

Supported Browsers

Oracle Cloud supports the following the minimum requirements for web browsers.

Web Browser	Version
Microsoft Internet Explorer	9 or 10 Notes: <ul style="list-style-type: none">Set Browser Mode to IE9 or IE10Set Document Mode to IE9 or IE10 standards
Google Chrome	29 and later
Mozilla Firefox	24 and later
Apple Safari	6

Known Issues

This section describes known issues associated with this release of Oracle Database Cloud Service.

Topics

- [Subset of actions are allowed with the Oracle Compute Cloud Service](#)
- [Attempts to create a deployment from a snapshot may fail](#)
- [Updating user credentials for backing up Oracle RAC deployments may return a false error](#)
- [Attempts to update the cloud tooling on an Oracle RAC deployment may fail](#)
- [After switchover on 11.2 Hybrid DR deployments, attempts to back up primary database may fail](#)
- [TDE wallet failure occurs during database readiness check when creating 11g or 12c Hybrid DR deployments](#)
- [Attempts to replace the database using the console may fail](#)
- [Attempts to create Oracle RAC deployments may fail](#)

- Instantiating from a cloud backup fails if the target DB name partially matches the source DB name
- Attempts to delete 12.2 database deployments may fail
- Attempts to scale up storage for Oracle RAC deployments may fail
- Failover and switchover on 18c Oracle RAC plus Data Guard deployments may fail
- Steps required to patch Spectre and Meltdown security vulnerabilities on existing instances
- Must use dgmgrl when primary is down on 11.2 Data Guard deployments
- Reinstate and switchover operations on 11.2 Oracle RAC plus Data Guard deployments may fail
- Reinstating a standby database on 11.2 Oracle RAC plus Data Guard deployments fails
- Updating the database password on 11.2 Oracle RAC plus Data Guard deployments may fail
- dbaascli dv {on|off} commands not supported on 12.2
- Cannot specify IP reservations when creating a deployment hosting a Data Guard Disaster Recovery configuration
- Enterprise Manager 11g Database Control not configured in new Oracle RAC database deployments
- Steps needed to stabilize Linux OS on existing and newly created deployments
- Rolling back Aug 2017 PSU on 11.2 invalidates database objects
- Application Express inaccessible after switchover or failover
- After Applying Aug 2017 12.2 RU, attempts to log into DBaaS Monitor fail
- Cannot use the console to update the credentials for backups to OCI Object Storage buckets
- Cannot use the console to update the password for cloud backups if it contains special characters
- Cannot specify IP reservations when creating a deployment hosting Database Clustering with RAC
- raccli update rdk command may change parameter values in the patching_properties file
- Console does not display backups and may not permit new backups after switchover or failover
- Manual backup configuration required after switchover or failover on Oracle RAC plus Data Guard deployments
- DEMOS PDB is not plugged-in in Oracle Database 12c Release 2
- Deleting a deployment and its backups can fail due to a timeout
- Application Express 5.0.0 and 5.0.4 not available after creating or replacing a Release 12c database using a cloud backup

- Older deployments use TLS 1.0
- Additional steps needed after applying Apr 2017 PSU to older deployments
- Attempts to add an SSH key to Oracle RAC deployments fail
- Backup pieces on cloud storage not deleted during deletion of database deployment
- Configure Backups fails to update credentials for backing up to cloud storage
- Updating the cloud tooling on a deployment hosting Oracle RAC requires manual update of the Oracle Database Cloud Backup Module
- Recovering an old backup causes database software and data to be out of sync
- Cloning a snapshot fails if snapshot's database was created or replaced using a cloud backup
- Configure Backups fails to update backup container password on Database Cloud Service instances hosting an Oracle RAC database
- Oracle Database Cloud Service console reports failed backups of Oracle RAC deployments as successful
- Cannot access EM 11g Database Control after Data Guard switchover
- Application Express, DBaaS Monitor and ORDS can become inaccessible
- Application Express, DBaaS Monitor and ORDS inaccessible after creating a 16.4.1 database deployment using a cloud backup of a 16.3.3 or 16.3.5 database deployment
- Application Express, DBaaS Monitor and ORDS inaccessible after creating a database deployment using a cloud backup
- Standard Edition deployment reports backup failure of Archivelogs Backup
- Creating a Java Cloud Service instance fails when a Database Cloud Service instance hosting an Oracle RAC database is specified
- Cannot access EM 11g Database Control for a Database Cloud Service instance hosting an Oracle RAC database
- Patching operation fails when database instance or virtual machine is restarted

Subset of actions are allowed with the Oracle Compute Cloud Service

When using Oracle Database Cloud Service, only limited use of the underlying Oracle Compute Cloud Service is supported.

The Compute Cloud Service Console can be used in the following ways:

- Using the Instance tile on the Overview page to view details about instances (virtual machines) and assign instances to network groups.
- Using the tiles on the Network page to manage network access to instances.
- Using the Security page to manage SSH keys

The Compute Cloud Service command line utilities cannot be used.

The Compute Cloud Service REST API cannot be used.

Attempts to create a deployment from a snapshot may fail

When you attempt to create a database deployment from a snapshot of another deployment, the create deployment operation may fail, reporting warnings about invalid parameters and the error `DB image not specified`.

Solution

To resolve this issue:

1. Update the cloud tooling on the database deployment from which the snapshot was made to version `18.2.3.1.0_180523.0000` or later by following the instructions *Updating the Cloud Tooling on Database Cloud Service* in *Administering Oracle Database Cloud Service*.
2. Create a new snapshot.
3. Create the database deployment from the new snapshot.

Updating user credentials for backing up Oracle RAC deployments may return a false error

When you use the Oracle Database Cloud Service console to update the user credentials for backing up a database deployment hosting Oracle Real Application Clusters (RAC), the operation may return a false error.

This problem occurs when you go to the Backup page, click **Configure Backups**, enter new user credentials, and click **Save**. The operation returns the following error: `Backup config operation on Oracle Database Cloud Service failed. This error is reported because the system cannot read the results file for the operation. Therefore, this error does not necessarily mean that the user credentials were not updated properly.`

This is a known limitation of Oracle Database Cloud Service, as of release 18.2.5.

Solution

To resolve this issue:

- For Database Clustering with RAC database deployments, perform the following steps on the RAC database.
- For Database Clustering with RAC and Data Guard Standby database deployments, perform the following steps on both RAC databases in your deployment, first on the primary RAC database, and then on the standby RAC database.

1. Perform the following steps on each of the two compute nodes hosting the RAC database:
 - a. Connect as the `opc` user to one of the compute nodes.
For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.
 - b. Start a root-user command shell:


```
$ sudo -s
#
```
 - c. Remove the results file:


```
# rm /tmp/bkupupdateconfig_dbname.out
```
 - d. Exit the root-user command shell and close your connection to the compute node:


```
# exit
$ exit
```
 - e. Repeat the preceding steps on the other compute node hosting the RAC database.
2. Connect as the `opc` user to either one of the compute nodes hosting the RAC database.
3. Update the server on both nodes:


```
$ raccli update server -tag 28039187 -allnodes
```
4. Disconnect from the compute node:


```
$ exit
```

After completing the preceding procedure, attempts to update the user credentials for backing up the database deployment should succeed.

Attempts to update the cloud tooling on an Oracle RAC deployment may fail

When you use the `raccli update rdk` command to update the cloud tooling to version 18.2.3 or later on a database deployment hosting Oracle Real Application Clusters (RAC), the update may fail and return the following error: `Failed to copy libopc.so`.

This error occurs under the following circumstances:

- For Database Clustering with RAC database deployments, the error occurs if automatic backups are not enabled for the RAC database.
- For Database Clustering with RAC and Data Guard Standby database deployments, the error occurs if automatic backups are enabled for the primary RAC database and disabled for the standby RAC database. This is the typical backup configuration for such deployments.

This is a known limitation of Oracle Database Cloud Service, as of release 18.2.3.

Workaround

To work around this issue:

- For Database Clustering with RAC database deployments, perform the following steps on compute node 1 for the RAC database.
- For Database Clustering with RAC and Data Guard Standby database deployments, perform the following steps on compute node 1 for the **standby** RAC database.

1. Connect as the `opc` user to the compute node.

For detailed instructions, see *Connecting to a Compute Node Through Secure Shell (SSH)* in *Administering Oracle Database Cloud Service*.

2. Verify that automatic backups are not enabled for the RAC database by running the following command:

```
$ raccli list backupconfig
```

- If the values for `diskEnabled` and `ossEnabled` are `false`, then automatic backups are not enabled for the RAC database and you have likely encountered this known issue. Continue to the next step.
 - Otherwise, you have encountered a different issue. Contact Oracle Support.
3. Copy and paste the results of the preceding command to a file, in case you want to restore the backup configuration to its original values after you have finished updating the cloud tooling.
 4. Enable automatic backups to an existing Oracle Cloud Infrastructure Object Storage Classic container by entering the following command.

Line breaks have been added for clarity; you must enter the command on a single line.

```
$ raccli update backupconfig -params '{"diskEnabled" : true, "ossEnabled" : true, "cloudStorageUser" : "username", "cloudStoragePwd" : "password", "cloudStorageContainerUrl" : "container-URL"}'
```

where:

- `username` is the user name of an Oracle Cloud user who has read/write access to the container.
- `password` is the password of the user specified for `cloudStorageUser`.
- `container-URL` is the URL of the Oracle Cloud Infrastructure Object Storage Classic container.

5. Attempt again to update the cloud tooling:

```
$ raccli update rdk -tag tag-number
```

6. (Optional) After the cloud tooling update is complete, restore the original automatic backup values that you recorded in step 3.

Enter the `raccli update backupconfig` command as shown in step 4, but specify the original values for each parameter.

After switchover on 11.2 Hybrid DR deployments, attempts to back up primary database may fail

When you perform a switchover on a Hybrid DR deployment, the cloud database becomes the primary database in the Oracle Data Guard configuration, and the on-premises database becomes the standby. After performing a switchover on a Hybrid DR deployment created using Oracle Database 11g Release 2, attempts to back up the cloud primary database may return Oracle Database error ORA-04062.

This is a known limitation of Oracle Database Cloud Service, as of release 18.2.3.

Solution

This issue occurs when the patch level of the on-premises standby database is lower than that of the cloud primary database. To resolve this issue:

1. Ensure that the on-premises database is up-to-date with patches. For more information, see "Patching a Hybrid DR Deployment" in *Administering Oracle Database Cloud Service*.
2. Attempt again to back up the cloud primary database.

TDE wallet failure occurs during database readiness check when creating 11g or 12c Hybrid DR deployments

The procedure for creating a Hybrid DR deployment includes running a readiness check script on the on-premises database that you intend to use in the Hybrid DR deployment.

Running the readiness check script (`setupdg.py`) on an Oracle Database 11g or Oracle Database 12c on-premises database may return the following error:

```
FAIL: TDE Wallet must be open in autologin mode
```

This is a known limitation of Oracle Database Cloud Service, as of release 18.2.3.

Workaround

To work around this issue:

1. On the on-premises database, open the following file in an editor:

```
/var/opt/oracle/hdg/setupdg.cfg
```

2. Change the value for `wallet_passwd` to the password for the `sys` user in the on-premises database.

After you do this, the values in the file for `wallet_passwd` and `sys_passwd` should be identical.

3. Save the file and exit from the editor.

4. Attempt again to run the `setupdg.py` script.

Attempts to replace the database using the console may fail

Attempts to use the Oracle Database Cloud Service console to replace the database on a deployment from a cloud backup may fail after 10 hours due to a timeout. This problem may occur if the cloud backup is very large or if a network problem occurs.

Solution

To resolve this issue, instead use `ibkup` actions of the `dbaasapi` utility to perform the replacement. For more information, see *Replacing the Database by Using ibkup Actions* in *Administering Oracle Database Cloud Service*.

Attempts to create Oracle RAC deployments may fail

When you create a database deployment and choose database type Database Clustering with RAC or database type Database Clustering with RAC and Data Guard Standby, the create deployment operation may fail.

The following error is returned:

```
Failed to configure RAC Oracle Database Server...[failed to create database db_unique_name, got exception failed to create database db_unique_name]
```

Solution

This is an intermittent issue. Attempt again to create the database deployment. If the second attempt also fails, continue trying to create the database deployment until the deployment is successfully created.

Instantiating from a cloud backup fails if the target DB name partially matches the source DB name

If you use the “instantiate from backup” feature to create a deployment using a cloud backup or to replace the database on an existing deployment using a cloud backup, the operation fails if the DB name on the deployment is a partial match of the DB name of the source database from which the cloud backup was taken.

The phrase “partial match” means that the target DB name is found within the source DB name. For example, the target DB name `ORCL` is a partial match of the source DB name `MYORCLDB`. However, the target DB name `MYDB` is not a partial match of the source DB name `MYORCLDB` because “MYORCLDB” does not contain the string “MYDB” as a single contiguous unit.

Solution

When creating the Database Cloud Service deployment, specify a DB name that is not a partial match of the DB name of the source database whose cloud backup you are going to use.

Attempts to delete 12.2 database deployments may fail

Attempts to delete Oracle Database Cloud Service database deployments hosting Oracle Database Release 12.2 may fail because the delete operation times out while waiting for backups to be removed from the Oracle Cloud Infrastructure Object Storage Classic container. This problem may occur if there are a large number of backups to remove, or if a network problem occurs.

Solution

To resolve this issue:

1. Update the database deployment to the latest cloud tooling. For more information, see "Updating the Cloud Tooling on Database Cloud Service" in *Administering Oracle Database Cloud Service*.
2. Delete the database deployment.

Attempts to scale up storage for Oracle RAC deployments may fail

Attempts to scale up data storage or backup storage for a database deployment hosting an Oracle Real Application Clusters (RAC) database may fail. This problem may occur if you specify less than 10 GB of additional storage for the scale up operation.

This is a known limitation of Oracle Database Cloud Service, as of release 18.1.3.

Workaround

To work around this issue:

1. Connect as the `opc` user to one of the compute nodes hosting the deployment.
For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell, and then switch to the `grid` user:

```
$ sudo -s
# su - grid
$
```

3. Invoke SQL*Plus and connect to the database as `SYSASM`:

```
$ sqlplus / as sysasm
```

4. Determine the disk group associated with the storage you are attempting to scale up. For data storage, it is disk group `DATA`. For backup storage, it is disk group `FRA`.

5. Determine the disk that failed during the scale up operation.
6. Find the name of the disk that failed. For example, to find the name of disk `'/dev/xvdh1'`:


```
SQL> select name, path from v$asm_disk where path='/dev/xvdh1';
```
7. Using the disk group name and disk name that you identified in the preceding steps, drop the disk from the disk group. For example:


```
SQL> alter diskgroup FRA drop disk FRA_0001;
```
8. Exit SQL*Plus, exit the `grid` user shell, exit the root-user shell, and disconnect from the compute node.


```
SQL> exit
$ exit
# exit
$ exit
```
9. Attempt again to scale up data storage or backup storage for the deployment, and specify at least 10 GB of additional storage when doing so.

Failover and switchover on 18c Oracle RAC plus Data Guard deployments may fail

A failover or switchover operation on a Database Clustering with RAC and Data Guard Standby database deployment hosting Oracle Database Release 18c may fail.

This problem may occur if the deployment uses the Oracle Active Data Guard option, and Oracle Active Data Guard is configured to use the In-Memory Column Store (IM column store), that is, the standby database has the following initialization parameter settings: `inmemory_size` is set to a value greater than 0 and `inmemory_adg_enabled` is set to `TRUE`.

This is a known limitation of Oracle Database Cloud Service, as of release 18.1.3.

Solution

To resolve this issue:

1. Connect as the `oracle` user to compute node 1 of the standby RAC database.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.
2. Log in to SQL*Plus as the `sys` user:


```
$ sqlplus / as sysdba
```
3. Run the following command to stop Redo Apply on the standby database:


```
SQL> alter database recover managed standby database cancel;
```
4. Run the following command to verify that the initialization parameter `inmemory_adg_enabled` is set to `TRUE`:

```
SQL> show parameter inmemory_adg_enabled
```

NAME	TYPE	VALUE
-----	-----	-----
inmemory_adg_enabled	boolean	TRUE

5. Run the following command to disable Oracle Active Data Guard for the IM column store on both instances of the standby RAC database:

```
SQL> alter system set inmemory_adg_enabled=FALSE sid='*' ;
```

6. Run the following command to start Redo Apply on the standby database in the background:

```
SQL> alter database recover managed standby database disconnect;
```

7. Wait two minutes for Redo Apply to start.
8. Exit SQL*Plus and disconnect from the compute node:

```
SQL> exit  
$ exit
```

9. Perform the failover or switchover operation.

Steps required to patch Spectre and Meltdown security vulnerabilities on existing instances

Oracle has released patches related to the Spectre (CVE-2017-5753, CVE-2017-5715) and Meltdown (CVE-2017-5754) security vulnerabilities.

Oracle strongly recommends that you apply these patches to existing Oracle Database Cloud Service database deployments created using release 18.1.2 or earlier. Deployments created using release 18.1.4 or later already contain the patches.

The steps you perform depend on the type of database deployment:

- [Steps for Single Instance Deployments](#)
- [Steps for Database Clustering with RAC Deployments](#)
- [Steps for Single Instance with Data Guard Standby Deployments](#)
- [Steps for Database Clustering with RAC and Data Guard Standby Deployments](#)
- [Steps for Data Guard Standby for Hybrid DR Deployments](#)

Steps for Single Instance Deployments

For deployments of single-instance databases, you apply the patches to the compute node hosting the deployment.

1. Connect as the `opc` user to the compute node.
For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.
2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Apply the patches by following the instructions in My Oracle Support Document ID 2351682.1, which can be accessed with the following URL: <https://support.oracle.com/epmos/faces/DocumentDisplay?id=2351682.1>.

Oracle recommends that you insert an additional step when following these instructions: After backing up the database, **shut down the database instance**, then proceed with applying the patches.

Steps for Database Clustering with RAC Deployments

For deployments of RAC databases, you apply the patches to the two compute nodes hosting the RAC database. You apply the patches first to one compute node and then the other, so the database remains accessible.

1. Perform the [Steps for Single Instance Deployments](#) on one of the compute nodes hosting the RAC database.
2. Perform the [Steps for Single Instance Deployments](#) on the other compute node hosting the RAC database.

Steps for Single Instance with Data Guard Standby Deployments

For deployments of Oracle Data Guard configurations where the primary and standby databases are single-instance databases, you apply the patches to the two compute nodes hosting the deployment. You use the Oracle Data Guard switchover operation, so the database remains accessible.

1. For the compute node associated with the **standby database**, perform the [Steps for Single Instance Deployments](#).
2. After the compute node and standby database have rebooted, allow a few minutes for redo data to be applied to the standby database.

To monitor the progress, connect as the `opc` user to the standby compute node, and run the following command:

```
$ dgmgrl show configuration verbose;
```

The operation is complete when the preceding command returns no errors.

3. Perform a switchover from the primary database to the standby database in your Oracle Data Guard configuration. See *Performing a Switchover Operation in Administering Oracle Database Cloud Service*.
4. For the compute node associated with the **new standby database** (this was the primary database before you performed the switchover), perform the [Steps for Single Instance Deployments](#).
5. After the compute node and new standby database have rebooted, allow a few minutes for redo data to be applied to the new standby database.

To monitor the progress, connect as the `opc` user to the new standby compute node, and run the following command:

```
$ dgmgrl show configuration verbose;
```

The operation is complete when the preceding command returns no errors.

6. Perform a switchover back to the original primary database in your Oracle Data Guard configuration. See *Performing a Switchover Operation in Administering Oracle Database Cloud Service*.

Steps for Database Clustering with RAC and Data Guard Standby Deployments

For deployments of Oracle Data Guard configurations where the primary and standby databases are RAC databases, you apply the patches to the four compute nodes hosting the deployment. You use the Oracle Data Guard switchover operation, so the database remains accessible.

1. For each of the two compute nodes associated with the **standby RAC database**, perform the [Steps for Single Instance Deployments](#).
2. After the compute nodes and standby RAC database have rebooted, allow a few minutes for redo data to be applied to the standby RAC database.

To monitor the progress, connect as the `opc` user to one of the standby RAC database compute nodes, and run the following command:

```
$ dgmgrl show configuration verbose;
```

The operation is complete when the preceding command returns no errors.

3. Perform a switchover from the primary RAC database to the standby RAC database in your Oracle Data Guard configuration. See *Performing a Switchover Operation in Administering Oracle Database Cloud Service*.
4. For each of the two compute nodes associated with the **new standby RAC database** (this was the primary RAC database before you performed the switchover), perform the [Steps for Single Instance Deployments](#).
5. After the compute nodes and new standby RAC database have rebooted, allow a few minutes for redo data to be applied to the new standby RAC database.

To monitor the progress, connect as the `opc` user to one of the new standby RAC database compute nodes, and run the following command:

```
$ dgmgrl show configuration verbose;
```

The operation is complete when the preceding command returns no errors.

6. Perform a switchover back to the original primary RAC database in your Oracle Data Guard configuration. See *Performing a Switchover Operation in Administering Oracle Database Cloud Service*.

Steps for Data Guard Standby for Hybrid DR Deployments

For Hybrid DR deployments, you apply the patches to the compute node for the cloud standby database by performing the [Steps for Single Instance Deployments](#).

For information on applying OS patches to the compute node for the on-premises primary database in a Hybrid DR deployment, refer to the documentation for the appropriate Oracle Database version.

Must use dgmgrl when primary is down on 11.2 Data Guard deployments

When the primary database is down and completely unreachable in a Database Cloud Service database deployment of Oracle Database Release 11.2 of type “Single Instance with Data Guard Standby” or “Database Clustering with RAC and Data Guard Standby”, attempts to use the Oracle Database Cloud Service console to perform Data Guard operations like failover fail after 120 minutes due to a timeout.

Solution

When the primary database is down, do not use the Oracle Database Cloud Service console to attempt Data Guard operations. Instead, connect to the healthy standby database and use the `dgmgrl` Data Guard command-line interface.

For example, to perform a failover operation to make the healthy standby database the primary database, you connect to a compute node of the standby database and then enter these commands:

```
# dgmgrl sys/<passwd>@<healthy_instance/standby target>  
DGMGRL> failover to <instance>
```

Reinstate and switchover operations on 11.2 Oracle RAC plus Data Guard deployments may fail

A reinstate or switchover operation on a Database Clustering with RAC and Data Guard Standby database deployment hosting Oracle Database Release 11.2 may fail if the initialization parameter `local_listener` is not set. This issue occurs because the DGMGRL static listener is defined on the local listener, and the Oracle Data Guard broker property `StaticConnectIdentifier` is defined on port 1521 of the scan listener.

This is a known limitation of Oracle Database Cloud Service, as of release 18.1.1.

Solution

To resolve this issue, define the DGMGRL static listener on the scan listener, instead of the local listener, for all compute nodes. Then, restart the local listeners and the scan listeners.

1. Perform the following steps on each of the four compute nodes hosting the Database Clustering with RAC and Data Guard Standby database deployment:
 - a. Connect as the `opc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\) in *Administering Oracle Database Cloud Service*](#).

- b. Start a root-user command shell, and then switch to the `grid` user:

```
$ sudo -s
# su - grid
$
```

- c. Open the following file in an editor:

```
/u01/app/12.2.0.1/grid/network/admin/listener.ora
```

- d. Locate the following entry in the file:

```
SID_LIST_LISTENER=
(SID_LIST=(SID_DESC=(GLOBAL_DBNAME=<dn_unique_name>_DGMGRL.<db_domain>)
(SID_NAME=<instance>)
(ORACLE_HOME=/u01/app/oracle/product/11.2.0.4/dbhome_1)))
```

- e. Modify the entry. The instructions for modifying the entry depend on which compute node you are currently connected to.

If you are connected to:	Modify the entry as follows:
The compute node hosting the first RAC node for the Oracle Data Guard primary database	Change <code>SID_LIST_LISTENER</code> to <code>SID_LIST_LISTENER_SCAN1</code> . See Example 1 .
The compute node hosting the second RAC node for the Oracle Data Guard primary database	Change <code>SID_LIST_LISTENER</code> to <code>SID_LIST_LISTENER_SCAN2</code> . See Example 2 .
The compute node hosting the first RAC node for the Oracle Data Guard standby database	Change <code>SID_LIST_LISTENER</code> to <code>SID_LIST_LISTENER_SCAN1</code> . See Example 1 .
The compute node hosting the second RAC node for the Oracle Data Guard standby database	Change <code>SID_LIST_LISTENER</code> to <code>SID_LIST_LISTENER_SCAN2</code> . See Example 2 .

Example 1

```
SID_LIST_LISTENER_SCAN1=
(SID_LIST=(SID_DESC=(GLOBAL_DBNAME=<dn_unique_name>_DGMGRL.<db_domain>)
(SID_NAME=<instance>)
(ORACLE_HOME=/u01/app/oracle/product/11.2.0.4/dbhome_1)))
```

Example 2

```
SID_LIST_LISTENER_SCAN2=
(SID_LIST=(SID_DESC=(GLOBAL_DBNAME=<dn_unique_name>_DGMGRL.<db_domain>)
(SID_NAME=<instance>)
(ORACLE_HOME=/u01/app/oracle/product/11.2.0.4/dbhome_1)))
```

- f. Save the file.

- g. Exit the `grid` user and root-user command shells, and disconnect from the compute node:


```
$ exit
# exit
$ exit
```
 - h. Repeat the preceding steps on the remaining compute nodes hosting the deployment.
2. Restart the local listeners and scan listeners on the primary and standby databases.
 - a. Perform the following steps on either one of the two compute nodes hosting the Oracle Data Guard *primary* database:
 - i. Connect as the `opc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\) in *Administering Oracle Database Cloud Service*](#).
 - ii. Start a root-user command shell, and then switch to the `grid` user:


```
$ sudo -s
# su - grid
$
```
 - iii. Resart the local listener and scan listener:


```
$ srvctl stop listener
$ srvctl start listener
$ srvctl stop scan_listener
$ srvctl start scan_listener
```
 - iv. Exit the `grid` user and root-user command shells, and disconnect from the compute node:


```
$ exit
# exit
$ exit
```
 - b. Repeat step a on either one of the two compute nodes hosting the Oracle Data Guard *standby* database.
 3. Perform the reinstate or switchover operation.

Reinstating a standby database on 11.2 Oracle RAC plus Data Guard deployments fails

If you attempt to reinstate a standby database on a Database Clustering with RAC and Data Guard Standby database deployment hosting Oracle Database Release 11.2, the operation returns Oracle Database error ORA-16653: failed to reinstate database. This error occurs only if the database you are attempting to reinstate was the standby database when the Data Guard configuration was initially configured.

This is a known limitation of Oracle Database Cloud Service, as of release 18.1.1.

Solution

To resolve this issue:

1. Connect as the `oracle` user to either one of the two compute nodes hosting the Oracle RAC standby database you are attempting to reinstate.

For detailed instructions, see *Connecting to a Compute Node Through Secure Shell (SSH)* in *Administering Oracle Database Cloud Service*.

2. Find the SCAN name by running the following command:

```
$ srvctl config scan
```

The SCAN name is listed in the output. For example:

```
SCAN name: myservice-scan-int, Network: 1
...
```

3. Invoke SQL*Plus and log in to the standby database as the `SYSTEM` user:

```
$ sqlplus system
Enter password: <enter the password for the SYSTEM user>
```

4. Run the following command to alter the `remote_listener` initialization parameter for both Oracle RAC instances. Set the parameter value to the SCAN name and port 1521, separated by a colon:

```
SQL> alter system set remote_listener='scan_name:1521' scope=both sid='*';
```

For example, using the sample output shown earlier in this procedure:

```
SQL> alter system set remote_listener='myservice-scan-int:1521' scope=both
sid='*';
```

5. Exit SQL*Plus and disconnect from the compute node:

```
SQL> exit
$ exit
```

6. Reinstate the standby database.

Updating the database password on 11.2 Oracle RAC plus Data Guard deployments may fail

Sometimes when you use the `raccli update databasepassword` command to update the database password on a Database Clustering with RAC and Data Guard Standby database deployment hosting Oracle Database Release 11.2, the operation fails. The failure includes the message “Local password is not valid. Please use 'raccli update databasepassword -saveonly' to save valid database password to local”.

Solution

After encountering the failure, perform the following steps:

1. Connect to each compute node of the primary database and check the timestamp of password file. On node 1 of the primary RAC database, the file is `$ORACLE_HOME/dbs/orapw<dbName>1` and on node 2 it is `$ORACLE_HOME/dbs/orapw<dbName>2`.

2. Compare the two timestamps to identify which password file is newer.
3. Copy the newer password file to the other compute node of the primary RAC database and to both compute nodes of the standby RAC database. After each copy operation, rename or delete the previous password file and rename the newly copied file to the name of the previous password file and set its file permissions to match those of the previous password file.

dbaascli dv {on|off} commands not supported on 12.2

The `dbaascli dv on` and `dbaascli dv off` commands are not supported on Oracle Database Cloud Service database deployments hosting Oracle Database Release 12.2. This is a known limitation of Oracle Database Cloud Service, as of release 17.4.1.

Cannot specify IP reservations when creating a deployment hosting a Data Guard Disaster Recovery configuration

When you create a database deployment that uses Oracle Data Guard and configure the standby database for disaster recovery, you cannot specify IP reservations (even though the Oracle Database Cloud Service console provides fields for IP reservations). Doing so will cause the deployment creation to fail.

This failure is a known limitation of Oracle Database Cloud Service, as of release 18.1.2.

Enterprise Manager 11g Database Control not configured in new Oracle RAC database deployments

When you create a database deployment specifying Oracle Database 11g Release 2 and a database type that includes Oracle RAC, the Enterprise Manager 11g Database Control web-based management tool is not configured.

Solution

To resolve this issue, you manually configure Enterprise Manager 11g Database Control:

1. As the `oracle` user on node 1 of the Oracle RAC deployment, create a `dbconsole.rsp` response file, as follows, based on your environment.

```
DB_UNIQUE_NAME=db-unique-name
SERVICE_NAME=db-unique-name.db-domain
PORT=scan-listener-port
LISTENER_OH=$GI_HOME
SYS_PWD=admin-password
DBSNMP_PWD=admin-password
SYSMAN_PWD=admin-password
```

```
CLUSTER_NAME=cluster-name
ASM_OH=$GI_HOME
ASM_SID+=ASM1
ASM_PORT=asm-listener-port
ASM_USER_NAME=ASMSNMP
ASM_USER_PWD=admin-password
```

To obtain the *cluster-name* value for your environment, run the command `$GI_HOME/bin/cemutlo -n`

2. Run the command to configure `dbcontrol` using the response file. The command will fail with an error listing commands you need to perform. You will perform these commands later.

```
$ $ORACLE_HOME/bin/emca -config dbcontrol db -repos create -cluster -silent \
-respFile dbconsole.rsp
```

Error securing Database Control. Database Control has not been brought-up on nodes *node1 node2*

Execute the following command(s) on nodes: *node1 node2*

1. Set the environment variable `ORACLE_UNQNAME` to the Database unique name.
2. `/u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emctl config emkey -repos -sysman_pwd <Password for SYSMAN user> -host node -sid <Database unique name>`
3. `/u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emctl secure dbconsole -sysman_pwd <Password for SYSMAN user> -host node -sid <Database unique name>`
4. `/u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emctl start dbconsole`

To secure Em Key, run `/u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emctl config emkey -remove_from_repos -sysman_pwd <Password for SYSMAN user>`

3. Edit the `$ORACLE_HOME/bin/emctl` file, changing the setting `CRS_HOME=` to `CRS_HOME=/u01/app/12.2.0.1/grid`.
4. Run the commands reported by `emca` in Step 2 with the proper values.
5. Configure `dbconsole` in node 2 so that the agent in node 1 reports to the `dbconsole` in node 1 and the agent in node 2 reports to the `dbconsole` in node 2:

```
$ $ORACLE_HOME/bin/emca -reconfig dbcontrol -silent -cluster \
-EM_NODE node1host -EM_NODE_LIST node2host -DB_UNIQUE_NAME db-unique-name \
-SERVICE_NAME db-unique-name.db-domain
```

6. Edit the `$ORACLE_HOME/bin/emctl` file, changing the setting `CRS_HOME=` to `CRS_HOME=/u01/app/12.2.0.1/grid`.
7. Check the `dbconsole` configuration status.

```
$ /u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emctl status agent
https://public-IP-for-Node1:1158/em
https://public-IP-for-Node2:1158/em
```

Steps needed to stabilize Linux OS on existing and newly created deployments

Configuration settings of certain Linux OS features on existing and newly created database deployments can cause noticeably slower I/O, system instability and, in certain extreme cases, system hang.

Oracle strongly recommends that you perform the following steps to alter these settings, both on the compute nodes of your existing database deployment and on the compute nodes of any database deployments you create.

The steps you perform depend on the type of database:

- [Steps for Single-Instance Databases](#)—Use these steps for deployments of single-instance databases and deployments of Oracle Data Guard configurations where the primary and standby databases are single-instance databases.
- [Steps for Oracle RAC Databases](#)—Use these steps for deployments of Oracle RAC databases and deployments of Oracle Data Guard configurations where the primary and standby databases are Oracle RAC databases.

Steps for Single-Instance Databases

For deployments of single-instance databases and deployments of Oracle Data Guard configurations where the primary and standby databases are single-instance databases, you download and run the latest `updt_blkfront.sh` script on each node of the deployment.

Note:

In performing these steps, you reboot the compute node and so cause the database to be inaccessible for a short period of time.

1. Connect as the `opc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Download the `updt_blkfront.sh` script:

```
# wget https://storage.us2.oraclecloud.com/v1/dbcswlibp-usoracle29538/
dbaas_patch/kernel_params/updt_blkfront.sh
```

4. Set the file permissions on the script to make it executable:

```
# chmod 0700 updt_blkfront.sh
```

5. Execute the script:

```
# ./updt_blkfront.sh
```

When prompted to execute the script, respond “Yes”.

6. Switch to the `oracle` user, shut down the database, and then return to being the `root` user.

```
# su - oracle
$ sqlplus '/ as sysdba'
...
SQL> shutdown immediate;
...
SQL> exit;
$ exit
#
```

7. Reboot the compute node:

```
# reboot
```

Your connection to the compute node is closed and the compute node reboots.

Steps for Oracle RAC Databases

For deployments of Oracle RAC databases and deployments of Oracle Data Guard configurations where the primary and standby databases are Oracle RAC databases, you use the `raccli` utility.

Note:

In performing these steps, you reboot each compute node of the RAC database in turn so that the database remains accessible.

1. Update the cloud tooling on the deployment to version 17.3.5.2 by following the instructions [Updating the Cloud Tooling by Using the raccli Utility](#) in *Administering Oracle Database Cloud Service*. When following these instructions, specify 17352 as the tag number.
2. Connect as the `opc` user to a compute node of the RAC database.
For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.
3. Run the following `raccli update server` command:

```
$ raccli update server -tag 17352
```

After the command completes, the compute node is rebooted.
4. After the compute node finishes rebooting, perform Steps 2 and 3 on the other compute node of the RAC database.

Rolling back Aug 2017 PSU on 11.2 invalidates database objects

After rolling back the August 2017 Patch Set Update (Aug 2017 PSU) on a database deployment hosting Oracle Database Release 11.2, some database objects might be marked `INVALID`. This is a known limitation of Oracle Database Cloud Service, as of release 17.3.6. This limitation is removed as of release 17.4.6.

Solution

To resolve this issue:

1. Connect as the `oracle` user to the compute node hosting the deployment.

For database deployments hosted on multiple compute nodes, connect to the compute node on which you initially applied the patch.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Check your database for invalid objects.
 - a. Invoke SQL*Plus and log in as the `sys` user with `sysdba` privileges.
 - b. Run the following query:

```
SQL> select count(*) from dba_objects where status='INVALID';
```

- If the query returns a count value of 0, the database does not have invalid objects. Exit SQL*Plus, disconnect from the compute node, and exit this procedure.
- Otherwise, continue to the next step.

3. If your deployment hosts an Oracle RAC database, run the following command:

```
SQL> alter system set cluster_database=FALSE scope=spfile;
```

4. Run the following commands:

```
SQL> shutdown immediate
SQL> startup upgrade
SQL> @?/sqlpatch/25434033/postinstall.sql
SQL> @?/rdbms/admin/utlrp.sql
```

5. If your deployment hosts an Oracle RAC database, run the following command:

```
SQL> alter system set cluster_database=TRUE scope=spfile;
```

6. Restart the database:

```
SQL> shutdown abort
SQL> startup
```

7. Verify that the database contains no invalid objects:

```
SQL> select count(*) from dba_objects where status='INVALID';
```

The query should return a count value of 0.

8. Exit SQL*Plus and disconnect from the compute node:

```
SQL> exit
$ exit
```

Application Express inaccessible after switchover or failover

After performing a switchover or failover operation on a database deployment that uses Oracle Data Guard, Application Express is inaccessible on the new primary instance. This situation persists even after you have enabled access to port 443 on the compute node for the primary instance.

Solution

To resolve this issue:

1. Connect as the `oracle` user to the compute node hosting the new primary instance.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Restart ORDS (Oracle REST Data Services):

```
$ /etc/init.d/ords restart
INFO: Stopping Oracle REST Data Services...
INFO: Oracle REST Data Services stopped
INFO: Starting Oracle REST Data Services...
INFO: Oracle REST Data Services bound to ports number, number, ...
INFO: Oracle REST Data Services started with PID number
```

3. On some database deployments, the preceding steps are sufficient to make Application Express accessible. Verify this by attempting to access Application Express on the new primary instance.

- If you can access Application Express, the issue is resolved. Disconnect from the compute node and exit this procedure.
- If you cannot access Application Express, continue to the next step of this procedure.

4. While still connected to the compute node as the `oracle` user, check the current DBaaS Toolset version:

```
$ rpm -q dbaastools
```

The preceding command returns a string of the form `dbaastools-1.0-1+nn.n.n.0.0_nnnnnn.nnnn.x86_64`, where the `n` characters represent numeric values. The string contains a release value (shown here in boldface type) between the plus sign (+) and underscore (_).

- If the release value is `17.4.3.0.0` or `17.4.5.0.0`, continue to the next step.
- Otherwise, contact Oracle Support.

5. Navigate to the directory where the ORDS configuration files are located:

```
$ cd /u01/app/oracle/product/ords
```

6. Repair the ORDS configuration:

```
$ hostname=$(hostname -f)
$ echo "db.hostname=$hostname" ords.properties
```

```
$ /u01/app/oracle/product/java/jdk1.8.0_74/bin/java -jar /u01/app/oracle/product/
ords/ords.war \
set-properties /u01/app/oracle/product/ords/ords.properties
$ rm -f ords.properties
```

7. Restart ORDS:

```
$ /etc/init.d/ords restart
INFO: Stopping Oracle REST Data Services...
INFO: Oracle REST Data Services stopped
INFO: Starting Oracle REST Data Services...
INFO: Oracle REST Data Services bound to ports number, number, ...
INFO: Oracle REST Data Services started with PID number
```

8. Verify that Application Express is accessible on the new primary instance.

After Applying Aug 2017 12.2 RU, attempts to log into DBaaS Monitor fail

After applying the August 2017 RU (release update) patch to a database deployment hosting Oracle Database Release 12.2, attempts to log into DBaaS Monitor may fail with an error.

Solution

To resolve this issue, you restart ORDS (Oracle REST Data Services):

1. Connect as the `opc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Restart ORDS:

```
# /etc/init.d/ords restart
INFO: Stopping Oracle REST Data Services...
INFO: Oracle REST Data Services stopped
INFO: Starting Oracle REST Data Services...
INFO: Oracle REST Data Services started with PID number
```

4. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

Cannot use the console to update the credentials for backups to OCI Object Storage buckets



This topic applies only to Oracle Cloud Infrastructure.

You cannot use the Oracle Database Cloud Service console to update the credentials (user name or password) for cloud backups to an Oracle Cloud Infrastructure Object Storage bucket. This is a known limitation of the console as of release 17.3.5.

Solution

To update the credentials:

1. Follow Steps 7 through 15 of the procedure Changing the Backup Configuration on Database Deployments Hosting Single-Instance Databases in *Administering Oracle Database Cloud Service*.
2. Edit the `/var/opt/oracle/creg/SID.ini` file, changing the label of the `oss_user` entry to `bkup_oss_user`.

Cannot use the console to update the password for cloud backups if it contains special characters

You cannot use the Oracle Database Cloud Service console to update the password for backups to cloud storage if the new password contains special characters like `$`, `&`, `>`, `)` and other punctuation. This is a known limitation of the console as of release 17.3.5.

Solution

To update the password:

1. Follow Steps 7 through 15 of the procedure Changing the Backup Configuration on Database Deployments Hosting Single-Instance Databases in *Administering Oracle Database Cloud Service*.
2. Edit the `/var/opt/oracle/creg/SID.ini` file, changing the label of the `oss_user` entry to `bkup_oss_user`.

Cannot specify IP reservations when creating a deployment hosting Database Clustering with RAC

When you create a database deployment and choose the Database Clustering with RAC database type, you cannot specify IP reservations for the two compute nodes. Doing so will cause the deployment creation to fail.

This failure is currently a known limitation of Oracle Database Cloud Service, as of release 17.3.5.

raccli update rdk command may change parameter values in the patching_properties file

The `raccli update rdk` command updates the cloud tooling on Oracle Database Cloud Service database deployments that use Oracle Real Application Clusters (RAC). Running this command on Oracle Cloud at Customer may erroneously change the source patching container values from internal OSS values to public OSS values in the `patching_properties` file. This will cause an error the next time you run a patching command.

Solution

After running the `raccli update rdk` command on Oracle Cloud at Customer, examine the `patching_properties` file and correct any erroneously modified values.

Here are the steps:

1. Connect as the `opc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user shell:

```
$ sudo -s
#
```

3. Open the following file:

```
/opt/oracle/dcs/rdbaas/config/patching_properties
```

4. Examine the values for the `URI` and `CONTAINER` parameters to determine if they have been changed to public OSS values.
 - If the `URI` value contains the host name `storage.us2.oraclecloud.com`, then it has been changed to a public OSS value.
 - If the `CONTAINER` value is `rdbaas_patches`, then it has been changed to a public OSS value.

The following are examples of public OSS values:

```
URI=https://storage.us2.oraclecloud.com/v1/rdbaascs-usoracle6789
CONTAINER=rdbaas_patches
```

5. If the `URI` and `CONTAINER` parameters have been changed to public OSS values, then edit the `patching_properties` file and substitute your internal OSS values.
 - Change the `URI` value to your internal OSS URI.
 - Change the `CONTAINER` value to `racdbaas_patching`.

Console does not display backups and may not permit new backups after switchover or failover

After performing a switchover or failover operation on a database deployment of type “Database Clustering with RAC and Data Guard Standby”, the Oracle Database Cloud Service console does not display backups taken before or after the operation. This

situation persists even after you have manually enabled backups on the new primary database.

Additionally, if you attempt to use the console to create a backup before you manually enable backups on the new primary database, the attempt fails and the **Backup Now** button becomes unavailable.

These are known limitations of the Oracle Database Cloud Service console, as of release 17.3.1.

Manual backup configuration required after switchover or failover on Oracle RAC plus Data Guard deployments

After performing a switchover or failover operation on a database deployment that uses both Oracle RAC and Oracle Data Guard, you must manually disable backups on the old primary database and manually enable them on the new primary database.

1. Connect as the `opc` user to the first compute node of the **old** primary database.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Display the current backup configuration:

```
$ raccli list backupconfig
```

3. Note down the values of the following settings:

- `diskEnabled`
- `ossEnabled`
- `cloudStorageUser`
- `cloudStorageContainerUrl`

4. Disable backups of the old primary database:

```
$ raccli update backupconfig -params '{"diskEnabled":false,"ossEnabled":false}'
```

5. Disconnect from the first compute node of the old primary database:

```
$ exit
```

6. Connect as the `opc` user to the first compute node of the **new** primary database.

7. Enable backups of the new primary database using information you noted down from the old primary database.

(In the following command, line breaks have been added for clarity. Do not include them when entering the command.)

```
$ raccli update backupconfig -params '{
"diskEnabled":saved-value,
"ossEnabled":saved-value,
"cloudStorageUser":"saved-value",
"cloudStoragePwd":"storage-user-password",
"cloudStorageContainerUrl":"saved-value"
}'
```

where *saved-value* are the values you noted down from the old primary database, and *storage-user-password* is the password of the `cloudStorageUser` user.

8. Disconnect from the first compute node of the new primary database:

```
$ exit
```

DEMOS PDB is not plugged-in in Oracle Database 12c Release 2

When you create a database deployment, you can optionally create the DEMOS pluggable database (PDB). In database deployments created using Oracle Database 12c Release 2, DEMOS PDB is not plugged in after you create the database deployment. The DEMOS PDB cannot be successfully plugged in due to the fact that it includes APEX.

Do not select **Include "Demos" PDB** when creating a database deployment using Oracle Database 12c Release 2.

This is a known limitation of Oracle Database Cloud Service , as of release 17.3.1.

Deleting a deployment and its backups can fail due to a timeout

When you delete a database deployment and specify the option to delete the backups associated with the database deployment, the delete operation will fail due to a timeout if the operation takes more than four hours.

To avoid this failure, delete the database deployment and then delete its backups.

Application Express 5.0.0 and 5.0.4 not available after creating or replacing a Release 12c database using a cloud backup

Beginning with release 17.2.5 in early June 2017, database deployments you create include Oracle Application Express 5.1. Therefore, if you use the “instantiate from backup” technique to create a database deployment using a cloud backup or replace the database in a newly created deployment using a cloud backup, the Application Express software and configuration required by a backed up database that uses Application Express 5.0.0 or 5.0.4 are not available.

Perform the following steps to download and configure the proper version of Application Express for the database that was “instantiated from backup”.

1. Connect as the `oracle` user to the database deployment's compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Check the version of Application Express expected by the database:

```

$ sqlplus / as sysdba
...
SQL> SELECT VERSION FROM DBA_REGISTRY WHERE COMP_ID = 'APEX';

VERSION
-----
5.0.0.00.31

```

3. On your computer, go to the Oracle Application Express 5.0 Archive page, <http://www.oracle.com/technetwork/developer-tools/apex/downloads/apex-5-archive-2606313.html>, and download the zip file for Oracle Application Express 5.0.4 or 5.0, as appropriate.
4. Transfer the zip file to the database deployment's computer node.
For instructions, see Copying Files to or from a Database Cloud Service Database Deployment in *Administering Oracle Database Cloud Service*.
5. While still connected as the `oracle` user to the database deployment's compute node, navigate to the directory when you transferred the zip file to.

6. Unzip the zip file:

```
$ unzip apex_version.zip -d /u01/app/oracle/product/apex/
```

where `version` is the version you downloaded and transferred.

7. Move the decompressed files to a directory corresponding to the version of Application Express:

- For Application Express 5.0.0.00.31:

```
$ mv /u01/app/oracle/product/apex/apex/ /u01/app/oracle/product/apex/5.0.0.00.31/
```

- For Application Express 5.0.4.00.12:

```
$ mv /u01/app/oracle/product/apex/apex/ /u01/app/oracle/product/apex/5.0.4.00.12/
```

8. Disconnect from the compute node.
9. Reconnect to the compute node as the `opc` user. Then, start a root-user command shell:

```
$ sudo -s
#
```

10. Run the ORDS assistant to configure the version of Application Express:

```
# cd /var/opt/oracle/ocde/assistants/ords
# ./ords -out="/var/opt/oracle/ocde/res/ords.out" -ords_action="configure_apex"
```

11. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

Older deployments use TLS 1.0

Database deployments created before the 17.2.3 release (early May 2017) were configured with TLS (Transport Layer Security) version 1.0. If your security

requirements demand you use TLS 1.2, you can upgrade such deployments to use TLS 1.2 by performing the following steps.

1. Update the cloud tooling on the database deployment's compute node by following the instructions in [Updating the Cloud Tooling by Using the dbpatchm Subcommand in *Administering Oracle Database Cloud Service*](#), but do not exit the root-user command shell or disconnect from the compute node.

2. While still connected to the compute node, run the `sslpatch.pl` script:

```
# perl /var/opt/oracle/misc/sslpatch.pl
```

This script creates the log file `/var/opt/oracle/log/misc/sslpatch/sslpatch.log`.

3. Exit the root-user command shell and disconnect from the compute node:

```
# exit  
$ exit
```

4. If you are upgrading a database deployment hosting a Data Guard configuration, repeat the preceding steps on the other compute node of the deployment.

Additional steps needed after applying Apr 2017 PSU to older deployments

After applying the April 2017 Patch Set Update (Apr 2017 PSU) to a database deployment created before early May 2017 (base image 17.2.3), you need to perform additional steps.

1. Update the cloud tooling on the database deployment's compute node to version 17.1.5.1 or later by following the instructions in [Updating the Cloud Tooling by Using the dbpatchm Subcommand in *Administering Oracle Database Cloud Service*](#).

2. Connect as the `opc` user to the database deployment's compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\) in *Administering Oracle Database Cloud Service*](#).

3. Start a root-user command shell:

```
$ sudo -s  
#
```

4. Execute the following script:

```
# /var/opt/oracle/patch/dst_postv28.pl
```

5. Exit the root-user command shell and disconnect from the compute node:

```
# exit  
$ exit
```

Attempts to add an SSH key to Oracle RAC deployments fail

Attempts to add an SSH key to a database deployment hosting an Oracle Real Application Clusters (RAC) database fail.

Solution

To resolve this problem, delete the `authorized_keys.bkp` file from **every computer node** of the deployment and then add the SSH key.

Here are the steps to delete the file from a single compute node.

1. Connect as the `opc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Delete the `authorized_keys.bkp` file:

```
# rm -rf /home/opc/.ssh/authorized_keys.bkp
```

4. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

Backup pieces on cloud storage not deleted during deletion of database deployment

When you delete a database deployment, you can specify the option to delete the backups associated with the database deployment. However, backups stored on cloud storage may not be completely deleted as expected.

If you are affected by this issue, then you must manually delete the backup pieces from cloud storage. See [Deleting Objects](#) for details.

Configure Backups fails to update credentials for backing up to cloud storage

Sometimes, attempts to use **Configure Backups** on the Backup page of the Oracle Database Cloud Service console on a Database Cloud Service database deployment hosting a single-instance database fail, reporting the error "Failed cloud storage credential reset for DBaaS service".

Solution

To resolve this issue, update the cloud tooling on the database deployment to version 16.4.5.1 at a minimum. For instructions, see [Updating the Cloud Tooling by Using the `dbpatchm` Subcommand](#) in *Administering Oracle Database Cloud Service*.

Updating the cloud tooling on a deployment hosting Oracle RAC requires manual update of the Oracle Database Cloud Backup Module

If you have used the `update rdk` subcommand of the `raccli` utility to update the cloud tooling to 16.4.5 or later on Database Cloud Service database deployments hosting Oracle Real Application Clusters (RAC), you must manually update the installer for the Oracle Database Cloud Backup Module before you use the `update backupconfig` subcommand.

You must perform this manual update on both RAC nodes.

Solution

Here are the steps to update the installer:

1. Go to the Oracle Database Cloud Backup Module page on OTN:

```
http://www.oracle.com/technetwork/database/availability/oracle-cloud-backup-2162729.html
```

2. Select Accept License Agreement and click All Supported Platforms to download the `opc_installer.zip` file.
3. Using a secure copy utility that supports key-based, passwordless authentication such as `scp`, copy the `opc_installer.zip` file into the `/scratch/oss` directory on the compute node.
4. Connect as the `opc` user to the database deployment's compute node.

For detailed instructions, see *Connecting to a Compute Node Through Secure Shell (SSH)* in *Administering Oracle Database Cloud Service*.

5. Start a root-user command shell:

```
$ sudo -s  
#
```

6. Unzip the `opc_installer.zip` file into the same directory (`/scratch/oss`) to create the `opc_installer.jar` file.

7. Exit the root-user command shell and disconnect from the compute node:

```
# exit  
$ exit
```

Recovering an old backup causes database software and data to be out of sync

If you recover a Database Cloud Service database deployment using a backup created at a lower database software version than the deployment is running, the software and data can be out of sync. This situation can arise, for example, if you

patch a database deployment and then recover using a backup created before the patch was applied.

This situation can cause other problems, such as failures if you attempt to roll back a patch.

Solution

To resolve this issue, you synchronize the software and data by manually running a script on the recovered deployment:

1. Connect as the `oracle` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Run the script:

- For Oracle Database 12c:

```
$ datapatch -verbose
```

- For Oracle Database 11g:

```
$ cd $OH/rdbms/admin
$ sqlplus / as sysdba
SQL> @@catbundle.sql psu apply
SQL> exit;
```

3. Disconnect from the compute node:

```
$ exit
```

Cloning a snapshot fails if snapshot's database was created or replaced using a cloud backup

Attempts to create a Database Cloud Service database deployment from a snapshot fail if the snapshot is of a database deployment whose database was created or replaced using the database from a cloud backup by using an "instantiate from backup" technique.

This failure is currently a known limitation of Oracle Database Cloud Service, as of release 16.3.5.

Configure Backups fails to update backup container password on Database Cloud Service instances hosting an Oracle RAC database

When you attempt to use **Configure Backups** on the Backup page of the Oracle Database Cloud Service console on a Database Cloud Service database deployment hosting an Oracle RAC database, the operation fails without notifying you of an error

and the password to access the Storage Cloud Service container used for backups is not updated.

Solution

Use the `raccli` utility to update the password. For instructions, see [Updating the Password by Using the raccli Utility in *Administering Oracle Database Cloud Service*](#).

Oracle Database Cloud Service console reports failed backups of Oracle RAC deployments as successful

The Oracle Database Cloud Service console Activity page and list of backups on the Backup page show that backups of database deployments hosting an Oracle RAC database were successful even if they failed.

This issue is currently a known limitation of Oracle Database Cloud Service, as of release 16.3.5.

Cannot access EM 11g Database Control after Data Guard switchover

After performing a switchover operation on a Database Cloud Service database deployment hosting an Oracle Database 11g Data Guard configuration, attempts to connect to Enterprise Manager 11g Database Control fail with the message “Unable to connect”.

Solution

1. Connect as the `oracle` user to the compute node of the new primary database.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\) in *Administering Oracle Database Cloud Service*](#).

2. Define the `ORACLE_UNQNAME` environment variable:

```
$ export ORACLE_UNQNAME=SID
```

where `SID` is the SID of the database.

3. Create an `emca` response file that contains these lines:

```
PORT=1521  
SID=ORCL  
SYS_PWD=password  
DBSNMP_PWD=password  
SYSMAN_PWD=password  
LISTENER_OH=oracle-home
```

where `password` is the password of the SYS user and `oracle-home` is the path of the Oracle home directory.

4. Change the ownership of the response file you created:

```
$ chown oracle:oinstall response-file
```

5. Reconfigure and stop Database Control:

```
$ emca -deconfig dbcontrol db -repos drop -silent -respFile response-file  
$ emca -config dbcontrol db -repos create -silent -respFile response-file  
$ emctl stop dbconsole
```

6. Check the status of `emkey`. When prompted, enter the password of the SYS.

```
$ emctl status emkey
```

7. If the check reports that `emkey` is not configured properly, configure it using this command:

```
$ emctl config emkey -repos -sysman_pwd "password"
```

where *password* is the password of the SYS user.

8. Secure and start Database Control:

```
$ emctl secure dbconsole -sysman_pwd "password"  
$ emctl start dbconsole
```

9. Disconnect from the compute node:

```
$ exit
```

Application Express, DBaaS Monitor and ORDS can become inaccessible

On some database deployments, Oracle Application Express, DBaaS Monitor and ORDS (Oracle REST Data Services) can become inaccessible.

If they were inaccessible after using the instantiate-from-backup technique to create 16.4.1 database deployment using a cloud backup of a 16.3.3 or 16.3.5 database deployment, see [Application Express, DBaaS Monitor and ORDS inaccessible after creating a 16.4.1 database deployment using a cloud backup of a 16.3.3 or 16.3.5 database deployment](#).

If they were inaccessible after using the instantiate-from-backup technique to create database deployment using a cloud backup, see [Application Express, DBaaS Monitor and ORDS inaccessible after creating a database deployment using a cloud backup](#).

For any other situation, use the following solution. Here are examples of other situations that can cause this problem:

- After a linked-clone database deployment is created from a snapshot
- After performing a switchover operation on a database deployment hosting an Oracle Data Guard configuration

Solution

 **Note:**

If you are applying this solution to a database deployment hosting a Data Guard configuration, perform the following steps **on both nodes**; that is, on the one hosting the primary database and on the one hosting the standby database.

To resolve this issue, you restart ORDS:

1. Connect as the `opbc` user to the compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Restart ORDS:

```
# /etc/init.d/ords restart
INFO: Stopping Oracle REST Data Services...
INFO: Oracle REST Data Services stopped
INFO: Starting Oracle REST Data Services...
INFO: Oracle REST Data Services started with PID number
```

4. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

5. If you are applying this solution to a database deployment hosting a Data Guard configuration, repeat the preceding steps on the other compute node of the deployment.

Application Express, DBaaS Monitor and ORDS inaccessible after creating a 16.4.1 database deployment using a cloud backup of a 16.3.3 or 16.3.5 database deployment

Oracle Application Express, DBaaS Monitor and ORDS (Oracle REST Data Services) are not accessible on Database Cloud Service 16.4.1 database deployments whose database is instantiated from a cloud backup of a Database Cloud Service 16.3.3 or 16.3.5 database deployment.

This issue arises because the instantiate-from-backup process does not correctly upgrade the version of ORDS on the new 16.4.1 database deployment. To resolve this issue, you manually upgrade ORDS to version 3.0.6.176.08.46 as described in the solution below.

Solution

Here are the steps to manually upgrade ORDS to version 3.0.6.176.08.46:

1. Connect as the `opc` user to the database deployment's compute node.
For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Stop the ORDS service:

```
# /etc/init.d/ords stop
```

4. Switch to the `oracle` user:

```
# su - oracle
```

5. Navigate to the `ords` installation directory and create an `ords.properties` file:

```
$ cd /u01/app/oracle/product/ords/
$ echo "bequeath.connect=true" > "./ords.properties"
```

6. Reinstall ORDS:

```
$ ../java/jdk1.8.0_74/bin/java -DuseOracleHome=true -jar ords.war setup basic --
schemaOnly --parameterFile ./ords.properties --silent
```

Messages during setup indicate that ORDS is being upgraded to version 3.0.6.176.08.46.

7. Remove the `ords.properties` file:

```
$ rm ords.properties
```

8. Verify the upgrade:

```
$ sqlplus / as sysdba
...
SQL> SELECT VERSION FROM ORDS_METADATA.ORDS_VERSION;
VERSION
-----
3.0.6.176.08.46
SQL> exit
```

9. Exit the `oracle` user session, navigate to the ORDS configuration assistant directory, and force a reconfiguration of ORDS:

```
$ exit
# cd /var/opt/oracle/ocde/assistants/ords/
# ./ords -out=/var/opt/oracle/ocde/res/ords.out -ords_action=reconfigure
```

10. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

Application Express, DBaaS Monitor and ORDS inaccessible after creating a database deployment using a cloud backup

Oracle Application Express, DBaaS Monitor and ORDS (Oracle REST Data Services) are not accessible on some Database Cloud Service database deployments whose database is instantiated from a cloud backup created using Oracle Database Backup Cloud Service (a technique called “instantiate from backup”).

To resolve this issue, download and run a patching script, as described in the solution below.

Note that after running this patching script, Application Express may remain inaccessible and the “Open Application Express Console” link in the Oracle Database Cloud Service console will not work.

Also note that even after restoring accessibility, DBaaS Monitor may incorrectly report the database as stopped when it is in fact open.

Solution

Here are the steps to download and run patching script:

1. Connect as the `opc` user to the database deployment’s compute node.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\)](#) in *Administering Oracle Database Cloud Service*.

2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Download the patching script:

```
# wget https://storage.us2.oraclecloud.com/v1/dbcswlibp-usoracle29538/
dbaas_patch/ibkp/bug-24322127.sh
```

4. Set the file permissions on the script to make it executable:

```
# chmod +x bug-24322127.sh
```

5. Execute the patching script:

```
# ./bug-24322127.sh database-password
```

where `database-password` is the password of the SYSTEM user of the source database from which the cloud backup was made.

6. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

Standard Edition deployment reports backup failure of Archivelogs Backup

When you connect to or view backup status of a Database Cloud Service database deployment hosting a Standard Edition database, you see a backup failure message that includes the text “Cannot complete the Archivelogs Backup to Cloud Storage” and the backup log includes a reference to error KBHS-01602, “backup piece *piece-name* is not encrypted”.

Solution

To resolve this issue, update the cloud tooling on the database deployment. For instructions, see *Updating the Cloud Tooling by Using the dbpatchm Subcommand in Administering Oracle Database Cloud Service*.

Creating a Java Cloud Service instance fails when a Database Cloud Service instance hosting an Oracle RAC database is specified

If you specify a Database Cloud Service instance hosting an Oracle RAC database as the database to use when creating a Java Cloud Service instance, the creation attempt fails, reporting Oracle Database error ORA-12514.

Workaround

To work around this issue, use the Server Control Utility to add a service to the database and enable the `ora_p2_db_listener` security rule of the Database Cloud Service instance.

1. Connect as the `opc` user to node 1 of the Database Cloud Service instance.
For detailed instructions, see *Connecting to a Compute Node Through Secure Shell (SSH) in Administering Oracle Database Cloud Service*.

2. Start a root-user command shell and then switch to the `oracle` user:

```
$ sudo -s
# su - oracle
$
```

3. Add a service to the Oracle RAC database using the Server Control Utility.

- For Oracle Database 12c:

```
$ srvctl add service -d database -s pdb.identity-domain.oraclecloud.internal \
-pdb pdb -q FALSE -e NONE -m NONE -w 0 -z 0 -r instance1,instance2
```

- For Oracle Database 11g:

```
$ srvctl add service -d database -s database.identity-  
domain.oraclecloud.internal \  
-q FALSE -e NONE -m NONE -w 0 -z 0 -r instance1,instance2
```

where:

- `database` is the unique database name; for example, `ORCL`.
- `pdb` is the name of the default PDB; for example, `PDB1`.
- `identity-domain` is the id of the identity domain; for example, `usexample1234`.
- `instance1` and `instance2` are the names of the two instances in the cluster; for example, `ORCL1` and `ORCL2`.

4. Start the newly added service:

```
$ srvctl start service -d database
```

where `database` is the unique database name.

5. Close your connection to the compute node.
6. Enable the `ora_p2_db_listener` security rule of the Database Cloud Service instance.

For detailed instructions, see [Enabling Access to a Compute Node Port in Administering Oracle Database Cloud Service](#).

Cannot access EM 11g Database Control for a Database Cloud Service instance hosting an Oracle RAC database

Even after enabling the `ora_p2_monitor_11g` security rule, you cannot access Enterprise Manager 11g Database Control using the **Open EM Console** link in the service console for a Database Cloud Service instance hosting an Oracle RAC database.

Workaround

To work around this issue, use the Enterprise Manager Configuration Assistant to replace the existing configuration with one that refers to a new service you create using the Server Control Utility.

1. Connect as the `opc` user to node 1 of the Database Cloud Service instance.

For detailed instructions, see [Connecting to a Compute Node Through Secure Shell \(SSH\) in Administering Oracle Database Cloud Service](#).

2. Start a root-user command shell and then switch to the `oracle` user:

```
$ sudo -s  
# su - oracle  
$
```

3. Remove any existing `dbcontrol` configuration:

```
$ /u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emca -deconfig dbcontrol db -  
repos drop -cluster
```


If this command fails, it means there was no previous configuration. You can continue with these instructions despite the failure.

4. Add a service to the Oracle RAC database using the Server Control Utility:

```
$ srvctl add service -d database -s dbconsole.identity-  
domain.oraclecloud.internal \  
-r instance1,instance2
```

where:

- **database** is the unique database name; for example, `orcl`.
- **identity-domain** is the id of the identity domain; for example, `usexample1234`.
- **instance1** and **instance2** are the names of the two instances in the cluster; for example, `orcl1` and `orcl2`.

5. Start the newly added service:

```
$ srvctl start service -d database -s service
```

where **database** is the unique database name and **service** is the service name you provided in the `srvctl add service` command; for example, `dbconsole.usexample1234.oraclecloud.internal`.

6. Create an `emca` response file to use later when configuring `dbcontrol`. The response file has this format:

```
DB_UNIQUE_NAME=dbunique-name  
SERVICE_NAME=service-name  
PORT=1521  
LISTENER_OH=/u01/app/12.1.0.2/grid  
SYS_PWD=adminpasswd  
DBSNMP_PWD=adminpasswd  
SYSMAN_PWD=adminpasswd  
CLUSTER_NAME=cluster-name  
ASM_OH=/u01/app/12.1.0.2/grid  
ASM_SID=+ASM1  
ASM_PORT=1521  
ASM_USER_NAME=ASMSNMP  
ASM_USER_PWD=adminpasswd
```

where:

- **dbunique-name** is the unique database name; for example, `orcl`.
- **service-name** is the service name you provided in the `srvctl add service` command; for example, `dbconsole.usexample1234.oraclecloud.internal`.
- **adminpasswd** is the administrator password provided when the Database Cloud Service instance was created.
- **cluster-name** is the name of the Database Cloud Service instance.

Here is an example response file for a service instance named `r11204` in the identity domain `usexample1234`, created using `orcl` as the database SID and `Pa55_WoRd` as the administrator password:

```
DB_UNIQUE_NAME=orcl  
SERVICE_NAME=dbconsole.usexample1234.oraclecloud.internal
```

```
PORT=1521
LISTENER_OH=/u01/app/12.1.0.2/grid
SYS_PWD=Pa55_WoRd
DBSNMP_PWD=Pa55_WoRd
SYSMAN_PWD=Pa55_WoRd
CLUSTER_NAME=r11204
ASM_OH=/u01/app/12.1.0.2/grid
ASM_SID=+ASM1
ASM_PORT=1521
ASM_USER_NAME=ASMSNMP
ASM_USER_PWD=Pa55_WoRd
```

7. Configure dbcontrol using the response file you created:

```
$ /u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emca -config dbcontrol db \
-repos create -cluster -silent -respFile response-file
```

where *response-file* is the fully qualified name of the response file you created; for example, /tmp/emca.rsp.

8. Propagate the configuration change to node 2 of the database:

```
$ /u01/app/oracle/product/11.2.0.4/dbhome_1/bin/emca -reconfig dbcontrol \
-silent -cluster -EM_NODE node2-name -EM_NODE_LIST node2-name \
-DB_UNIQUE_NAME dbunique-name -SERVICE_NAME service-name
```

where:

- *node2-name* is the name of node 2; for example, r112042.
- *dbunique-name* is the unique database name; for example, orcl.
- *service-name* is the service name you provided in the `srvctl add service` command; for example, dbconsole.usexample1234.oraclecloud.internal.

9. Close your connection to the compute node.

 **Note:**

As a security precaution, after you confirm access to Enterprise Manager 11g Database Control, you should delete the `emca` response file you created because it contains passwords in clear text.

Patching operation fails when database instance or virtual machine is restarted

When applying a patch to an Oracle Database Cloud Service instance, the operation fails if the database instance is manually shut down or restarted (shut down and then started up) or if the virtual machine hosting the service instance is rebooted.

Workaround

None. This failure is expected behavior. The Oracle Database Cloud Service tooling for patching service instances requires that the database instance be running throughout the patching operation. The patch tooling itself shuts down and then starts up the database instance as part of the patching operation.

After a patching failure of this sort, check the patch log file, `/var/opt/oracle/log/dbpatchm/dbpatchm.log` to determine when the patching operation failed:

- If the operation failed before the `config` phase, you can simply reapply the patch.
- If the operation failed during or after the `config` phase, you need to roll back the partially applied patch before you attempt to reapply it.

Deprecated Features/Commands

The following features or commands are deprecated or removed in this release of Oracle Database Cloud Service:

- Oracle DBaaS Monitor is being removed from Database Cloud Service database deployments whose database type is “Single Instance” or “Single Instance with Data Guard Standby”. Instead, Oracle SQL Developer Web will be installed.

SQL Developer Web is a new, browser-based application that incorporates features of both Oracle SQL Developer and Oracle DBaaS Monitor. To find out more about it, see *Using Oracle SQL Developer Web in Database Cloud Service* in *Administering Oracle Database Cloud Service*.

If you have an older database deployment, you should replace DBaaS Monitor with SQL Developer Web as soon as is feasible by updating your cloud tooling, as described in *Updating the Cloud Tooling on Database Cloud Service* in *Administering Oracle Database Cloud Service*.

Notice of Future Deprecations and Removals

The following features will be removed in a future release of Oracle Database Cloud Service:

- Beginning January 2019, you will no longer be able to choose software release **Oracle Database 11g Release 2** or **Oracle Database 12c Release 2** when creating a database deployment.

Existing deployments using these software releases will continue to operate and be fully supported through their Premier Support period or their Extended Support period (when applicable). At the end of the Premier Support period or the Extended Support period, these releases will no longer be supported in Oracle Database Cloud Service.

If you are using, or are planning to use, Oracle Database 11g Release 2 or Oracle Database 12c Release 2, Oracle recommends that you plan an upgrade to Oracle Database 12c Release 1 or Oracle Database 18c.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Oracle® Cloud Known Issues for Oracle Database Cloud Service,
E56337-45

Copyright © 2014, 2018, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.