**STANDBY/DATAGUARD**

**StandBy Database (DATAGUARD)** :- It enables very high-speed movement of data and metadata from one database to another means providing Disaster recovery, Complete automatic data protection, and High availability. Servers working on Oracle 11g database with base OS Oracle Linux Enterprise. One is currently open and Up On which huge no. of transaction are going on and while other is in mount mode with Different geographical locations.

As automate the backups of archives& datafiles from server 1 to server 2, Whenever if there is problem in currently up server 1 or get destroyed in disaster or burnt out or require maintenance so to avoid failure of ongoing transactions decrease or decrease in hit ratio or to maintain no time lag or server down problem, then with few commands to run the server 2 will automatically up and working as primary server So no loss of ongoing transactions and Data Loss.

**Base OS :-** Oracle Linux Enterprise.

**Technology :-** Oracle 11g, Oracle Data Pump.

**Tools :-** Linux, Type-2 Hypervisor, Putty.

Two VMWARE Nodes having base OS as ORACLE LINUX ENTERPRISE

1. Primary DB :- Oracle Linux Enterprise on which Oracle Universal Installer (OUI) and Database COnfriguration Asssistant (dbca) should be installed.

2. Secondary DB:- Only Oracle Linux Enterprise Later this will be converted into StandBy Database or DATAGUARD.

what is standby database..

:: suppose there is a db p&g this is primary or target db and it is up and all users which are connected are able to transact

So for user connectivity to this the db should be open.

Until we done whatever recovery is media recovery like files are corrupted or lost which recovered through backups on server machine

:: but suppose somehow due to any disaster building in which server machine is kept is destroyed or server machine of database is burn.

Now db is lost even we have archive in tapes which are of no use because we don’t have base db on which db we apply this archives

:: to solve this we create another database which connected with base db So what ever trans happen on base will be automatically happens on this another db on together different location

That another db should mount or open ?

that will be on MOUNT(stand by database will always put on mount) not on open because when it is on open then datafile and redofile is available then transact is taking place and generate SCN no.s and these SCN number will not match with primary database

if both are open then both db become primary and work individually so stand by database will always put on mount

As the base db controlfile are updating so is base controlfile will same or different for standby db

We require archives to be updated only for standby so controlfile for stby are different so lets we call them stby controlfile

As base db will generate stby controlfile with some command On which we apply archives which are not destroyed.

Otherwise if both are the Master controlfile then how no will obeys each other

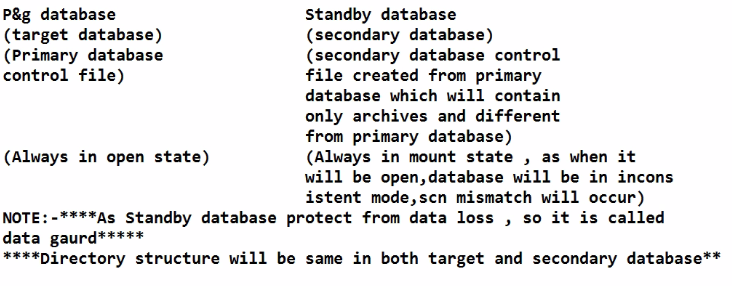
:: because data is safe in standby database that why it is called dataguard..

:: directory structure will be same on both stby and base db because the exactly the same ORACLE\_HOME and oracle\_base are copied on stby from base db

We cannot use template because the time may varies

Both are same but base db will in open mode while stby is in mount mode

Base will apply the transact while stby will only listen the instruction



**PRACTICAL**

**PRIMARY DATABASE**

**SQL>startup**

For standby db archives are imp so base db must enabled archive mode

check that primary database is enabled archive mode if not than put..

**SQL> archive log list;**

set the archive to a different location (So to check multiplexing of archives )

create a new folder with oracle user So to avoid the conflict of permission take care that you should not login with root user if the switch it to oracle user using “su – oracle”

**[oracle@akku ~]$ mkdir -p /u01/app/arch**

**[oracle@akku ~]$ cd ..**

**[oracle@akku ~]$ ls –l**

To check the current destination of archive logs

**SQL>show parameter log\_archive\_dest;**

Almost 31 destination will shown so if four destination is set then ARCH process will make copy of all 4 set destinations means doing multiplexing so on every logswitch 4 copies will created so for four destination is set so one of destination is set by us now

:: setting of archive destination

**SQL> alter system set log\_Archive\_dest\_2='location=/u01/app/arch' scope=spfile;**

:: now startup force so that new destination of arch will be updated

SQL>startup force

By default location of archives is FRA when FRA is configured and no other destination is not given

But if the destination is set the oracle will not go to FRA for archives it will go new destination will set

For multiplexing you can set **log\_Archive\_dest\_3, log\_Archive\_dest\_4 etc**

Now do some logswitches 2 or 3 archives ( archives are going to above location and extension will be **.dbf** if archive location is other than FRA

**SQL>alter system switch logfile ;**

**SQL>/**

**SQL>/**

Now to check where is Archive are created

**SQL>desc dictionary;**

**SQL>select table\_name from dictionary where table\_name like ‘%ARCHIVE%’ ;**

**SQL>desc V$ARCHIVE\_DEST;**

**STANDBY DATABASE**

create a new folder with root user as it is a fresh having linux

**[root@gaurav u01]# mkdir -p /u01/app**

change the owner ship to oracle:dba

**[root@gaurav u01]# chown -R oracle:dba app**

create a new folder with oracle user

**[root@gaurav u01]#su - oracle**

**[oracle@gaurav ~]$ mkdir -p /u01/app/arch**

**[oracle@gaurav ~]$cd /u01**

**[oracle@gaurav ~]$ls –l (to check ownership)**

now create **oracle\_base** location of base db on stby db location must be same as base db with oracle user

**[oracle@gaurav ~]$mkdir -p /u01/app/oracle/oradata/prod (here prod is your db name)**

now create **oracle\_home** location of priamry database on standby database location must be same as base db with oracle user

**[oracle@gaurav ~]$mkdir -p /u01/app/oracle/product/11.2.0** (dbhome will not copied here)

As now we have to copy datafiles from base db to stby db but due various transac going on then it is not possible copy the file without freezing the transac

So we have to **freeze the header of PRIMARY DATABASE** to maintain consistency of SCN no

**PRIMARY DATABASE**

**SQL>alter database begin backup;**

As now we have copy datafiles from base db to stby db so we require the ip and username to login using scp command

to copy the data file from primary database to standby database goto location of stndby database's base location..

**STANDBY DATABASE**

**[oracle@gaurav 11.2.0]$ cd /u01/app/oracle/oradata/prod/ (prod is db name)**

**[oracle@gaurav prod]$ scp oracle@192.168.232.128:/u01/app/oracle/oradata/prod/\* .**

here the ip is of primary database..and scp is linux command stand for secure copy

**PRIMARY DATABASE**

create controlfile for standby database from primary database

**SQL> alter database create standby controlfile as '/u01/app/oracle/oradata/prod/stdby\_control.ctl';**

this command will create **stdby\_control.ctl** file with primary database's files..

**STANDBY DATABASE**

Now copy stdby controlfile to standby database..

**[oracle@gaurav prod]$ scp oracle@192.168.232.128:/u01/app/oracle/oradata/prod/stdby\_control.ctl .**

**PRIMARY DATABASE**

end backup from primary database

**SQL> alter database end backup;**

Check whether archives are going to above location or not(not neccesary)

if archives are in the **fra** ,then move it to **/u01/app/arch** folder by cp command

**[oracle@gaurav arch]$cd /u01/app/oracle/fast\_recovery\_area/**

**[oracle@gaurav fast\_recovery\_area]$ls**

**[oracle@gaurav fast\_recovery\_area]$cd PROD**

**[oracle@gaurav PROD]$ls**

**[oracle@gaurav PROD]$cd archivelog**

**[oracle@gaurav archivelog]$ls**

**[oracle@gaurav archivelog]$** **cd 2016\_04\_02 (go to date folder and copy all archives)**

**[oracle@gaurav 2016\_04\_02]$cp \*.\* /u01/app/arch**

**[oracle@gaurav 2016\_04\_02]$cd ..**

**[oracle@gaurav archivelog]$** **ls**

**[oracle@gaurav archivelog]$** **cd 2016\_04\_03**

**[oracle@gaurav 2016\_04\_03]$cp \*.\* /u01/app/arch**

**[oracle@gaurav 2016\_04\_03]$cd /u01/app/arch**

**[oracle@gaurav 2016\_04\_03]$ls**

**STANDBY DATABASE**

copy all the archive file from base db to stby db

**[oracle@gaurav arch]$ scp oracle@192.168.232.128:/u01/app/arch/\* .**

**[oracle@gaurav arch]$ls**

go to location of **dbhome** at sdby db

**[oracle@gaurav 11.2.0]$ cd /u01/app/oracle/product/11.2.0**

**[oracle@gaurav 11.2.0]$ ssh oracle@192.168.232.128 "(cd /u01/app/oracle/product/11.2.0;tar -cf - dbhome\_1|gzip)"|gunzip|tar -xf -**

If password is asked then syntax is correct otherwise syntax is wrong

**description-->**

ssh--secure shell command

tar--target archive used to it store all file which are written with it and create only one file id with it so.. here supose there are 1000 file in dbhome\_1 folder if we didn't use tar than 1000 inode entry is genrated..to avoid this we use tar command..

as we create as many files in harddisk as the range of FILE\_ID of the OS

-cf-- to keep original as it is..

-xf-- extract

**[oracle@gaurav 11.2.0]$du –sh (check the size these files must equal to that files size of base db)**

As all files DBHOME is copied on stby db means we can run all binary commands like dbca or sqlplus etc for that we have update the .bash\_profile of stby database with .bash\_profile of base db means copy all data of .bash\_profile of base db to .base\_profile of stby db

Open .base\_profile of db in two different vi editor

**STANDBY DATABASE**

:: run the bash profile . ./.bash\_profile

**[oracle@gaurav ~]$. ./.bash\_profile**

Is there requirement for listener on which side stby db or base side and why ?

Because base db need contact with stby db to store all its archive on stby db so listener is require on stdb side

So there must be listener on stby database as base db is client for stby db as base db require to store its archive on stby db

**STANDBY DATABASE**

If listener is started on stby side it means we ca run binary on this side

So start the listener

**[oracle@gaurav ~]$lsnrctl start**

**[oracle@gaurav ~]$sqlplus / as sysdba (if this not run then open this in new instance of putty)**

**SQL> create pfile from spfile;**

open pfile in vi editor and change the controlfile path with stdby control file..

open **initorcl.ora** in vi editor and delete one path of control file

and change it to

**[oracle@gaurav ~]$cd /u01/app/oracle/product/11.2.0/dbhome\_1/dbs/**

**[oracle@gaurav ~]$ls**

**[oracle@gaurav ~]$vi initorcl.ora**

**\*.control\_files='/u01/app/oracle/oradata/prod/stdby\_control.ctl'**

**//add two new parameter**

**\*.standby\_archive\_dest='/u01/app/arch'** -->to set destination of archive

**\*.standby\_file\_management='AUTO'**  -->to files management automatically

Means what ever change in file must be reflect on stby db automatically

File close

**comment db\_recovery\_file\_dest parameter**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*NOTE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**NOT run this command but this command is very useful because in industry if don’t have access of password of sys user then this command is used because sometime senior dba don’t give password of sys user the junior dba can change the password using it when database is down**

**Because we can up db without password But we db is up we change the password using command**

If password file is not present then to create password file fire the following command on root (this is only method to set password of sys user when database is down )

#**orapwd=/u01/app/oracle/product/11.2.0/dbhome\_1/dbs/orapwprod password=oracle**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

remove all file from dbs folder other than pfile **initprod.ora**

and password file **prodpwprod.ora**..

**[oracle@gaurav ~]$cd /u01/app/oracle/product/11.2.0/dbhome\_1/dbs/**

**[oracle@gaurav ~]$ls**

**[oracle@gaurav dbs]$ rm -rf hc\_DBUA0.dat init.ora lkPROD spfileprod.ora hc\_prod.dat**

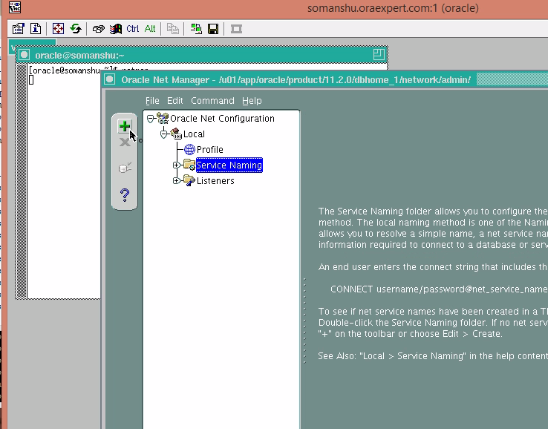
**PRIMARY DATABASE**

create a tns entry of stby db on primary database..

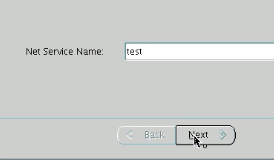
**[oracle@gaurav ~]$vncserver :1**

Open tight vnc VNC host of primary database

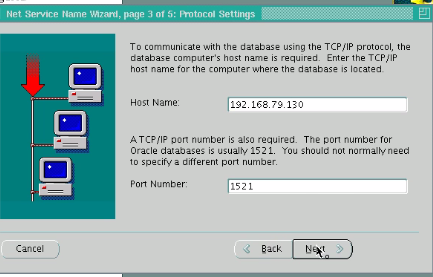
**[oracle@gaurav ~]$netmgr**



Create tns entry of “**test**” name



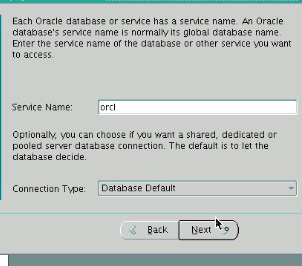
1. host name(ip) will be of stndby datanbase

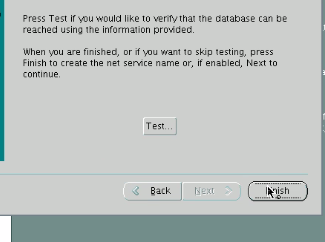


1. service name will be of prmry db global id here it is **(prod)**

if not known then check it through listener by start it and check it through **[oracle@gaurav ~]$lsnrctl start [oracle@gaurav ~]$lsnrctl status**

1. if tns enrty is not correct then archives cannot be copied on stby db side

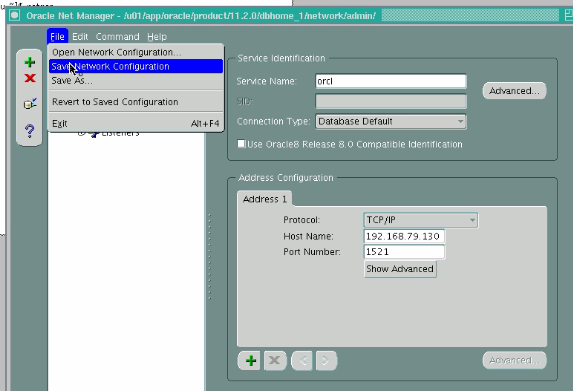
**(prod)**



**service name:- test**

**host name:- standby ip**

**net service name:- prod**



At last save and exit close the tight VNC

Run on primary database

**[oracle@gaurav ~]$lsnrctl start**

**[oracle@gaurav ~]$tnsping test ( to check tns entry is working right or not)**

**STANDBY DATABASE**

**[oracle@gaurav ~]$cd /u01/app/oracle/product/11.2.0/dbhome\_1/dbs/**

**[oracle@gaurav ~]$ls**

As we have created dbhome in stby db but more folders also create on primary db kike

**adump,bdump,cdump, udump**  which must be created on stby db

create adump,bdump,cdump folder on standby database if not created mannualy and check path in pfile

**[oracle@gaurav ~]$cat initprod.ora**

**[oracle@gaurav ~]$mkdir -p /u01/app/oracle/admin/prod/adump**

(location of bdump is same of udump run **SQL> show parameter background\_dump\_dest** on base db **to get location** )

**[oracle@gaurav ~]$mkdir -p /u01/app/oracle/diag/rdbms/prod/prod /trace** ( show parameter background\_dump\_dest)

(location of cdump run **SQL> show parameter core** on base dbto get location)

**[oracle@gaurav ~]$mkdir -p /u01/app/oracle/diag/rdbms/ prod/prod/cdump**  ( show parameter core and tab )

**PRIMARY DATABASE**

Now the archive which are locally made must be sent to remotely to stbt db

to set that archive will also go to stndby db

**SQL> alter system set log\_archive\_dest\_3="SERVICE=TEST";**

**SQL> startup force** (to primary database)

**SQL> alter system set standby\_file\_management=auto scope=spfile;**

**STANDBY DATABASE**

**SQL> startup nomount**  on stby database after base db will start

**SQL>alter database mount standby database;** --> to mount the standby database

As now check that if any new archive is made on primary db side on startup or shutdown time if it made then copy it stby db side in “/u01/app/arch”

**[oracle@gaurav ~]$cd /u01/app/arch/**

**[oracle@gaurav ~]$scp oracle@192.168.79.129:/u01/app/arch/\* .**

**SQL> recover automatic standby database;** this command will check for archive if any of them is not applied to stndby

Description:- If it is asking for new archive and if it is not made, then cancel

**SQL> recover managed standby database disconnect;** -> this will take database out of recovery mode;

open alert file for both primary and standby database and primary db sql prompt.

**[oracle@gaurav ~]$:- cd /u01/app/oracle/rdbms/orcl/orcl/trace**

**[oracle@gaurav ~]$:- tail -f alert\_prod.log**

Now to check stby is working or not logswitch on primary db side

**PRIMARY DATABASE**

**SQL> alter system switch logfile;**

(see both sequences are updated in primary and standby database)

to make standby as primary

now create a datafile on primary side it must be automatically created on stby side and if not then do one logswitch even if not then your stby db is not made properly

**SQL>create tablespace test '/u01/app/oracle/oradata/prod/test01.dbf' size 5m;**

**SQL> alter system switch logfile;**

SUCCESSFULL

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*NOTE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DON’T DO ONLY FOR KNOWLEDGE

**TO MAKE STANDBY DATABASE AS PRIMARY DATABASE(STANDBY DATABASE WILL ACT AS**

**PRIMARY DATABASE)**

**STANDBY DATABASE**

**SQL>shut immediate**

**SQL>startup nomount**

**SQL>alter database mount standby database;**

**SQL>alter database activate standby database;**

**SQL>alter database open;**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*