Oracle

- Acces base oracle
- Vérifier et débloquer un compte verrouillé
- Problème demarrage timezone

Acces base oracle

Sudo su - oracle
Setup l'affichage et voir les session connecte à la basse :

```
SET pages 400 lines 200
COLUMN spid FORMAT A10
COLUMN username FORMAT A20
COLUMN program FORMAT A45
```

```
SELECT s.inst_id,
       s.sid,
       s.serial#,
       p.spid,
       s.username,
       s.program
FROM   gv$session s
       JOIN gv$process p ON p.addr = s.paddr AND p.inst_id = s.inst_id
WHERE  s.type != 'BACKGROUND';
```

Générer le path et sid

```
export PATH=$PATH:/app/oracle/product/11.2.0.4/dbhome_1/bin/
export ORACLE_SID=Nom database
export ORACLE_HOME=/app/oracle/product/11.2.0.4/dbhome_1/
```

. oraenv mettre le sid et path en rapport

```
sqlplus / as sysdba
```

```
drop user TABLE cascade;
```

Vérifier et débloquer un compte verrouillé

- Se connecter au serveur en SSH via putty ou winscp (> putty).
- Se connecter en tant que compte oracle
su oracle

- Ajouter les arguments pour les variables d'environnement pour se connecter à l'instance

```
export ORACLE_HOME=/u01/app/oracle/product/11gr2/db_1
export PATH=$PATH:$ORACLE_HOME/bin
export ORACLE_SID=gcldev
```

- Se connecter à l'instance SQL:

```
sqlplus /nolog
connect / as sysdba
```

- Lister tous les comptes de l'instance SQL :

```
SELECT username, account_status FROM dba_users;
```

- Lister tous les comptes avec leurs dates de création, date d'expiration et l'état de verrouillage :

```
SELECT username, account_status, created, lock_date, expiry_date  FROM dba_users  WHERE account_status != 'OPEN';
```

- Requête pour débloquer un compte (nomducompte à remplacer) :

```
ALTER USER nomducompte ACCOUNT UNLOCK;
```

- Note en plus pour reset un password :

```
ALTER USER nomducompte BY new_password;
```

Problème demarrage timezone

```
ps -ef | grep pmon
```

```
[root@prdoran1 diag]# ps -ef | grep pmon
oracle    5547     1  0 12:15 ?        00:00:00 ora_pmon_PIMS1
oracle   12253     1  0 12:23 ?        00:00:00 ora_pmon_RECPIMS1
grid     16367     1  0 11:25 ?        00:00:00 asm_pmon_+ASM1
grid     16851     1  0 11:25 ?        00:00:00 apx_pmon_+APX1
oracle   16863     1  0 11:25 ?        00:00:01 ora_pmon_ADIX51
oracle   16867     1  0 11:25 ?        00:00:01 ora_pmon_BIDATA1
oracle   16913     1  0 11:25 ?        00:00:01 ora_pmon_PRDBD111
oracle   17311     1  0 11:25 ?        00:00:01 ora_pmon_RHINTERM1
oracle   17406     1  0 11:26 ?        00:00:01 ora_pmon_PRDWEB1
oracle   17416     1  0 11:26 ?        00:00:01 ora_pmon_NODHOS1
oracle   17607     1  0 11:26 ?        00:00:01 ora_pmon_PRDBD1
root     27555 26395  0 13:56 pts/4    00:00:00 grep --color=auto pmon
```

Pour redémarrer les bases manuellement :

```
export ORACLE_HOME=/app/oracle/product/10.2.0.5SE/dbhome_1
export ORACLE_BASE=/app/oracle/product/10.2.0.5SE/dbhome_1
export ORACLE_SID=PIMS1
sqlplus / as sysdba
```

Démarrer la base

```
startup
```

Pour arrêter la base:

```
shutdown immediate
```

==========

Lorsque l'on exécute la commande (sqlplus / as sysdba) surtout sur la base l'on obtient le message d'erreur suivant :

```
```

> ORA-01804: failure to initialize timezone information

Solution :

Prendre un autre fichier timezone sur n'importe quel répertoire :

```
export ORA_TZFILE=/app/oracle/product/11.2.0.4/dbhome_1/oracore/zoneinfo/timezlrg_14.dat
```

==========

Pour supprimer les logs de listener de plus de Xjour (exemple +24)

```
cd /app/oracle/diag/tnslsnr/prdoran1/listener/alert
find . -name "log_*.xml" -ctime +24 -exec rm -f  {} \;
```

Pour lister les traces +15 j :

```
 find . -name "*.trc" -ctime +15 -exec ls -l  {} \;
```

Avant de remplacer "ls -l" par "rm -f" assure toi que la base est démarrée depuis plus de 15 j
==========
Il existe des bases selon la version d'oracle, pour les lister :

```
cd /app/oracle/product/
```

Ensuite la version ex :

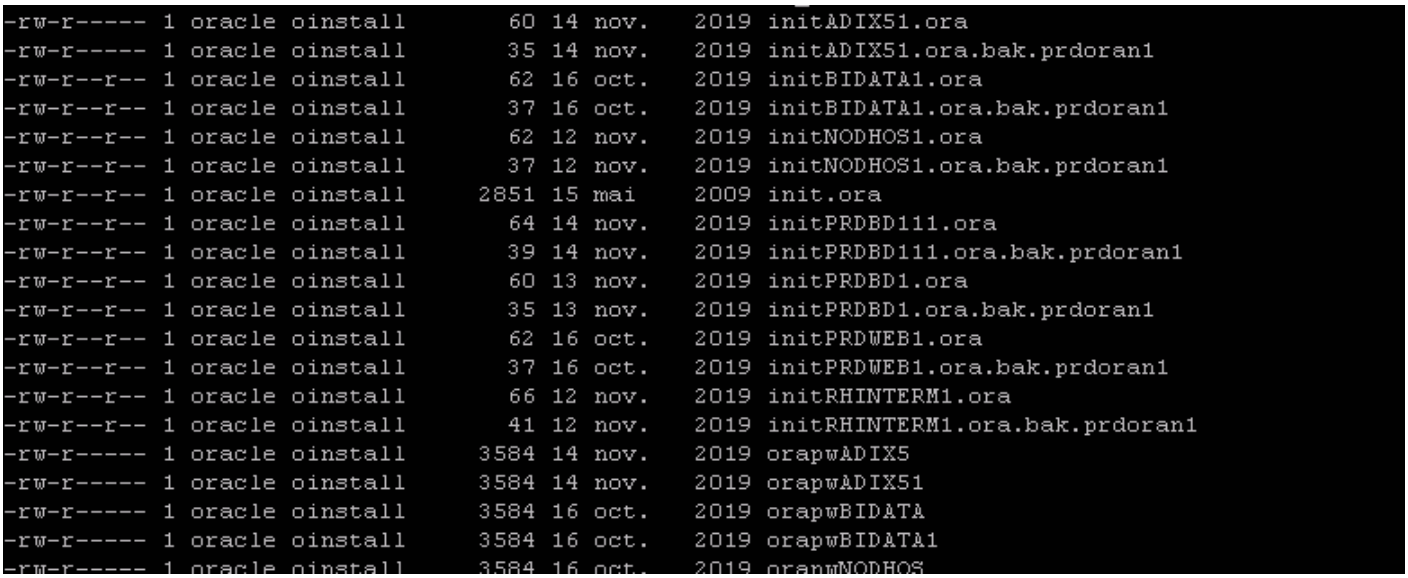
```
cd /app/oracle/product/11.2.0.4/dbhome_1/dbs/
```

Ce qui nous intéresse c'est le init* pour savoir le nom de la base (ex: la base initNODHOS1.ora ça représente la base NODHOS1 donc pour l'utiliser on doit lancer l'export :

```
export ORACLE_HOME=/app/oracle/product/11.2.0.4/dbhome_1
```

ensuite :

```
export ORACLE_SID=NODHOS1
```



```
-rw-r----- 1 oracle oinstall       60 14 nov.   2019 initADIX51.ora
-rw-r----- 1 oracle oinstall       35 14 nov.   2019 initADIX51.ora.bak.prdoran1
-rw-r--r-- 1 oracle oinstall       62 16 oct.   2019 initBIDATA1.ora
-rw-r--r-- 1 oracle oinstall       37 16 oct.   2019 initBIDATA1.ora.bak.prdoran1
-rw-r--r-- 1 oracle oinstall       62 12 nov.   2019 initNODHOS1.ora
-rw-r--r-- 1 oracle oinstall       37 12 nov.   2019 initNODHOS1.ora.bak.prdoran1
-rw-r--r-- 1 oracle oinstall     2851 15 mai    2009 init.ora
-rw-r--r-- 1 oracle oinstall       64 14 nov.   2019 initPRDBD111.ora
-rw-r--r-- 1 oracle oinstall       39 14 nov.   2019 initPRDBD111.ora.bak.prdoran1
-rw-r--r-- 1 oracle oinstall       60 13 nov.   2019 initPRDBD1.ora
-rw-r--r-- 1 oracle oinstall       35 13 nov.   2019 initPRDBD1.ora.bak.prdoran1
-rw-r--r-- 1 oracle oinstall       62 16 oct.   2019 initPRDWEB1.ora
-rw-r--r-- 1 oracle oinstall       37 16 oct.   2019 initPRDWEB1.ora.bak.prdoran1
-rw-r--r-- 1 oracle oinstall       66 12 nov.   2019 initRHINTERM1.ora
-rw-r--r-- 1 oracle oinstall       41 12 nov.   2019 initRHINTERM1.ora.bak.prdoran1
-rw-r----- 1 oracle oinstall     3584 14 nov.   2019 orapwADIX5
-rw-r----- 1 oracle oinstall     3584 14 nov.   2019 orapwADIX51
-rw-r----- 1 oracle oinstall     3584 16 oct.   2019 orapwBIDATA
-rw-r----- 1 oracle oinstall     3584 16 oct.   2019 orapwBIDATA1
-rw-r----- 1 oracle oinstall     3584 16 oct.   2019 orapwNODHOS
```

Après on va pouvoir lancer sqlplus en tant que NODHOS1 et démarrer la base