Oracle External Password Store (Wallet)

Oracle External Password Store (Wallet) on Unix servers

Contents

[Document history 2](#_Toc68786776)

[Introduction 2](#_Toc68786777)

[UNIX Requirements 2](#_Toc68786778)

[Wallet creation requirements/process 3](#_Toc68786779)

[Installation 3](#_Toc68786780)

[Update sqlnet.ora with the location of the password store 3](#_Toc68786781)

[Create TNS alias in tnsnames.ora file used by Password Store 3](#_Toc68786782)

[Create “Wallet” - Password Store 3](#_Toc68786783)

[Create credential in wallet Password Store 4](#_Toc68786784)

[How to handle issues with PMSP (Cyberark) 5](#_Toc68786785)

[How to setup a segregated wallet design 5](#_Toc68786786)

[Test the connection 6](#_Toc68786787)

[Using the password store - Sqlplus 6](#_Toc68786788)

[Oracle database utilities 6](#_Toc68786789)

[JDBC thin 6](#_Toc68786790)

[Other examples on JDBC thin 7](#_Toc68786791)

[Managing the Password Store 7](#_Toc68786792)

[Change the password 7](#_Toc68786793)

[List entries 7](#_Toc68786794)

[Delete credential 8](#_Toc68786795)

[Where do we save a record of the password to the External Password Store 8](#_Toc68786796)

[Wallet’s password - store on the server. 8](#_Toc68786797)

[Wallet’s password - store on Nodbotp. 9](#_Toc68786798)

[Create new Entry 10](#_Toc68786799)

[Update Existing Entry 11](#_Toc68786800)

## Document history

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Name** | **CommentComment** |
| 14-09-2017 | 1.0 | Peter Mynster Lyndgaard | Initial version |
| 27-08-2018 | 1.1 | Peter Mynster Lyndgaard | SQL to Save wallet information  in Nodbotp instead of Gridp |
| 23-09-2020 | 1.2 | Peter Mynster Lyndgaard | Documentation review, no updates |
| 08-04-2021 | 1.3 | Peter Mynster Lyndgaard | Added description for a segregated design |

## Introduction

When accessing Oracle databases from application servers, it is a necessity to provide credential to login to the database. As we do not want to write passwords in clear text in command procedures, we will use Oracle External Password Store.

Oracle both refer to is as External password store and as a wallet.

This implementation is adapted for SoD (Segration of duties), each Unix user have a wallet, that is secured from other unix users.

Prior to the wallet creation, the O/S user for whom we are creating the wallet must be able to run commands from the $ORACLE\_HOME, for this reason the environment variables must be correctly set for the OS user on the application server

## UNIX Requirements

A new local group “orawallet” must be created, the “ORACLE” user and the application OS must both be part of this group. (Must be ordered in a CRQ to UNIX group).

To check this run the following as the oracle user:

cat /etc/group |grep -i wallet

eg:  
orawallet:x:38000684:calypt,oracle

As the wallet will be created and owned by the ORACLE user, the above will allow the O/S user access to read the wallet.

Note: The parameters below must be added to the [.bash\_profile(linux)/.profile(solaris)]

export ORACLE\_HOME=/u01/app/oracle/product/11.2.0.4/client\_1 (ORACLE\_HOME Location)

export LD\_LIBRARY\_PATH=$ORACLE\_HOME/lib

export PATH=$ORACLE\_HOME/bin:$PATH

## Wallet creation requirements/process

Now we will make backup copies of files we are due to change (in case of a rollback requirement)

cp $ORACLE\_HOME/network/admin/sqlnet.ora $ORACLE\_HOME/network/admin/sqlnet.ora-pre\_<CR>

cp $ORACLE\_HOME/network/admin/tnsnames.ora $ORACLE\_HOME/network/admin/tnsnames.ora-pre\_<CR>

Each user will have a password store placed in the directory for this reason it is essential that the OS user can resolve the variable ‘$ORACLE\_HOME’

$ORACLE\_HOME/network/admin /.orawallet

## Installation

### Update sqlnet.ora with the location of the password store

WALLET\_LOCATION = (SOURCE = (METHOD = FILE) (METHOD\_DATA = (DIRECTORY =$ORACLE\_HOME/network/admin/.orawallet) ) )  
SQLNET.WALLET\_OVERRIDE = TRUE

Or

echo 'WALLET\_LOCATION = (SOURCE = (METHOD = FILE) (METHOD\_DATA = (DIRECTORY =$ORACLE\_HOME/network/admin/.orawallet) ) ) SQLNET.WALLET\_OVERRIDE = TRUE' >> $ORACLE\_HOME/network/admin/sqlnet.ora

The placement of the wallet might not be the same on all servers, if a clustered server is used, the wallet must be placed in a common place.

Tradewarehouse (TWP) currently uses one for each login and is placed differently.

WALLET\_LOCATION = (SOURCE = (METHOD = FILE) (METHOD\_DATA = (DIRECTORY =$HOME/.orawallet$LOGNAME) ) )  
SQLNET.WALLET\_OVERRIDE = TRUE

### Create TNS alias in tnsnames.ora file used by Password Store

Update tnsnames.ora file with username entries (should be in the CRQ), the standard is  
<database|database\_service>\_<username> = tns string  
  
Eg: Database: TWD, Username: test  
twd\_test =(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = ora\_twd)(PORT = 1521))(CONNECT\_DATA =(SERVICE\_NAME = twd)))

### Create “Wallet” - Password Store

Create and protect directory to hold Password Store

mkdir -p $ORACLE\_HOME/network/admin/.orawallet

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -create

The above will create the following two files in $ORACLE\_HOME/network/admin/.orawallet

cwallet.sso  
ewallet.p12

chgrp -R orawallet $ORACLE\_HOME/network/admin/.orawallet

chmod -R 750 $ORACLE\_HOME/network/admin/.orawallet

Note: this must be done every time the wallet is touched (created or updated).

Next a “Wallet” password for the store is required as this will be needed when doing operations on the wallet.

The password must follow Nordea rules for ADM accounts:

<https://intservices.sed1.root4.net/portal/internalportal/appmanager/nordeaportal/desktop/Content/Notes/intranet/homepage/home0483.nsf/Released/b1cc2f34be4740c6c1257e3f00475756?OpenDocument&language=uk&linkCount=1&searchType=site>

### Create credential in wallet Password Store

Create credential with the command;

*mkstore -wrl <wallet\_location> -createCredential <db\_connect\_string> <username> [<password>]*

*e.g;*

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -createCredential <tns\_entry> <Username>  
  
It will first ask for the password for the user and then password for the wallet.  
Password can be given on command, line but it’s advised not to use it, since it can be seen in history,

Eg1:  
Unix twbatch  
Oracle user wallet\_test  
tnsentry/db\_connect\_string twd\_wallet\_test

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -createCredential twd\_wallet\_test wallet\_test

Oracle Secret Store Tool : Version 11.2.0.3.0 – Production  
Copyright (c) 2004, 2011, Oracle and/or its affiliates. All rights reserved.  
Your secret/Password is missing in the command line   
Enter your secret/Password: Tset\_tellaw  
Re-enter your secret/Password: Tset\_tellaw

Enter wallet password: <enter wallet password>Create credential oracle.security.client.connect\_string2

Eg2:  
Unix tmbatch  
Oracle user wallet\_test  
tnsentry/db\_connect\_string twd\_wallet\_test

mkstore -wrl .orawallettmbatch -createCredential twd\_wallet\_test wallet\_test  
…  
Enter your secret/Password: Tset\_tellaw   
…

### How to handle issues with PMSP (Cyberark)

There might be issues with creating wallet and wallet entries when connected with SSH through PSMP (Cyberark), this can be handled by using code lines like below (database\_username is the tnsentry in the wallet):

# Create wallet  
( read -s -p "Enter password for password store : " PASSWORD; echo ; mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -create <<< "$PASSWORD"$'\n'"$PASSWORD"$'\n'; unset PASSWORD )  
# Enter password for password store  
  
#Create entries into password store  
( read -s -p "Enter password for user : " UPASSWORD; echo ;read -s -p "Enter password for password store : " PASSWORD; echo ; mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -createCredential database\_username username<<< "$UPASSWORD"$'\n'"$UPASSWORD"$'\n'"$PASSWORD"$'\n'; unset UPASSWORD; unset PASSWORD )  
# Enter password for user  
# Enter password for password store

### How to setup a segregated wallet design

A segregated design for unix users has been created, if wallet is not allowed to be shared between osusers

Each os user will then have their own Wallet

Modify Sqlnet.ora to contain

WALLET\_LOCATION = (SOURCE = (METHOD = FILE) (METHOD\_DATA = (DIRECTORY =$ORACLE\_HOME/network/admin/.orawallet[$LOGNAME]) ) )

Create wallet at: $ORACLE\_HOME/network/admin/.orawallet<osuser name>

mkdir -p $ORACLE\_HOME/network/admin/.orawallet<osuser name>

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet<osuser name> -create

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -createCredential <tns\_entry> <Username>

Apply access security

chgrp -R orawallet<osuser name> $ORACLE\_HOME/network/admin/.orawallet<ouser name>

chmod -R 750 $ORACLE\_HOME/network/admin/.orawallet<osuser name>

## Test the connection

### Using the password store - Sqlplus

Eg1:  
twbatch

sqlplus /@twd\_wallet\_test  
show user  
select sys\_context('USERENV','INSTANCE\_NAME') FROM DUAL;

Eg2:  
tmbatch

sqlplus /@twd\_wallet\_test

Sqlplus /@< tnsnames.ora entry>  
  
Connect through (Proxy connect) to user\_a with wallet

Sqlplus [user\_a]@< tnsnames.ora entry >

Oracle database utilities

sqlldr parfile=connect\_file.par  
connect\_file.par:

userid=/@< tnsnames.ora entry >

JDBC thin

(example from Calypso setup on ap-calyp1t)  
Oracle 11.2.0.4 Clients needs to be installed

Setup Oracle env for calypt unixuser

LD\_LIBRARY\_PATH=/u01/app/oracle/product/11.2.0.4/client\_1/lib  
ORACLE\_HOME=/u01/app/oracle/product/11.2.0.4/client\_1  
PATH=:/u01/app/oracle/product/11.2.0.4/client\_1/bin:…  
  
Java auth file:  
 -Doracle.net.tns\_admin=$ORACLE\_HOME/network/admin

-Doracle.net.wallet\_location=$ORACLE\_HOME/network/admin/.orawallet

CLASSPATH="\

…

: $ORACLE\_HOME/jlib/oraclepki.jar\

: $ORACLE\_HOME/jlib/osdt\_cert.jar\

: $ORACLE\_HOME/jlib/osdt\_core.jar\

…

DBURL=jdbc:oracle:thin:/@calypth\_calypso

Calypso already has a ojdbc.jar in the classpath, but if it’s not already there, add the following as well:  
: $ORACLE\_HOME/jlib/ojdbc6.jar\

./jdbc/lib/ojdbc6.jar

./jdbc/lib/ojdbc7.jar

#### The calypso application requires that DBUSER and DBPASSWORD are also set, but this will override the Oracle wallet, so this needs to be commented out in DataSource.config.xml in /calypso/app/resources/appconfig <bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close"> <property name="username" value="${DBUSER}" /> <property name="password" value="${DBPASSWORD}" /> <property name="url" value="${DBURL}" /> … </bean>

#### Alter it to

#### <bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close"> <!-- < property name="username" value="${DBUSER}" /> <property name="password" value="${DBPASSWORD}" /> --> <property name="url" value="${DBURL}" /> … </bean>

### Other examples on JDBC thin

#### http://stackoverflow.com/questions/7634196/what-is-correct-jdbc-url-syntax-if-oracle-wallets-are-used http://tech.shopzilla.com/2011/09/oracle-wallet-with-thin-driver-with-connection-pool-with-database-timeouts/

## Managing the Password Store

Change the password

*orapki* wallet change\_pwd -wallet wallet\_location [-oldpwd password ] [-newpwd password]

-oldpwd and –newpwd is optional on commandline, but it’s advised not to use it, since it can be seen in history, orapki will ask for it, if not given.

Eg:

orapki wallet change\_pwd -wallet $ORACLE\_HOME/network/admin/.orawallet

### **List entries**

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -listCredential

Oracle Secret Store Tool : Version 11.2.0.3.0 - Production

Copyright (c) 2004, 2011, Oracle and/or its affiliates. All rights reserved.

Enter wallet password:

List credential (index: connect\_string username)

1: twd\_wallet\_test wallet\_test

Change password for stored credentials

mkstore -wrl  $ORACLE\_HOME/network/admin/.orawallet -modifyCredential <tnsnames.ora entry> <oracle username> [<new password>]

<Password> is optional on command line, but it’s advised not to use it, since it can be seen in history, mkstore will ask for it , if not given   
  
Eg:

mkstore -wrl  $ORACLE\_HOME/network/admin/.orawallet -modifyCredential twd\_wallet\_test wallet\_test

Update the rights for the wallet  
chmod –r 750 $ORACLE\_HOME/network/admin/.orawallet

Note: this must be done every time the wallet is touched (created or updated).

### Delete credential

mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -deleteCredential <tnsnames.ora entry>

Eg  
mkstore -wrl $ORACLE\_HOME/network/admin/.orawallet -deleteCredential twd\_wallet\_test

## ****Where do we save a record of the password to**** the ****External Password Store****

### Wallet’s password - store on the server.

Don't lose the wallet password, store on the server as hidden file and only access to the OS user Oracle.

Save it at : /me01/oracle/dbadmin/.OSPS-key<unixusername>  
Alternative place (for application servers)  
/u01/oracle/dbadmin/.OSPS-key<unixusername>  
or  
/u01/app/oracle/dbadmin/.OSPS-key<unixusername>  
Make sure only Oracle can read it  
chmod 700 /u01/app/oracle/dbadmin/.OSPS-key<unixusername>

### Wallet’s password - store on Nodbotp.

Four tables has been created on the sweeper schema, wallets should be also be inserted there

NODBOT.WALLET\_HOST

Hotname for Unix servers with Wallets

NODBOT.WALLET\_PASSWORD\_STORE  
Store password for Secure Password Stores (SPS)s here.

NODBOT.WALLET  
Store wallets entries here, there is a FK to SWEEPER.WALLET\_PASSWORD\_STORE. WSTORE\_SEQ

NODBOT.WALLER\_MEMBER  
Store osmembers that can read the wallet here.  
There is a FK to SWEEPER.WALLET\_PASSWORD\_STORE. WSTORE\_SEQ

Dont specify any sequence (\*\_SEQ) numbers, all has been replaced with GENERATED ALWAYS AS IDENTITY And will be created by Oracle

### Create new Entry

SQL to run, alter hostname,password,tnsname,and membername etc to correct value.

-- INSERTING into Secure password store and Wallet

-- Dont specify any \*\_SEQ numbers, all has been replaced with GENERATED ALWAYS AS IDENTITY

-- And will be created by Oracle

Insert into WALLET\_HOST (HOSTNAME) values ('ap-dse15d.oneadr.net');

Insert into WALLET\_PASSWORD\_STORE (HOSTNAME,WALLET\_OS\_OWNER,PASSWORD,ORACLE\_HOME,WALLET\_LOCATION,WALLET\_HOST\_SEQ)

select 'ap-dse15d.oneadr.net','oracle','<password>','/u01/app/oracle/product/12.2.0/client\_1','(SOURCE = (METHOD = FILE) (METHOD\_DATA = (DIRECTORY =$ORACLE\_HOME/network/admin/.orawallet) ) )',WALLET\_HOST\_SEQ

from WALLET\_HOST where hostname='ap-dse15d.oneadr.net';

-- Wallet\_os\_owner default owner is oracle, only speficy one if not Oracle

-- This is NOT OS user on application server that can read the wallet

-- WALLET\_OS\_GROUPNAME default os orawallet, only speficy one if not orawallet

Insert into WALLET (TNSNAMES,DB\_USERNAME,CHANGE\_ID,WSTORE\_SEQ)

select 'ARC733D\_DB\_DATASTORE\_USER\_PREPROD=(DESCRIPTION=(LOAD\_BALANCE=OFF)(FAILOVER=ON)(CONNECT\_TIMEOUT=5)(TRANSPORT\_CONNECT\_TIMEOUT=3)(RETRY\_COUNT=3)(ADDRESS\_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=ora-arc733d-a.qaoneadr.local)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=ora-arc733d-b.qaoneadr.local)(PORT=1521)))(CONNECT\_DATA=(SERVICE\_NAME=arc733d)(FAILOVER\_MODE=(TYPE=SELECT)(METHOD=BASIC))))','DB\_DATASTORE\_USER\_PREPROD','CRQ000081670913',a.WSTORE\_SEQ

from WALLET\_PASSWORD\_STORE a, WALLET\_HOST b

where a.wallet\_os\_owner='oracle' and b.hostname='ap-dse15d.oneadr.net'

and a.wallet\_host\_seq=b.wallet\_host\_seq;

Insert into WALLET\_MEMBER (WALLET\_OS\_MEMBERNAME,WALLET\_OS\_GROUPNAME,WSTORE\_SEQ,WALLET\_HOST\_SEQ)

select 'dsadm','orawallet',a.WSTORE\_SEQ,b.WALLET\_HOST\_SEQ

from WALLET\_PASSWORD\_STORE a, WALLET\_HOST b

where a.wallet\_os\_owner='oracle' and b.hostname='ap-dse15d.oneadr.net'

and a.wallet\_host\_seq=b.wallet\_host\_seq;

-- Only one OS user is alloweed in WALLET\_OS\_MEMBERNAME pr row.

### Update Existing Entry

If wallet have already installed on the server and we adding more wallet entries there, then we need to update the repository for every additional wallets entry using followings;

-- INSERTING new Wallets into existing Secure password store

Insert into WALLET (TNSNAMES,DB\_USERNAME,CHANGE\_ID,WSTORE\_SEQ)

select 'ARC733D\_DB\_DATASTORE\_USER\_PREPROD=(DESCRIPTION=(LOAD\_BALANCE=OFF)(FAILOVER=ON)(CONNECT\_TIMEOUT=5)(TRANSPORT\_CONNECT\_TIMEOUT=3)(RETRY\_COUNT=3)(ADDRESS\_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=ora-arc733d-a.qaoneadr.local)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=ora-arc733d-b.qaoneadr.local)(PORT=1521)))(CONNECT\_DATA=(SERVICE\_NAME=arc733d)(FAILOVER\_MODE=(TYPE=SELECT)(METHOD=BASIC))))','DB\_DATASTORE\_USER\_PREPROD','CRQ000081670913',a.WSTORE\_SEQ

from WALLET\_PASSWORD\_STORE a, WALLET\_HOST b

where a.wallet\_os\_owner='oracle' and b.hostname='ap-dse15d.oneadr.net'

and a.wallet\_host\_seq=b.wallet\_host\_seq;

Insert into WALLET\_MEMBER (WALLET\_OS\_MEMBERNAME,WALLET\_OS\_GROUPNAME,WSTORE\_SEQ,WALLET\_HOST\_SEQ)

select 'dsadm','orawallet',a.WSTORE\_SEQ,b.WALLET\_HOST\_SEQ

from WALLET\_PASSWORD\_STORE a, WALLET\_HOST b

where a.wallet\_os\_owner='oracle' and b.hostname='ap-dse15d.oneadr.net'

and a.wallet\_host\_seq=b.wallet\_host\_seq;

-- Only one OS user is allowed in WALLET\_OS\_MEMBERNAME pr row.