the double colon format, e.g., to only use the TLS port 636, you should set it as follows:

```
DSI_DIRECTORY_SERVERS = (AD1.DOM1.EXAMPLE.COM::636, AD2.DOM2.EXAMPLE.COM::636)
```

DSI\_DEFAULT\_ADMIN\_CONTEXT is optional. This parameter is to limit the CMU search scope for AD users and groups.

DSI\_DEFAULT\_ADMIN\_CONTEXT should only be set if you want to limit the search scope so that only the AD users and/or groups under the subtrees of the directory specified by the DNs can be found by CMU.

In case that multiple Windows domains or multiple AD servers are to be used, set them in DSI\_DIRECTORY\_SERVERS, and separate each AD server information with a comma (,), e.g.:

```
DSI_DIRECTORY_SERVERS = (AD1.DOM1.EXAMPLE.COM:389:636, AD2.DOM2.EXAMPLE.COM:389:636)
```

In case that multiple DN of search bases are to be used, set them in DSI\_DEFAULT\_ADMIN\_CONTEXT, and separate each DN with semicolon(;), e.g.:

```
DSI_DEFAULT_ADMIN_CONTEXT = "ou=MyOU1,dc=example,dc=com; ou=MyOU2,dc=example,dc=com"
```

Note that setting DSI\_DEFAULT\_ADMIN\_CONTEXT will limit the search scope for AD users and groups.

### 3. Set the parameters ldap\_directory\_access to PASSWORD

```
SQL> alter system set ldap_directory_access='PASSWORD';  
  
System altered.
```

4. Set the ldap\_directory\_sysauth parameter to YES, so that administrative users from Active Directory can log in to Oracle Database with the SYSDBA, SYSOPER, SYSBACKUP, SYSDG, SYSKM, or SYSRAC administrative privilege.

```
SQL> alter system set ldap_directory_sysauth = 'YES' scope=spfile;  
  
System altered.
```

Note: LDAP\_DIRECTORY\_SYSAUTH is static parameter and change in this parameter value needs DB bounce

## III Configure Password Authentication for Centrally Managed Users

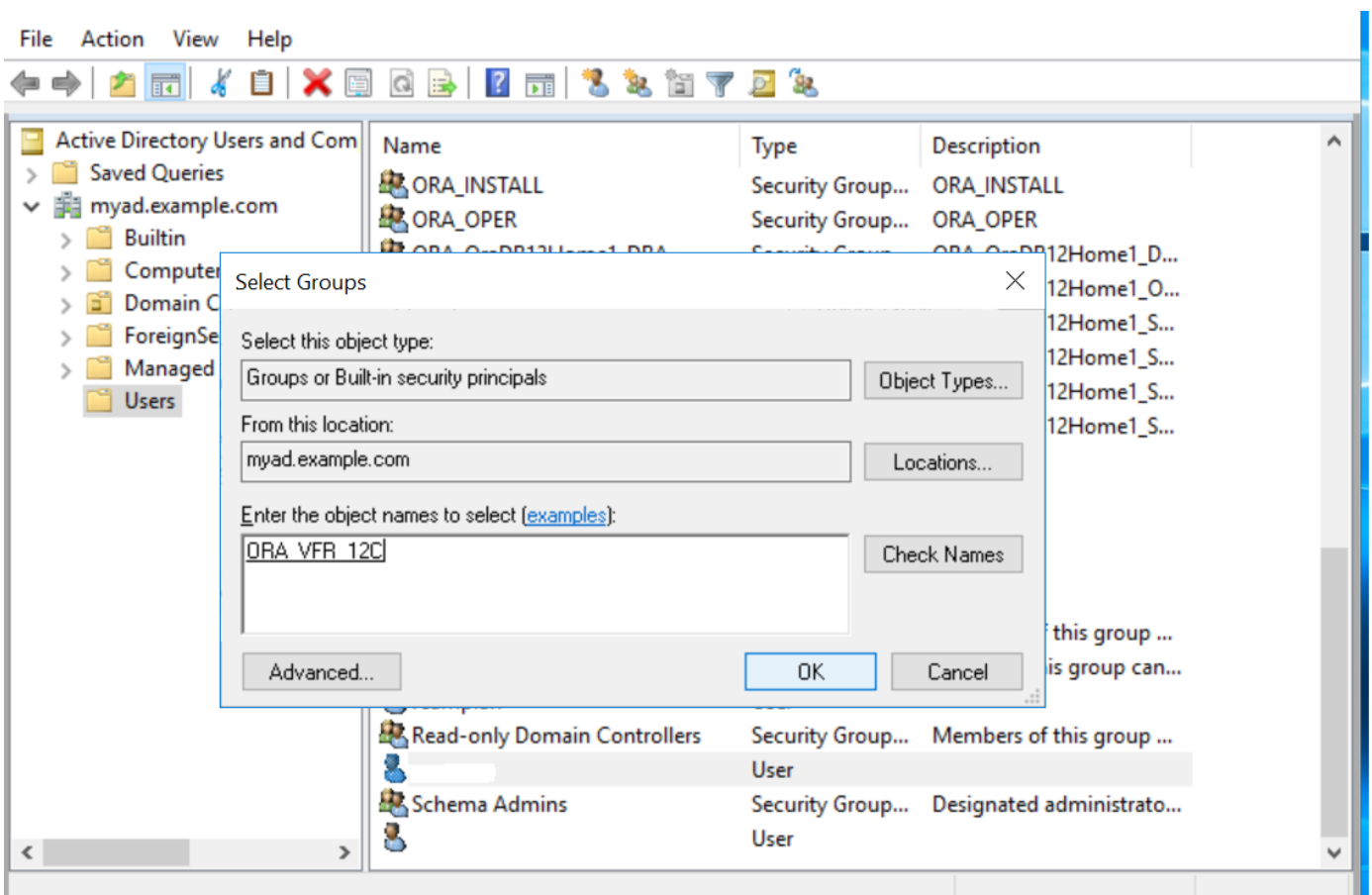
### Steps to be performed in the Active Directory:

1. Install the Password Filter and Extend the Microsoft Active Directory Schema using opwdintg.exe downloaded from this note. This step creates the following three verifier groups:

- ORA\_VFR\_MD5 is required when the Oracle Database WebDAV client is used.
- ORA\_VFR\_11G enables the use of the Oracle Database 11G password verifier.
- ORA\_VFR\_12C enables the use of the Oracle Database 12C password verifier.

```
Select C:\Windows\SYSTEM32\cmd.exe
Do you want to extend AD schema? [Yes/No]:Yes
Schema master is frosty.myad.example.com
=====
Extending AD schema with orclCommonAttribute for user object in AD domain:
DC=myad,DC=example,DC=com
=====
Schema extension for this domain will be permanent. Continue?[Yes/No]:Yes
Connecting to "frosty.myad.example.com"
Logging in as current user using SSPI
Importing directory from file "etadschm.ldf"
Loading entries.....
4 entries modified successfully.
The command has completed successfully
Done, Press Enter to continue...
```

2) Add the Active Directory users to any of the groups ORA\_VFR\_MD5, ORA\_VFR\_11G, and ORA\_VFR\_12C.



### 3) Update the database password file to version 12.2.

```
cd $ORACLE_HOME/dbs  
orapwd FILE='$ORACLE_HOME/dbs/orapworcl18' FORMAT=12.2
```

## Configuring Authentication for Centrally Managed Users

The users can be created in the database using exclusive schema or shared schema. In case of exclusive schema one Active Directory user is mapped to one global database user while in case of shared schema one directory group can be mapped to a shared global database user.

### 1. Create the globally identified user using an exclusive schema:

```
SQL> create user abc identified globally as 'cn=abc,cn=users,dc=myad,dc=example,dc=com';  
User created.  
SQL> grant create session to abc;  
Grant succeeded.  
SQL> connect "myad\abc"@orcl18  
Enter password:  
Connected.
```

```
select SYS_CONTEXT('USERENV','NETWORK_PROTOCOL') net_proto,sys_context('USERENV',  
'AUTHENTICATED_IDENTITY') auth_identity,SYS_CONTEXT('USERENV','SESSION_USER')  
db_user,sys_context('USERENV','ENTERPRISE_IDENTITY') ent_identity from dual;
```

NET_PROTO	AUTH_IDENTITY	DB_USER
ENT_IDENTITY		

TCP	myad\abc	ABC
cn=abc,cn=Users,dc=myad,dc=example,dc=com		

### 2. Create the globally identified user using a shared schema:

#### Note:

- The DN below 'cn=db-access,cn=Users,dc=myad,dc=example,dc=com' refers to an AD Group and is not an AD user.
- The AD user 'myad\abc' is a member of the AD group 'db-access'.

```
SQL> create user shared_user identified globally as 'cn=db-access,cn=Users,dc=myad,dc=example,dc=com';
```

User created.

```
SQL> grant create session to shared_user;
```

Grant succeeded.

```
SQL> connect abc@orcl18
```

```
Enter password:
```

```
Connected.
```

```
SQL>
```

```
select SYS_CONTEXT('USERENV','NETWORK_PROTOCOL') net_proto,sys_context('USERENV',  
'AUTHENTICATED_IDENTITY') auth_identity,SYS_CONTEXT('USERENV','SESSION_USER')  
db_user,sys_context('USERENV','ENTERPRISE_IDENTITY') ent_identity from dual;
```

NET_PROTO	AUTH_IDENTITY	DB_USER	ENT_IDENTITY
-----------	---------------	---------	--------------

-----	-----	-----	-----
-------	-------	-------	-------

TCP	MYAD\ABC	SHARED_USER	
cn=abc,cn=Users,dc=myad,dc=example,dc=com			

## IV Configure Authorization for Centrally Managed Users

Besides Authentication with CMU it is possible to configure authorization. There are flexibility and manageability advantages when using database global roles instead of granting privileges to a database global user/schema directly.

If an AD user is added to or removed from an AD group, the authorization will change automatically on database side.

For example:

Suppose that the AD user 'myad\abc' is a member of the AD group 'db-access', then the AD user will have 'create session' privilege when accessing oracle database, and can further obtain different authorization after being added to or removed from other AD groups such as 'support' group or 'developers' group, as illustrated below.

```
SQL> create user shared_user identified globally as 'cn=db-  
access,cn=Users,dc=myad,dc=example,dc=com';  
SQL> create role db_access_role identified globally as 'cn=db-  
access,cn=Users,dc=myad,dc=example,dc=com';  
SQL> grant create session to db_access_role;  
  
SQL> create role support_role identified globally as  
'cn=support,cn=Users,dc=myad,dc=example,dc=com';  
SQL> create role developer_role identified globally as  
'cn=developers,cn=Users,dc=myad,dc=example,dc=com';  
  
SQL> grant <appropriate_privileges> to support_role;  
  
SQL> grant <appropriate_privileges> to developer_role;  
  
SQL> connect abc@orcl18
```

Depending on which groups the AD user is a member of, the AD user will have appropriate privileges when logging in Oracle database through the shared\_user.

## V Configure Kerberos Authentication

To configure Kerberos settings on the database server and on the client please use as a reference one of the following notes:

[Note 1996329.1](#) How To Configure Kerberos Authentication In A 12c Database

[Note 1304004.1](#) Configuring ASO Kerberos Authentication with a Microsoft Windows 2008 R2 Active Directory KDC

The user will be created this way:

```
SQL> create user shared_user identified globally as 'cn=db-
access,cn=Users,dc=myad,dc=example,dc=com';
User created

SQL> grant create session to shared_user;

Grant succeeded.

$ sqlplus /@orcl18

SQL*Plus: Release 18.0.0.0.0 Production on Tue Oct 16 14:00:44 2018
Version 18.1.0.0.0

Copyright (c) 1982, 2017, Oracle. All rights reserved.

Last Successful login time: Tue Oct 16 2018 13:56:16 +03:00

Connected to:
Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production
Version 18.3.0.0.0

SQL> show user
USER is "SHARED_USER"
SQL>

SQL> select SYS_CONTEXT('USERENV','NETWORK_PROTOCOL') net_proto,sys_context('USERENV',
'AUTHENTICATED_IDENTITY') auth_identity,SYS_CONTEXT('USERENV','SESSION_USER')
db_user,sys_context('USERENV','ENTERPRISE_IDENTITY') ent_identity from dual;

      NET_PROTO              AUTH_IDENTITY              DB_USER
-----
tcp                      def@MYAD.EXAMPLE.COM              SHARED_USER
cn=def,cn=Users,dc=myad,dc=example,dc=com
```

## VI Configure SSL authentication

The general steps to configure SSL authentication and encryption in the database are detailed in the following MOS note:

[Note 1381035.1](#) Configuring SSL Authentication With Client Certificates Signed By The Server Using orapki

In case of Centrally Managed Users the database must be integrated with Active Directory so the database client must use SSL user certificates that have been issued for the Active Directory users (with correct DN of AD users).

For the SSL setup we will use the database server wallet created at the step **I** and we will create the self\_signed certificate for Oracle Certification Authority:

```
$ orapki wallet add -wallet . -dn "cn=orcl18" -keysize 2048 -self_signed -validity 365 -pwd
<wallet password>

Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Operation is successfully completed.

$ orapki wallet display -wallet /u02/app/oracle/admin/orcl18/wallet

Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Requested Certificates:
User Certificates:
Subject: CN=orcl18

Oracle Secret Store entries:
ORACLE.SECURITY.DN
ORACLE.SECURITY.PASSWORD
ORACLE.SECURITY.USERNAME

Trusted Certificates:
Subject: CN=myad-FROSTY-CA,DC=myad,DC=example,DC=com
Subject: CN=orcl18
```

The next steps are to create and export the request for the client certificate and signed it using the Oracle CA.

On the client we create and export the request for the client certificate:

```
$ orapki wallet add -wallet . -dn "cn=def,cn=users,dc=myad,dc=example,dc=com" -keysize 2048 -pwd
<wallet password>

Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Operation is successfully completed.

$ orapki wallet export -wallet . -dn "cn=def,cn=users,dc=myad,dc=example,dc=com" -request
cert_user.txt -pwd <wallet password>

Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0
Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.
Operation is successfully completed.
```

We copy the file cert\_user.txt in the server wallet to be signed using Oracle CA:

```
$ orapki cert create -wallet . -request cert_user.txt -cert def.crt -validity 365 -pwd <wallet password>

Oracle PKI Tool Release 18.0.0.0.0 - Production

Version 18.1.0.0.0

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Operation is successfully completed.
```

Still in the server wallet we export the trusted certificate and copy it along with the signed user certificate:

```
$ orapki wallet export -wallet . -dn "CN=orcl18" -cert trust_orcl18.crt

Oracle PKI Tool Release 18.0.0.0.0 - Production

Version 18.1.0.0.0

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Operation is successfully completed.

$ cp trust_orcl18.crt def.crt /home/oracle/TNS/client_wallet
```

In the client wallet we will import the signed user certificate and the trusted certificate:

```
$ orapki wallet add -wallet . -user_cert -cert def.crt -pwd <wallet_password>

Oracle PKI Tool Release 18.0.0.0.0 - Production

Version 18.1.0.0.0

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Operation is successfully completed.

$ orapki wallet add -wallet . -trusted_cert -cert trust_orcl18.crt -pwd <password>

Oracle PKI Tool Release 18.0.0.0.0 - Production

Version 18.1.0.0.0

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Operation is successfully completed.
```

Client wallet will have the following content:

```

$ orapki wallet display -wallet .

Oracle PKI Tool Release 18.0.0.0.0 - Production
Version 18.1.0.0.0

Copyright (c) 2004, 2017, Oracle and/or its affiliates. All rights reserved.

Requested Certificates:
User Certificates:
Subject: CN=def,CN=users,DC=myad,DC=example,DC=com

Trusted Certificates:
Subject: CN=orcl18

```

To test the connection we will use the `shared_user` identified globally created at the step **IV**

```

$ sqlplus /@orcl18_ssl

SQL*Plus: Release 18.0.0.0.0 Production on Sat Nov 17 15:33:01 2018

Version 18.1.0.0.0

Copyright (c) 1982, 2017, Oracle. All rights reserved.

Last Successful login time: Sat Nov 17 2018 15:28:56 +03:00

Connected to:

Oracle Database 18c Enterprise Edition Release 18.0.0.0.0 - Production
Version 18.3.0.0.0

SQL> set linesize 800

column NET_PROTO format a10
column AUTH_IDENTITY format a60
column DB_USER format a80
column ENT_IDENTITY format a80
column DB_USER format a20

SQL> select SYS_CONTEXT('USERENV','NETWORK_PROTOCOL') net_proto,sys_context('USERENV',
'AUTHENTICATED_IDENTITY') auth_identity,SYS_CONTEXT('USERENV','SESSION_USER')
db_user,sys_context('USERENV','ENTERPRISE_IDENTITY') ent_identity from dual;

SQL>

```

NET_PROTO	ENT_IDENTITY	AUTH_IDENTITY	DB_USER
tcps	CN=def,CN=users,DC=myad,DC=example,DC=com	cn=def,cn=users,dc=myad,dc=example,dc=com	SHARED_USER

## VII Known Issues and Troubleshooting Steps

### 1. Connection using password authentication fails with ORA-28276:



```
SQL> connect "myad\def"@orcl18
```

Enter password:

ERROR:

ORA-28276: Invalid ORACLE password attribute.

**Cause:** The orclCommonAttribute attribute has not been correctly populated with user password.

Example:

```
[oracle@seclin lib]$ ldapsearch -h <AD_Server> -p 389 -D
"cn=oracleservice,cn=users,dc=myad,dc=example,dc=com" -w **** -U 2 -W "file:<wallet_path>" -P
<password> -b "dc=myad,dc=example,dc=com" -s sub "(sAMAccountName=def*)" dn orclCommonAttribute

CN=def,CN=Users,DC=myad,DC=example,DC=com

orclCommonAttribute=
```

**Solution:** Run 'opwdintg.exe' to install password filter on *every* Windows domain controller in the domain for AD, restart *each* Windows domain controller machine, then assign AD user to appropriate ORA\_VFR group, reset user password on Active Directory and run ldapsearch to check that password has been generated. (Note: *each* Windows domain controller must be restarted after installing the password filter, otherwise the password filter will not start to work on that Windows domain controller.)

2. The connection with a kerberos user fails with ORA-28030 while the connection with password authentication fails with ORA-28043. Ldapbind on the port 636 completes successfully.

```
$ sqlplus /@orcl18

SQL*Plus: Release 18.0.0.0.0 Production on Mon Oct 1 16:53:01 2018

Version 18.1.0.0.0

Copyright (c) 1982, 2017, Oracle. All rights reserved.

ERROR:

ORA-28030: Server encountered problems accessing LDAP directory service


Enter user-name:


$ sqlplus "myad\def"

SQL*Plus: Release 18.0.0.0.0 Production on Mon Nov 5 15:52:34 2018

Version 18.1.0.0.0

Copyright (c) 1982, 2017, Oracle. All rights reserved.

Enter password:

ERROR:

ORA-28043: invalid bind credentials for DB-OID connection
```

```
$ ldapbind -h <AD_Server> -p 636 -D "cn=oracleservice,cn=users,dc=myad,dc=example,dc=com" -w  
<service_user_password> -U 2 -W "file:<wallet_path>" -P <wallet_password>  
  
bind successful
```

**Cause:** [Bug 31404487](#)

**Solution:** Apply <Patch 31404487>

### 3. Connection fails with ORA-28293:

```
$ sqlplus /@orcl18  
  
SQL*Plus: Release 18.0.0.0.0 Production on Mon Oct 1 16:53:01 2018  
  
Version 18.1.0.0.0  
  
Copyright (c) 1982, 2017, Oracle. All rights reserved.  
  
ERROR:  
  
ORA-28293: No matched Kerberos Principal found in any user entry
```

**Cause:** [Bug 31404487](#)

**Solution:** Apply [Patch 31404487](#)

### 4. Connection fails with ORA-28300: No permission to read user entry in LDAP directory service.

ORA-28300 error is observed in the CMU trace as follows:

```
2019-06-27 19:51:55.0 - KZLG_ERR: failed to modify user status Insufficient access  
2019-06-27 17:57:27.0 - KZLG_ERR: LDAPERR=50, OER=28300
```

**Cause:** The Oracle service directory user does not have required permissions to access the AD user who tries to login to Oracle database.

**Solution:** Follow the instructions in this document to grant the service user the required permissions to access the properties of the AD user who tries to login to database.

```
- Steps to be performed on the Active Directory Server  
  2. Grant the Oracle service directory user account the "Read Properties" and "Write lockoutTime"  
  (permissions to access the properties of the AD user who tries to login to the database), and also  
  the permission "Control Access" on the 'orclCommonAttribute' of the AD users.
```

### 5. Connection fails with ORA-28274: No ORACLE password attribute corresponding to user nickname exists.

```
28274, 0000, "No ORACLE password attribute corresponding to user nickname exists."  
// *Cause: LDAP user entry corresponding to user nickname does not have a  
//          ORACLE password attribute or the attribute is not initialized.  
// *Action: Make sure user entries in LDAP are correctly provisioned with  
//          correct ORACLE password attribute values.
```

Cause specific to CMU-AD:

1. The AD schema has Not been extended and/or populated properly.
2. The Oracle service directory user does not have required permissions to access the orclCommonAttribute of the user who tries to login to Oracle database.

Solution specific to CMU-AD:

1. Use 'opwdintg.exe' to extend AD schema (which only needs to be done once), and install password filter (which needs to be done on *every* Windows domain controller); restart the Windows DC; Then add AD users to appropriate ORA\_VFR groups; and reset passwords for the AD users.
2. Follow the instructions in this document to grant the service user the required permissions to access the properties of the AD user who tries to login to database.
  - See "Steps to be performed on the Active Directory Server" in this note:
    - Grant the Oracle service directory user account the Read properties and Write lockoutTime (permissions to access the properties of the AD user who tries to login to the database), and also the permission "Control Access" on the 'orclCommonAttribute' of the AD users.

6. CMU-AD user logon fails with ORA-1017 using @ and \_ characters in AD user password. CMU-AD user authentication works when AD user password has # or there are no special characters.

CMU-AD user password is not Oracle user password. The "Oracle Password Complexity - List Of Allowed Special Characters. (Doc ID 2220631.1)" does not apply to CMU-AD users.

Double quotes should be placed around the AD user's password with special characters such as @ and \_.

For example:

Assuming that a CMU-AD user has an AD user name "adu test", and the AD user's password is "Welcome@\_", then the AD user should be able to logon to Oracle database 'orcl18' with double quotes placed around both the username and the password, as follows.

If the AD user is from the same domain as the Oracle service user:

```
SQL> conn "adu test"/"Welcome@_"@orcl18
Connected.
```

Or if the AD user is from a different domain than the Oracle service user, the Windows domain name "EXAMPLE" must be included in the logon username:

```
SQL> conn "EXAMPLE\adu test"/"Welcome@_"@orcl18
Connected.
```

```
SQL> conn "EXAMPLE\adu test"@orcl18
Enter password: *****
Connected.
```

Note that there are 11 characters (9 characters for Welcome@\_ , plus 2 characters for 2 double quotes) entered following the "Enter password" prompt.

## VIII How to investigate connection issues

- 1) KZLG tracing using [Note 2470608.1](#) Tracing CMU connection issues

### Additional Information:

Users Supported by Centrally Managed Users with Microsoft Active Directory

[https://docs.oracle.com/en/database/oracle/oracle-](https://docs.oracle.com/en/database/oracle/oracle-database/18/dbseg/integrating_mads_with_oracle_database.html#GUID-845BFA4E-6BB4-4E01-854B-DF9E5A37221B)

[database/18/dbseg/integrating\\_mads\\_with\\_oracle\\_database.html#GUID-845BFA4E-6BB4-4E01-854B-DF9E5A37221B](https://docs.oracle.com/en/database/oracle/oracle-database/18/dbseg/integrating_mads_with_oracle_database.html#GUID-845BFA4E-6BB4-4E01-854B-DF9E5A37221B)

Active Directory users could accidentally (or on purpose) be a member of multiple groups in Active Directory that are mapped to different shared schemas on the same database. The user could also have an exclusive mapping to a database schema. In cases where the user has multiple possible schema mappings when they login, the following precedence rules apply:

If an exclusive mapping exists for a user, then that mapping takes precedence over any other shared mappings.

If multiple shared schema mappings exist for a user, then the shared user mapping with lowest schema ID (USER\_ID) takes precedence.

Oracle recommends only having one possible mapping per user so unexpected schema mappings do not occur.

If the "Write lockoutTime" permission cannot be granted to Oracle service user due to corporate security policy, the following parameter can be used to implement CMU without granting Oracle service directory user account the permission of "Write lockoutTime" on those AD users who need to login Oracle database:

```
alter system set "_ldap_reset_user_account_flg"=FALSE;
```

Note that the side effects with setting above parameter to FALSE are:

- 1) The AD user's Failed Login Count (FLC) in Active Directory will not be automatically reset by AD user's subsequent successful login to Oracle database.

- 2) If an AD user account is locked out, then after the AD user has waited longer than the account lockout duration, the FLC will not be reset by the AD user's subsequent successful login to Oracle database.

Both situations can result in that an AD user account may get locked out unexpectedly as its FLC continue to accumulate without being reset properly.

There will be a warning message written to the alert.log when the attempt to unlock a locked-out AD user account fails:

[WARNING] Failed to modify account status for AD user <AD\_username>

1. The following underscore parameter can be set to FALSE to enable AD users login Oracle database through nested AD groups. Note that setting this parameter to FALSE may cause performance issue on AD server and slow down the AD users' login to database.

```
alter system set "_ldap_use_all_direct_groups_only"=FALSE;
```

2. If the imported AD CA root certificate in the database wallet expires, the expired certificate should be removed from the database wallet, and the renewed AD CA root certificate needs to be exported from Windows domain controller which is the CA root, and then be imported to the database wallet.

## REFERENCES

---

Didn't find what you are looking for?