CRONTAB SCHEDULER JOB

Automate RMAN Backup using Shell Script

In a real environment, you will not manually trigger all the Oracle database backups. You need an automated mechanism to trigger RMAN backups.

Create directory

On your database server, create a directory to hold RMAN backups and all related files. All the RMAN backups, logs and backup scripts are kept in one directory.

mkdir -p /u01/rman

Create RMAN backup script file

Let us create the RMAN backup script file to trigger DB FULL backup

vi /u01/rman/full_backup.sh

```
#!/bin/bash
. /home/oracle/.bash_profile
export ORACLE_SID=sha
export ORACLE_HOME=/u01/app/oracle/product/12.1.0.1
export DATE=$(date +%y-%m-%d_%H%M%S)
rman target / log=/u01/rman/proddb_${DATE}.log << EOF
run
{
allocate channel ch1 device type disk format '/u01/rman/proddb_full_bkp_%u';
allocate channel ch2 device type disk format '/u01/rman/proddb_full_bkp_%u';
crosscheck backup;
delete noprompt obsolete;
backup database;
backup archivelog all delete input;
release channel ch1;
release channel ch2;
}
EOF
```

Schedule Backup Under Crontab

Give execute permissions on the shell script

chmod 775 /u01/rman/full_backup.sh

Now you can go ahead and schedule the backup under the crontab. For example, we are scheduling backup to trigger at 2 am and 8 pm everyday

crontab -e

00 02,20 * * * /u01/rman/full_backup.sh

CRONTAB ENTRY for blocking session

##Give permission to file for execution.

*/30 * * * * /opt/app/oracle/admin/orcl/scripts/session_lock_orcl.sh >/opt/app/oracle/admin/orcl/scripts/orcl_lock_cron.log 2>&1

[oracle@server1 ~]$ cat /opt/app/oracle/admin/orcl/scripts/session_lock_orcl.sh

```
#!/bin/bash
export FILE_PATH=/opt/app/oracle/admin/orcl/scripts
cd $FILE_PATH
export ORACLE_SID=orcl
export ORACLE_HOME=/opt/app/oracle/product/12.2.0/db_1

$ORACLE_HOME/bin/sqlplus -S sys/oracle123@orcl as sysdba  <<\EOF > deadlock_alert_orcl.log
SET FEEDBACK OFF;
SET PAGES 200 LINES 200;
COL SESS FOR 9999;
COL serial# FOR 9999;
COL type FOR a3;
COL USERNAME FOR a10;
COL SCHEMANAME FOR a100;
COL status FOR a100;
COL OSUSER FOR a100;
SELECT DECODE(request, 0, 'Holder: ', 'Waiter: ') || l.sid sess,
      s.serial#,
      s.inst_id,
     a.SQL_FULLTEXT,
      l.type,
      s.USERNAME,
      s.SCHEMANAME,
      s.status,
```

```
    s.OSUSER,
    l.CTIME,
    l.id1,
    l.id2,
    l.lmode,
    l.request
 FROM gV$LOCK l, gV$session s,
    gv$sqlarea a
 WHERE (l.id1, l.id2, l.type) IN
    (SELECT l1.id1, l1.id2, type FROM gV$LOCK l1 WHERE request > 0)
  and l.sid = s.sid
  and l.type = 'TX'
  and l.inst_id=s.inst_id
 and s.inst_id=a.inst_id
  and a.SQL_ID=s.SQL_ID
 ORDER BY l.id1, l.request;

exit
EOF
cat deadlock_alert_orcl.log | grep -v "rows" > adeadlock_alert_orcl.log
sed '/-/d' deadlock_alert_orcl.log > final_deadlock_alert_orcl.log
awk '{if (++dup[$0] == 1) print $0;}' final_deadlock_alert_orcl.log > finaldeadlock_alert_orcl.log

MAIL_ALERT=`cat finaldeadlock_alert_orcl.log | wc -l`
if [ "$MAIL_ALERT" -gt "0" ]
then
##EMAIL_ID=`cat /opt/oracle/backup/scripts/mail_id_list`
EMAIL_ID=ansh.kumar@oracle.com
export EMAIL_ID
mailx -s "Lock Detected in orcl `date +\%d\%m\%Y\%R`" $EMAIL_ID < $FILE_PATH/deadlock_alert_orcl.log
fi
rm *deadlock_alert_orcl.log
```

Expdp Backup Using Shell Script

Configure a shell script, to take export backup of table daily at 3 PM and send the log to DBA.

Script: **(expdp_schema.sh)**

[oracle@prod_11g logical_bkp]$ cat expdp_schema.sh

```
export ORACLE_SID=db11
export ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/usr/lib
export PATH=$ORACLE_HOME/bin:$PATH:/usr/local/bin
export TIMESTAMP=`date +%a%d%b%Y`
export EXP_DIR=/export/home/oracle

echo =======
echo Export command
echo =======
echo $ORACLE_HOME
$ORACLE_HOME/bin/expdp \'/ as sysdba\' directory=test
dumpfile=expdp_tab_${TIMESTAMP}_%U.dmp logfile=expdp_log_${TIMESTAMP}.log
schemas=scott PARALLEL=1  COMPRESSION=ALL

echo SEND MAIL TO STAKE HOLDERS
echo =======
mailx -s "$ORACLE_SID $TIMESTAMP Export backup logfile" debkumar267@gmail.com <
$EXP_DIR/expdp_log_${TIMESTAMP}.log
echo Export completed at $TIMESTAMP
```

give 755 permission to expdp_schema.sh

[oracle@prod_11g logical_bkp]$ chmod 755 expdp_schema.sh

Now configure the script in crontab as below(pass the ORACLE_SID)

```
#00 15  * * * /u01/app/oracle/product/script/logical_bkp/expdp_schema.sh >>
/u01/app/oracle/product/script/logical_bkp/expdp_full.log  db11
```