Practice 11

Applying Patch Set Update (PSU)

Practice Overview

In this practice, you will apply Patch Set Update (PSU) number 26635815 (version 12.1.0.2.171017) on the RAC database stack, including the Grid Infrastructure home and Oracle Database home.

Note: Make sure to take a backup of the entire Virtualbox appliance files, including the shared disk files, before you start this practice. It is common to face issues when you apply PSUs.

Note: This patch includes updates for both the GI home and Database home and it can be applied in a rolling fashion.

Note: Bear in mind that this practice is a long practice. It can easily take longer than a couple of hours to implement it all.

Note: Do not consider this document as a reference to apply a PSU. Always read the Readme file that is shipped with the PSU installation file.

Practice Assumptions

- The practice assumes that you have the Oracle RAC database up and running in the virtual machines `srv1` and `srv2`.

- Downloaded Patch Set Update (PSU) number **26635815** (version 12.1.0.2.171017) for Linux 64 from Oracle Support site.
  The PSU file name is `p26635815_121020_Linux-x86-64.zip` and its size is 1.6 GB.

- Downloaded the latest OPatch version for 12.1 releases. At the time of this writing, the most recent OPatch version is 12.2.0.1.10. It could be downloaded from the following link.

  Following is the screenshot of the page when I downloaded the named OPatch version.

Preparing the RAC database for Applying the PSU

In this section of the practice, you will perform steps to prepare your RAC database stack for applying the PSU. You will extract the PSU installation files, upgrade the OPatch utility, and create testing client sessions.

Note: although you will use the OPatch utility that exists in the GI home, you will upgrade it in the database home as well. This might be handy for applying PSU in the future where you might use the OPatch utility that exists in the database home.

Preparing the PSU installation directory

1. Extract the PSU zip file locally in the hosting PC into the folder that you configured it as shared folder in `srv1` when you installed the software.

    Note: If you have lost that configuration, open the "**Settings**" of `srv1`, click on "**Shared Folders**" link in the right-hand pane. Add shared folder by pressing "**plus**" icon. Then select path to the location of the extracted PSU folder, and mark both boxes "**Read-only**" and "**Auto-mount**".

    This is better than copying the file straight to the virtual machine because it saves its disk space.

2. In the hosting PC, extract the downloaded OPatch installation zip file into the shared folder.

3. Open a Putty session and login as `root` to `srv1`.

4. Check the release of OPatch

    This PSU requires OPatch of version 12.1.0.1.7 or later.

    ```
    /u01/app/12.1.0/grid/OPatch/opatch version
    ```

5. Remove the existing OPatch folder from the GI home and Oracle database home.

    ```
    rm -fr /u01/app/12.1.0/grid/OPatch
    rm -fr /u01/app/oracle/product/12.1.0/db_1/OPatch
    ```

6. Chang the current directory to the directory `/media/sf_staging`. This is the shared folder directory.

    ```
    cd /media/sf_staging
    ```

7. Copy the OPatch directory to the GI home and database home directories.

    ```
    cp -R /media/sf_staging/OPatch /u01/app/12.1.0/grid/
    cp -R /media/sf_staging/OPatch /u01/app/oracle/product/12.1.0/db_1/
    ```

8. Change the ownership of the Opatch directory files to the home owner/group.

    ```
    chown -R grid:oinstall /u01/app/12.1.0/grid/OPatch
    chown -R oracle:oinstall /u01/app/oracle/product/12.1.0/db_1/OPatch
    ```

9. Verify the OPatch has been upgraded.

    ```
    /u01/app/12.1.0/grid/OPatch/opatch version
    ```

10. (Optional) The same way you upgraded Optach in `srv1`, upgrade the Optach in `srv2`. This is just in case you decide to use Opatch in `srv2` in the future.

```
ssh srv2
rm -fr /u01/app/12.1.0/grid/OPatch
rm -fr /u01/app/oracle/product/12.1.0/db_1/OPatch

cp -r /media/sf_staging/OPatch /u01/app/12.1.0/grid
cp -r /media/sf_staging/OPatch /u01/app/oracle/product/12.1.0/db_1

chown -R grid:oinstall /u01/app/12.1.0/grid/OPatch
chown -R oracle:oinstall /u01/app/oracle/product/12.1.0/db_1/OPatch

/u01/app/12.1.0/grid/OPatch/opatch version

exit
```

11. Validate the OPatch can retrieve the stack components and their versions.

If the following commands fail, stop at this step. You cannot proceed with applying the PSU and you need to receive Oracle support to look into the issue.

```
su - grid
$ORACLE_HOME/OPatch/opatch lsinventory -detail

su - oracle
$ORACLE_HOME/OPatch/opatch lsinventory
```

Take Backup of the RAC database

You do not want the backup job to run while you are applying the PSU. In the following steps, you will disable the backup job in `cron` and then take a backup of the entire database.

12. Run the following command and disable the backup job. You can disable any job by simply remarking the job line.

```
crontab –e
```

```
# * 0,4,8,12,16,20 * * *  "/home/oracle/scripts/rman_script.sh" > /dev/null
```

13. Run the following script to take a full backup of the RAC database.

```
/home/oracle/scripts/rman_script.sh
```

14. Verify the backup was successful by checking out the generated log file.

```
tail -n 50 ~/scripts/rman.log


rman target /
list backup of database;
```

Create Testing User sessions

In the following steps, you will create a service then make a group of client sessions connect to the database through this service. The target is to examine the response of the client sessions to applying the PSU on the RAC database.

15. Create a service named as `racsrv` where the basic TAF configuration is enabled in it. Start the service afterwards.

```
srvctl add service -db rac -service racsrv -preferred rac1 -available rac2 -
failovermethod BASIC -failovertype SELECT -failoverretry 10 -failoverdelay 5

srvctl start service -db rac -service racsrv

lsnrctl services | grep racsrv
```

16. In the `tnsnames.ora` file in your hosting PC, add the following connection descriptor. Do not copy it from the PDF file.

```
racsrv=
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.56.91)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = racsrv.localdomain)
    )
  )
```
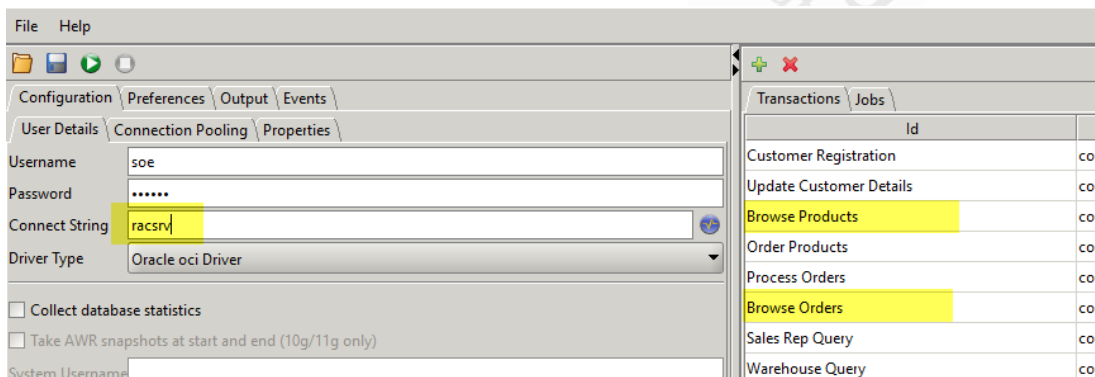
17. In the hosting PC, test the added connection descriptor.

```
sqlplus soe/soe@racsrv
```

18. Change the directory to the Swingbench directory and then run Swingbench.

```
cd D:\swingbench\winbin
swingbench.bat
```

19. Change the "**Connect String**" to `racsrv`. Select only the transactions "**Browse Products**" and "**Browse Orders**", as shown in the following screenshot.

20. Start the benchmark run and wait till the "Number of Transactions per Minute" saturates.

---

21. In the hosting PC, open a new command prompt window. Invoke SQL*Plus and make sure all the Swingbench sessions are connected to `rac1`.

```
sqlplus system/oracle@racsrv

SELECT INST_ID, COUNT(*)
FROM GV$SESSION
WHERE USERNAME='SOE' GROUP BY INST_ID;
```

Applying the PSU on the RAC database

In this section of the practice, you will apply the PSU first on `srv1` then on `srv2`. Meanwhile, you will observe how the client sessions respond to applying the PSU.

22. In the Putty terminal window, switch to the `root`, change the current directory to the PSU directory. In my case, I put the PSU extracted files in the sub-directory under the staging directory as shown in the following code.

```
su -
cd /media/sf_staging/psu12.1.0.2.171017/26635815

[root@srv1 26635815]# ls -l
total 77
drwxrwx---. 1 root vboxsf     0 Oct  2 10:27 21436941
drwxrwx---. 1 root vboxsf     0 Oct  2 10:27 26392164
drwxrwx---. 1 root vboxsf     0 Oct  2 10:27 26392192
drwxrwx---. 1 root vboxsf     0 Oct  2 10:27 26713565
drwxrwx---. 1 root vboxsf     0 Oct  2 10:27 automation
-rwxrwx---. 1 root vboxsf  4987 Oct  3 02:26 bundle.xml
-rwxrwx---. 1 root vboxsf 73591 Oct 16 07:47 README.html
-rwxrwx---. 1 root vboxsf     0 Oct  2 10:28 README.txt
```

23. (optional) Check for any patch conflicts.

In our case, no patch has ever been applied on our RAC database. Therefore, you can skip this step to save yourself some time.

```
export PATH=$PATH:/u01/app/12.1.0/grid/OPatch/
opatchauto apply -analyze
```

24. Apply the PSU using the `opatchauto` command.

Note: make sure no utility, such as SQL*Plus or `rman`, is running from ORACLE_HOME when you apply the PSU. Otherwise, the OPatch will fail because it will not be able to update some library files locked by those utilities.

```
export PATH=$PATH:/u01/app/12.1.0/grid/OPatch/
opatchauto apply
```

If Opatch fails, it stops with error and saves details about the error in a log file. If this happens to you, consider performing the following:

a) Look into the log file reported by the Opatch.

b) Fix the root cause of the reported issue.

c) Resume the Opatch using the following command:

```
opatchauto apply resume
```

25. Keep monitoring the output of the Opatch utility. When you see the message "Bringing down CRS service on home ...", re-run the query which monitors the Swingbench sessions. Some sessions should be failed over to `rac2`.

In my case, not all the sessions have been failed over. Some of them were disconnected. I believe this is because they were having some opened DML transactions.

You may observe the Swingbench GUI window is not responsive after its sessions got migrated to `rac2`. This is an application side issue. The sessions are still intact in `rac2`.

26. Wait till the PSU apply is finished. In my case, it took about 50 minutes to finish.

27. Verify that the patches have been applied in `srv1`.

```
su - grid
$ORACLE_HOME/OPatch/opatch lsinventory -detail

su - oracle
$ORACLE_HOME/OPatch/opatch lsinventory
```

28. Verify that the database is up and running in its nodes.

```
srvctl status database -d rac
```

29. Just to provide more free disk space in FRA diskgroup, invoke `rman` and delete the archivelog files.

```
rman target /
DELETE ARCHIVELOG ALL;
```

30. Open Putty, login to `srv2` as `root`, and repeat the same steps to apply the PSU.

When you see the message "Preparing to bring down database service...", observe that the Swingbench sessions have been relocated to `rac1`.

```
su -

cd /media/sf_staging/psu12.1.0.2.171017/26635815
export PATH=$PATH:/u01/app/12.1.0/grid/OPatch/
opatchauto apply
```

31. Verify that the patches have been applied in `srv2`.

```
su - grid
$ORACLE_HOME/OPatch/opatch lsinventory

su - oracle
$ORACLE_HOME/OPatch/opatch lsinventory
```

32. Exit from the Putty session connected to `srv2`.

Post-installation Procedure

33. Make sure the database is up and running.

```
srvctl status database -d rac
```

34. Delete the existing database `rac` backup and create a new backup.

```
# delete existing backup sets (just to avoid running out of free disk space)
rman target /
DELETE BACKUPSET;

# perform backup:
/home/oracle/scripts/rman_script.sh

# Verify the backup was successfully made
tail -n 50 ~/scripts/rman.log

# verify that backupset files have been put in their place:
rman target /
list backup of database;
```

35. Invoke SQL*Plus and login to `rac` database as `sysdba`.

```
sqlplus  / as sysdba
```

36. Run the following query to retrieve the patches applied on the database.
`DBA_REGISTRY_SQLPATCH` view does not get populated until the patch is applied in all the RAC nodes.

```
SELECT PATCH_ID,VERSION,ACTION,STATUS,DESCRIPTION
FROM DBA_REGISTRY_SQLPATCH;
```

37. Close Swingbench. If its GUI window is not responding, from the command prompt window that you used to invoke Swingbench, press on [Ctrl]+[c] to shut down the application.

38. Close the opened Putty sessions and the command prompt windows.

39. Shutdown the appliance machines.

40. Take backup of the Virtualbox appliance machine files including the shared disk files.

Summary

Normally, using Opatch, applying a PSU is an easy procedure. In Oracle RAC database, PSUs can be applied with almost no downtime using rolling fashion method.