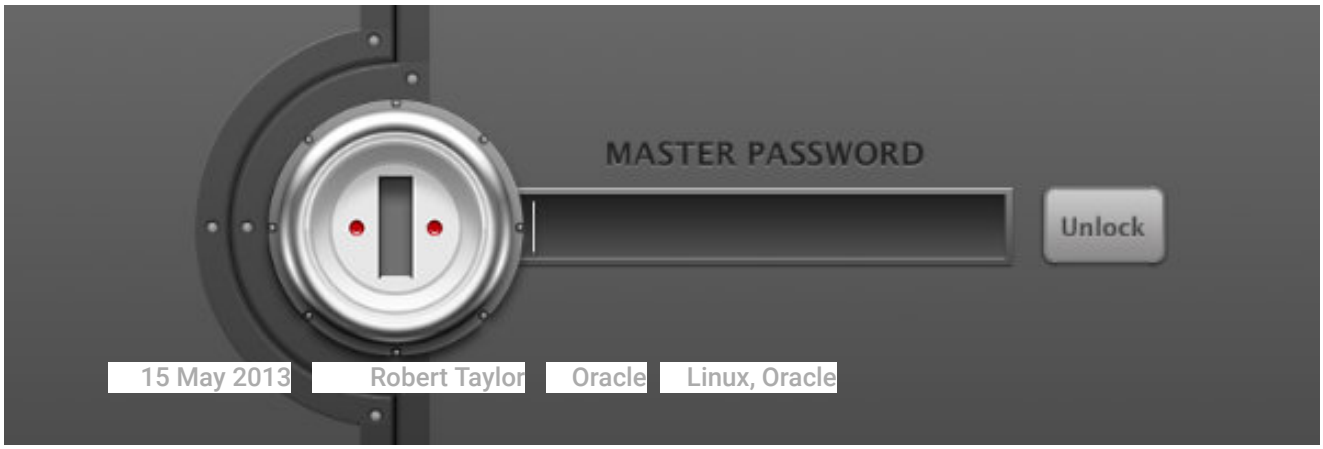

A guide to Oracle Wallet

It is often necessary to make connections to the database from shell scripts held on the filesystem. This can be a major security issue if these scripts contain the database connection details. One solution is to use OS Authentication, but Oracle 10g Release 2 gives us the option of using a secure external password store where the Oracle login credentials are stored in a client-side Oracle wallet. This allows scripts to contain connections using the `/@db_alias` syntax.

The wallet is simply a directory on the server where the passwords are written (in an encrypted form) by the oracle `mkstore` commaand. You tell Oracle where to find the wallet by configuring specific parameters in the `sqlnet.ora` file and you retrieve/use a stored password by referencing a

TNS alias configured in your `tnsnames.ora` file (detailed below). There are no services to start or stop, and nothing to be installed.

Creating a Wallet

Use the `mkstore` command on an empty directory as follows:

```
mkdir -p /oracle/admin/DBNAME/wallet
mkstore -wrl /oracle/admin/DBNAME/wallet -create
```

You will be prompted for a password to secure the wallet. Make sure it is something secure, and record the password in your central password store.

Next, add the following lines to your sqlnet.ora configuration file.

```
SQLNET.WALLET_OVERRIDE = TRUE
WALLET_LOCATION=(
  SOURCE=(METHOD=FILE)
  (METHOD_DATA=(DIRECTORY=/oracle/admin/DBNAME/wallet))
)
```

Note: There are implications for both Clusterware and OS authentication when using wallet_override:, so please see the section "Known Issues / Gotchas" at the end of the article)

Adding a username and password to the wallet

Before adding the username and password, we create an alias in the `tnsnames.ora` file that will be used whenever we want to log in using the stored credentials. Only one password may be stored in the wallet per TNS alias: In our example below, we have created an alias called "DBFS":

```
# Connectiong string for DBFS Oracle Wallet
DBFS =
(DESCRIPTION =
(ADDRESS=(PROTOCOL=TCP)(HOST=`hostname`-vip)(PORT=1528))
(CONNECT_DATA=(SID=PMLOC1_1))
)
```

Now to add a username and password to an existing wallet, use the mkstore command with the -createCredential option as follows:

```
mkstore -wrl <wallet_location> -createCredential <TNS_al
```

Example:

```
mkstore -wrl /oracle/admin/DBNAME/wallet -createCredenti
```

Testing the Wallet

That's it, your wallet is created and you've stored a username and password inside it. Now all you need to do is test it using the TNS alias you setup (DBFS in our example):

```
sqlplus /@DBFS
```

Administering the Wallet

Listing credentials stored in the wallet:

```
mkstore -wrl <wallet_location> -listCredential
```

Modifying credentials stored in the wallet:

```
mkstore -wrl <wallet_location> -modifyCredential <dbase_
```

Deleting credentials stored in the wallet:

```
mkstore -wrl <wallet_location> -deleteCredential <db_al:
```

Deleting the whole wallet:

```
rm -rf <wallet_location>
```

A NOTE ON SECURITY

Remember that any user that has access to the wallet can use any password stored in the wallet. Therefore it is recommended that you create one wallet per user, rather than using a common wallet.

To that effect, I would recommend saving a sqlnet.ora and tnsnames.ora configured for wallet access separately from the common oracle TNS configuration. For example we could copy both files to the wallet directory, then set the environmental variable TNS_ADMIN to point to our wallet directory at the start of any script that needs to use the wallet.

Also, it is important to remember that **the security of the wallet is only-file-based.** Thus the security of the wallet is only marginally better than

a hard-coded password within a shell script as both methods depend on OS file and directory permissions for their security. There is nothing to stop an attacker copying the wallet to another machine (if they have read access) and using it to authenticate with the database.

KNOWN ISSUES / GOTCHAS

#1 Using OS Authentication results in ORA-01017: Invalid username/password

If the WALLET_OVERRIDE=TRUE parameter is present in your sqlnet.ora file that any attempt to use OS authentication will result in an ORA-01017 error as shown below:

```
$ sqlplus /
```

```
SQL*Plus: Release 10.2.0.2.0 - Production on Thu Aug 10
Copyright (c) 1982, 2005, Oracle. All Rights Reserved.
```

```
ERROR:
ORA-01017: invalid username/password; logon denied
```

You cannot connect to Oracle using an external password in conjunction with the secret store. That would defeat the purpose. When using OS Authentication (an external password), your account is already authenticated to Oracle because the operating system has authenticated you. There would be no reason to keep an encrypted set of of credentials for you.

The sqlnet.ora parameter SQLNET.WALLET_OVERRIDE=TRUE is laterally telling Oracle client to use the wallet manager instead of OS

Authentication.

If you want to connect to the same database with different accounts, then you would need separate entries in the tnsnames.ora file; however, you should be setting up services for this purpose and creating a new tnsnames.ora entry for each service. Connecting to the SID of the database is a <8i methodology and should not be practiced anymore.

Also, if you are using multiple accounts, each account should have its own wallet anyway. Therefore, they could share a common tnsnames.ora file, but use different credentials based on the entries in the wallet.

#2 Wallet and Grid Infrastructure (Clusterware)

If you add the WALLET_OVERRIDE parameter to the sqlnet.ora file used by the Grid Infrastructure the crsd service will fail during cluster initialisation and the logs will report misleading errors about being unable to read/access the OCR. See metalink Note 1153244.1

(I suspect this is directly related to issue #1 above, and the clusterware is using OS authentication during initialisation).

Moving the Start screen between multiple monitors

Installing FUSE on linux (for DBFS)

OR SIGN UP WITH DISQUS ?

Name

ajay govind • 6 months ago

If i have multiple users for same database (TNS Entry) then how would sqlplus responds.?
Example:- Database is testdb and the users are test1 and test2 and are created in different wallets as tns name has to be unique per wallet. But how would 'sqlplus /@testdb' would respond, which user to will it connect?
How to handle this multiple user for same database scenario?

▲ ▼ • Reply • Share ›

Robert Taylor Mod ➚ ajay govind • 6 months ago

SQL*Plus will use the wallet that has been configured in the sqlnet.ora file. If you want your users to use different wallets, then you will need to ensure they are using different sqlnet.ora files. -- see https://www.dba-resources.c... for details on how oracle decides which sqlnet.ora / tnsnames.ora files to use

▲ ▼ • Reply • Share ›

ajay govind ➚ Robert Taylor • 6 months ago

Thanks for your reply, it helped me.
What about sqlldr? It also needs db password to connect and i want sqlldr to get that password from oracle wallet.
When i used "sqlldr [test1]@testdb
control = /home/infa_sgo/smarttrial_ODR_setup/sqlldr-add-new.ctl"
it asked password..But I want to handle everything through shell script.
Kindly suggest how to handle this one.
Thanks in advance...

▲ ▼ • Reply • Share ›

Robert Taylor Mod ➚ ajay govind • 6 months ago

sqlldr and sqlplus function in the same way, so you need to ensure you are using the correct tnsnames.ora and sqlnet.ora files.

Is that a typo in your comment?

> `[test1]@testdb`

is not a wallet style connection - it should be

> `/@testdb`

▲ ▼ • Reply • Share ›

zarafiq • a year ago

Thanks for this excellent article!

Most other articles on the subject forget to mention this very important part: "There is nothing to stop an attacker copying the wallet to another machine (if they have read access) and using it to authenticate with the database."

Reply • Share ›

Emmanuel Petit • 2 years ago

great stuff truly helpful.

Reply • Share ›

Jeff • 3 years ago

I think you have a typo in this article. Under the "Creating a wallet" section, I believe the second line should be "mkstore", not "mkdir".

Reply • Share ›

Robert Taylor Mod → Jeff • 3 years ago

Thanks Jeff, well spotted!
I've updated the post.

Reply • Share ›

ALSO ON **DBA RESOURCES**