RMAN RESTORE ON DIFFERENT SERVER

DESCRIPTION

This document explains how to restore the RMAN backup from tape to server.

ENV DETAILS:

Target Server : HKLVDPAPP071

Source Server : HKLPDSS2B002 (Standby of prod database where RMAN backup is running).

Backup Server : HKLVIPAPP021

DB Name : POHK2CCD

PRE-REQUESTIES

1. Make sure server is identified for restore with help of PSS team
2. Get the source server,backup server and RECOVER_POOL details from backup team.
3. Make sure the backup is successful using backup logs placed in source server under below path

   "/home/oracle/log/`echo $ORACLE_SID`_`date +%d%m%y%H%M%S`_Rman_hot_level_0.msglog"

4. Get the required storage to perform restore and make sure ORAFRA has sufficient space to perform RMAN activity.
5. Create Pfile for target database to perform RMAN restore.

PFILE

Important parameter to consider for restore is db_file_name_convert and log_file_name_convert. PFB parameters for reference. Also attached sample pfile used for restore.

*.db_file_name_convert='+DATA_CCD02' , '/oradata'

*.log_file_name_convert='+REDO1' , '/oraredo1' , '+REDO2', '/oraredo2'

rman_restore_pfile.txt

RESTORE

1. Make sure change is placed to perform this activity and respective CI added to change with sufficient time frame.

2. Below details will be shared by backup team once they load the backup as clone to Pool.

   NSR_SERVER=HKLVIPAPP021

   NSR_CLIENT=HKLPDSS2B002

   NSR_RECOVER_POOL="DC1 DB Clone DD11"

3. Login to target server using OV. If the database is already exists, please cleanup the DB before restoring. After dropping the DB, make sure space on ORADATA and ORAFRA is claimed.

   export ORACLE_SID=POHK2CCD

   sqlplus /as sysdba

   SQL> shu immediate

   SQL> startup mount exclusive restrict;

   SQL> drop database;

4. STARTUP NOMOUNT

   export ORACLE_SID=POHK2CCD

   SQL> startup nomount pfile='/u01/app/oracle/local/dbs/initPOHK2CCD.ora';

   Get the Control file backup piece details from backup log, which exists on source server

   /home/oracle/log/`echo $ORACLE_SID`_`date +%d%m%y%H%M%S`_Rman_hot_level_0.msglog

5. RESTORE CONTROL FILE

   export ORACLE_SID=POHK2CCD

   RMAN > Connect target /

   RMAN > run

   {

   allocate channel CH1 type 'SBT_TAPE';

   allocate channel CH2 type 'SBT_TAPE';

send 'NSR_ENV=(NSR_SERVER=HKLVIPAPP021,NSR_CLIENT=HKLPDSS2B002,NSR_RECOVER_POOL="DC1 DB Clone DD11",NSR_DEBUG_LEVEL=1,NSR_DPRINTF=TRUE,NSR_DIAGNOSTIC_DEST=/oradump/POHK2CCD_19c/nsr/apps/logs)';

restore primary controlfile from 'control_POHK2CCD_8c0rndje_234764_1_1'; -- Get piece details from backup log

release channel CH1;

release channel CH2;

}

RMAN > Exit;

Once control file restore is done, make sure database is in mount stage

6. RESTORE

   If the database size is huge, start the restore in nohup mode. Please find the Shell script for restore.

   

   RMAN_RESTORE_SH.txt

   export ORACLE_SID=POHK2CCD

   nohup sh RMAN_RESTORE_SH.sh >> RMAN_RESTORE_LOG.log &

7. Once restore and recover is done, follow below steps

   Add temp file to both PDB and CDB

   CREATE TEMPORARY TABLESPACE TEMP2 TEMPFILE '/oradata/pohk2ccd_dg/temp2_01.dbf' SIZE 100M AUTOEXTEND ON NEXT 1024M MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M;

   CREATE TEMPORARY TABLESPACE TEMP2 TEMPFILE '/oradata/pohk2ccd_dg/b186c9434c086277e0538ac7150a0ad8/temp2_01.dbf' SIZE 100M AUTOEXTEND ON NEXT 1024M MAXSIZE UNLIMITED EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M;

   ALTER DATABASE DEFAULT TEMPORARY TABLESPACE TEMP2;

   Change standby mode from maximum protection to maximum performance to open the database.

alter database set standby to maximize performance;

Open the database

SQL> alter database open;

Note:

If the backup is not consistent, we must do the incomplete recovery using until SCN or can be use hidden parameter (**only if oracle recommends**). Please follow below steps and open the database with resetlogs option

INCOMPLETE RECOVERY

Use below script to restore

RMAN_UNTIL_CLAUSE.txt

HIDDEN PARAMETER

After restoring, use the Hidden parameter to open the database. Steps to be follow

1. After we get error, shutdown the database and add _ALLOW_RESETLOGS_CORRUPTION = TRUE hidden parameter to pfile and startup the database till mount.
2. Procedd with open resetlogs
3. Once DB opened, check any files under v$recover_file. If nothing, shutdown the DB and remove hidden parameter from pfile.
4. Startup the database and do a sanity check.
5. Export the tables