ORACLE
Cloud Infrastructure

REST API & CLI for ExaCS

Level 300

Bal Sharma

May 2019

ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Objectives

After completing this lesson, you should be able to:

- Describe the API & CLI available for ExaCS operations
- Understand dbaascli and examples.
- Understand how to setup OCI CLI & examples.
- ExaCLI and examples.
- Dbaasapi use case and example
- Demo

Exadata Cloud Service CLIs/APIs - Command Line Utilities

dbaascli

- Supports a variety of life-cycle and administration operations- Database Patching, SW library Updates, Oracle Home maintenance, PDB operations, TDE Management etc

OCI CL

- Almost all of the operations which can be performed from console –Database System Launch, DB creation/deletion VCN and related resource operation, CPU scaling etc

exacli

- Used to execute specific cellcli commands from compute node to the Exadata Storage Servers that are associated with your ExaCS environment. Use case is for getting Storage Cell metrices and diagnostics info.

dbaasapi

- Manual Database operations, though recommended method is to use OCI CL or console for DB tasks such as DB creation & deletion.

bkup_api

- Supports Backup life cycle – Creating configuration, Changing configuration, Backup, restore operations

Dbaascli-Manage life cycle operation of databases

dbaascli is a command line interface for different tools to be used with Oracle Cloud DB. This command line interface supports logging, command history and autocomplete.

Note: You must specify the dbname as an additional parameter for all commands:

--dbname <dbname> - where <dbname> is the name of the instance that you want to work with.

A variety of life-cycle and administration operations are supported such as:

- Starting and stopping a database
- Starting and stopping the Oracle Net listener
- Viewing information about Oracle Homes
- Moving a database to another Oracle Home
- Deleting an unused Oracle Home
- Performing database configuration changes
- Managing Oracle Database software images
- Managing pluggable databases (PDBs)
- Performing database recovery
- Rotating the master encryption key

Updating Cloud Tooling on Exadata Cloud Service

Cloud tooling include the fixes for existing issues and new features so it is highly recommended to upgrade the cloud tooling once new version or release is available. This is important before trying Patching using dbaascli.
Note: When updating the cloud tooling on database deployments hosting a Data Guard configuration, you must perform the update on both nodes; that is, on the one hosting the primary database and on the one hosting the standby database.

Check your current version of cloud tooling

```
[root@xdprod-n53zg1 ~]# rpm -qa|grep -i dbaastools
dbaastools_exa-1.0-1+18.2.3.1.0_190415.1015.x86_64
```

Configuring Automatic Cloud Tooling Updates

```
#dbaascli patch tools auto enable
```

Check for cloud tooling updates

```
[root@xdprod-n53zg1 ~]# dbaascli patch tools list
```

Disabling Automatic Cloud Tooling Updates

```
# dbaascli patch tools auto disable
```

Upgrade to Latest Cloud Tooling

```
[root@xdprod-n53zg1 ~]# dbaascli patch tools apply --patchid LATEST
```

Repeat the same steps on another instance if it is data guard setup for dbaascli cloud tooling upgrade
Note: How to upgrade DBAAS Cloud Tooling using dbaascli (Doc ID 2350471.1)

Dbaascli- uses

To use the dbaascli utility:

- Connect to a compute node associated with the Exadata Cloud Service deployment.
- Commands using the dbhome, dbimage, cswlib, or orec subcommands must be run with root administrator privileges. In this case, first connect as the opc user and then start a root-user command shell by executing the sudo —s command.
- Otherwise, connect as the oracle user.
- Run the dbaascli utility using a command of the form:

dbaascli subcommand subcommand-options

Ref: https://docs.oracle.com/en/cloud/paas/exadata-cloud/csexa/dbaascli.html

```
[oracle@xdprod-n53zg1 ~]$ dbaascli
DBAAS CLI version 18.2.3.1.0
DBAAS>help
Help for dbaascli
```

```
[root@xdprod-n53zg1 ~]# dbaascli cswlib download
DBAAS CLI version 18.2.3.1.0
Executing command cswlib download
INFO: CSWLIB update latest db image bits
INFO: Log file is: /var/opt/oracle/log/misc/cswlib/cswlib_2019-04-10_21:01:29.
INFO: CSWLIB update_bits of latest 11204 succeded !
INFO: CSWLIB update_bits of latest 12102 succeded !
INFO: CSWLIB update_bits of latest 12201 succeded !
```

```
[root@xdprod-n53zg1 ~]# dbaascli cswlib list
DBAAS CLI version 18.2.3.1.0
Executing command cswlib list
############ List of Available BP #############
-APR2017 (For DB Versions  12201 12102 11204)
-JAN2018 (For DB Versions  12201 12102 11204)
-APR2018 (For DB Versions  12201 12102 11204)
-JUL2018 (For DB Versions  18000 12201 12102 11204)
-OCT2018 (For DB Versions  18000 12201 12102 11204)
-JAN2019 (For DB Versions  18000 12201 12102 11204)
-NCAPR2018 (For DB Versions  12201 12102)
-NCJAN2019 (For DB Versions  12201 12102)
```

DBAAS>

result of list

Very Long Text, press q to quitAvailable commands: cleandb enable cns dbdisable cns dbenable cns dbstatus cns disable cns enable cns getprop cns setprop cns status database bounce database changepassword database move database start database status database stop dataguard failover dataguard reinstate dataguard status .....

Dbaascli- uses

```
dbaascli database bounce --dbname dbname
# dbaascli cswlib list
#dbaascli cswlib download [--version software_version] [--bp software_bp]
dbaascli database changepassword --dbname dbname
dbaascli database move --dbname dbname --ohome oracle_home
dbaascli database start --dbname dbname
dbaascli database status --dbname dbname
dbaascli database stop --dbname dbname
dbaascli database update --dbname dbname --redosize redo_size [--groups num_groups] [--precheck]
dbaascli database update --dbname dbname --db_unique_name dbname_uniquename [--precheck]
dbaascli dbhome info
dbaascli dbhome purge
# dbaascli dbimage list
dbaascli listener bounce --dbname dbname
dbaascli listener start --dbname dbname
dbaascli listener status --dbname dbname
dbaascli listener stop --dbname dbname
dbaascli pdb checkdb --dbname dbname
dbaascli pdb checknode --node nodenum --dbname dbname
$ dbaascli pdb checkpdb --pdbname pdbname --dbname dbname
```

Use Case: dbaascli- Update Your Software Library to Include the non-CDB Images

By default, Exadata Cloud Service, creates an Oracle Database 12c (or later) database using the multitenant architecture with a container database (CDB) and pluggable databases (PDBs). Many existing databases, especially E-Business Suite implementations, use the non-CDB database architecture.

Use the following command to display software library updates available:

```
[root@xdprod-n53zg2 ~]# dbaascli patch tools list—Check for latest tool, if not latest update using apply clause.
root@xdprod-n53zg1 ~]# dbaascli cswlib list
DBAAS CLI version 18.2.3.1.0
Executing command cswlib list
############# List of Available BP ##############
-APR2017 (For DB Versions  12201 12102 11204)
-JAN2018 (For DB Versions  12201 12102 11204)
-APR2018 (For DB Versions  12201 12102 11204)
-JUL2018 (For DB Versions  18000 12201 12102 11204)
-OCT2018 (For DB Versions  18000 12201 12102 11204)
-JAN2019 (For DB Versions  18000 12201 12102 11204)
-NCAPR2018 (For DB Versions  12201 12102)
-NCJAN2019 (For DB Versions  12201 12102)
```

```
[root@xdprod-n53zg1 ~]# dbaascli patch tools apply --patchid LATEST
DBAAS CLI version 18.2.3.1.0
Executing command patch tools apply --patchid LATEST
Current tools version on xdprod-n53zg1: 18.2.3.1.0_190328.0930
Patchid to apply  LATEST
Updated dbaastools rpm to dbaastools_exa-1.0-1+18.2.3.1.0_190409.1120.x86_64
Current tools version on xdprod-n53zg2: 18.2.3.1.0_190328.0930
Patchid to apply  LATEST
Updated dbaastools rpm to dbaastools_exa-1.0-1+18.2.3.1.0_190409.1120.x86_64
[root@xdprod-n53zg1 ~]#
```

Upgrade the software library on your service with the required non-CDB image version and BP.

```
[root@xdprod-n53zg2 ~]# dbaascli cswlib download --version 12102 --bp JAN2019 --cdb no
DBAAS CLI version 18.2.3.1.0
Executing command cswlib download --version 12102 --bp JAN2019 --cdb no
INFO: CSWLIB update NCJAN2019 db image bits
INFO: Log file is: /var/opt/oracle/log/misc/cswlib/cswlib_2019-04-12_18:34:40.555932313408.log
INFO: CSWLIB update_bits of NCJAN2019 12102 succeded !
```

Use Case: dbaascli- Patching operation on databases

You can patch Oracle databases on ExaCS using dbaascli utility.

Database Patch list -

dbaascli patch db list --oh hostname:/u02/app/oracle/product/12.1.0/dbhome_2

Database Patch prereq

dbaascli patch db prereq --patchid <patchid> --instance1 hostname:<oracle_home>
dbaascli patch db prereq --patchid <patchid> --dbnames <dbname>
dbaascli patch db prereq --patchid <patchid> --dbnames <dbname> -alldbs
dbaascli patch db prereq --patchid <patchid> --dbnames=<dbname1,dbname2> -alldbs

Database Patch apply -

dbaascli patch db apply --patchid <patchid> --instance1 hostname:<oracle_home> --dbnames <dbname1,dbname2> --run_datasql 1
dbaascli patch db apply --patchid <patchid> --dbnames <dbname>
dbaascli patch db apply --patchid <patchid> --dbnames <dbname> -alldbs

Database patch Rollback -

dbaascli patch db switchback --patchid <patchid> --instance1 hostname:<oracle_home> --dbnames <dbname1,dbname2> --run_datasql 1
dbaascli patch db switchback --patchid <patchid> --dbnames <dbname> -alldbs
dbaascli patch db switchback --patchid <patchid> --dbnames <dbname1,dbname2>

Dbaascli-regdb-Registering On premise database in cloud

Dbaascli Provide API for customer to registering on-premise database or any manual database you create so that tooling can be used for subsequent activities like backup/patching etc. No downtime is required to use this utility if the database meets all the requirements.

Commands

dbaascli regdb prereqs [--dbname <dbname>]
dbaascli regdb begin [--dbname <dbname>]

Register Flow

1. Create a new database from Cloud UI with the same as the on Premise database that is getting migrated
2. Remove the Cloud database using RMAN
3. Copy/Setup the on premise database to the Cloud VM
4. Setup Wallets and configure encryption
5. Run the prechecks using dbaascli

Register database using dbaascli

Dbaascli- regdb example

Run the the prereqs

[root@xdprod-n53zg1 ~]# dbaascli regdb prereqs --dbname exadb

DBAAS CLI version 18.2.3.1.0

Executing command regdb prereqs

INFO: Logfile Location: /var/opt/oracle/log/exadb/regdb/regdb_2019-05-01_22:22:18.066352380150.log

INFO: Prereqs completed successfully

Assumptions:

Database from on-premise must be of same name as Cloud database created as well Database should be at the same patch level

Run the the Register

[root@xdprod-n53zg1 ~]# dbaascli regdb begin --dbname exadb

DBAAS CLI version 18.2.3.1.0

Executing command regdb begin

Logfile Location: /var/opt/oracle/log/exadb/regdb/regdb_2019-05-01_22:25:16.701997385957.log

Running prereqs

DBAAS CLI version 18.2.3.1.0

Executing command regdb prereqs

INFO: Logfile Location: /var/opt/oracle/log/exadb/regdb/regdb_2019-05-01_22:25:20.350358386168.log

INFO: Prereqs completed successfully

Prereqs completed

Running OCDE .. will take time ..

OCDE Completed successfully.

Database exadb registered as Cloud database

Ref: Migration to Exadata Cloud using Simple Data Guard Approach with Minimal Downtime (Doc ID 2386116.1)

OCI- CLI

- OCI CLI provides all feature access of OCI console through CLI.
- This tool facilitates automation when interacting with OCI services.
- OCI CLI needs to be configured with proper authentication credentials to communicate OCI service.
- In order to install OCI CLI you can either use Linux or Windows compute environments.

https://docs.cloud.oracle.com/iaas/tools/oci-cli/latest/oci_cli_docs/index.html

OCI CLI supports several operations for ExaCS such as

- To control Exadata I/O Resource Manager(IORM)---console or CLI
- To manage database deployment
- To manage compute node operations and to view service details
- Scaling etc..
- Adding Keys..
- Provisioning

OCI CL installation for accessing Rest API's

Steps to be followed:

- Open the Linux terminal.
- To run the installer script, run the following command:

```
bash -c "$(curl -L https://raw.githubusercontent.com/oracle/oci-cli/master/scripts/install/install.sh)"
```

- Choose the directory where you want to save the executables and scripts of OCI.
- If the internet connectivity is active, then oci-cli will start installing.

Note: The linux OS should have internet connectivity.

- It will prompt you to update your $PATH…. Select y
- After selecting the path, installation will become successful and you will see it in message.
- Then to check the oci-cli installation, navigate to the oci location and type 'oci –help'

Once CLI is downloaded and installed we need to configure it for your Oracle tenancy.

OCI CL Configuration steps for accessing cloud environment

Below command confirms that it is installed correctly. Now you need to configure the CLI.

[opc@baloem133 ~]$ oci --help
Usage: oci [OPTIONS] COMMAND [ARGS]...

If the above is displayed then the installation is successful.

After installation you should configure oci-cli to connect to your instance in tenancy. To configure your cli run the command “oci setup config” . You can accept defaults for config location and need to provide several other details as we will see in example.

You can find the details required in the web console .
For getting values for OCID of user you need to follow below steps to find:
After logging to OCI console->Identity-Users->Select the user-> copy value of OCID

Note: OCID is an unique identifier for all resources in your tenancy

Configuration continued

After running the “oci setup config” then it will ask you the location for your config file

After giving the location it will ask you to enter the “user ocid”. (user ocid is obtained from the console under Console->Identity->Users->Your User)

Then, it will prompt you to give a “tenancy ocid”.(tenancy ocid is obtained from the console under Administration-> Tenancy details)

Then, it will ask you the region (enter the region your admin has subscribed) (eg: us-ashburn-1)

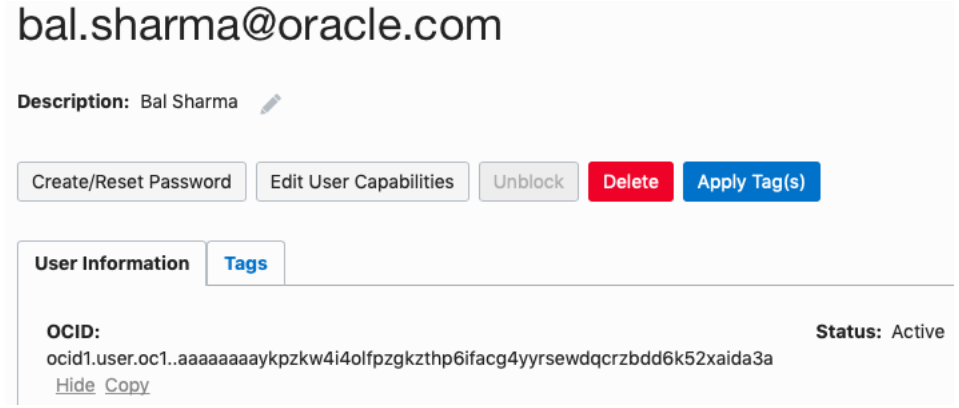
Then, it will ask for the generation of new RSA key.

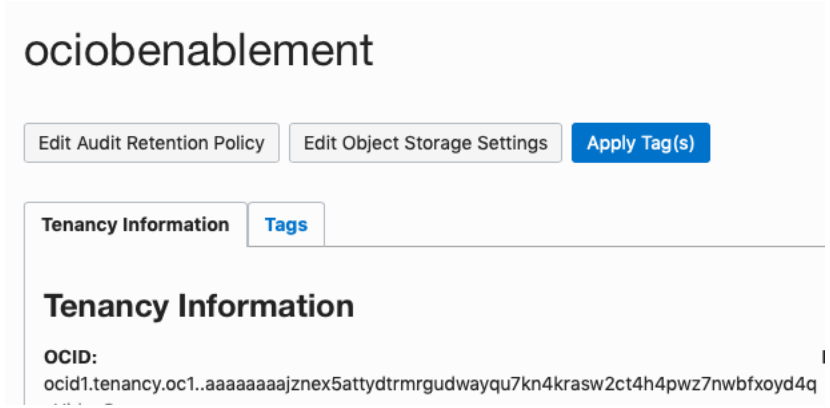
(If you don’t have the key, type ‘y’, so it will generate one for you) or (If you have one then type ‘n’ then it will prompt to give the location of your private key)

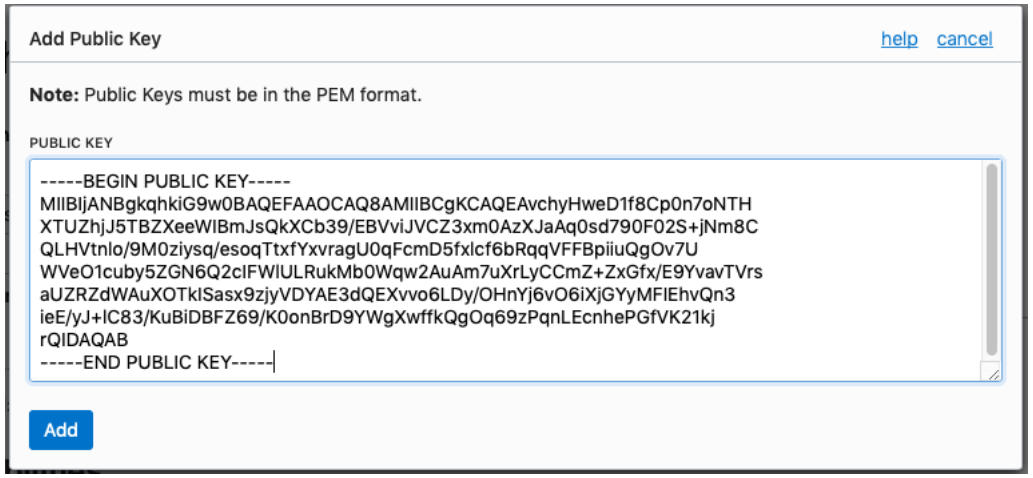
Note: After generating a new private key or after specifying the path of your private key you should add the corresponding public key in the console. For uploading the public key navigate to the users under Identity and then add public key, as shown in the images in example slide.

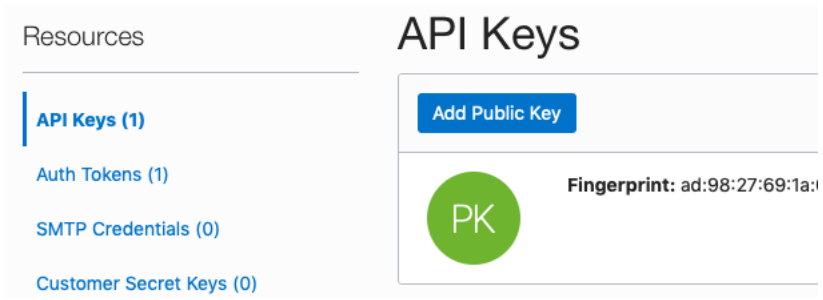
Once adding public key is done, the configuration would be complete and you will be connected to the tenancy.

Uploading API Keys









OCI CL Configuration Example

```
[[opc@baloem133 ~]$ oci setup config
    This command provides a walkthrough of creating a valid CLI config file.

    The following links explain where to find the information required by this
    script:

    User OCID and Tenancy OCID:

        https://docs.us-phoenix-1.oraclecloud.com/Content/API/Concepts/apisigningkey.htm#Other

    Region:

        https://docs.us-phoenix-1.oraclecloud.com/Content/General/Concepts/regions.htm

    General config documentation:

        https://docs.us-phoenix-1.oraclecloud.com/Content/API/Concepts/sdkconfig.htm


[Enter a location for your config [/home/opc/.oci/config]:
[Enter a user OCID: ocid1.user.oc1..aaaaaaaaykpzkw4i4olfpzgkzthp6ifacg4yyrsewdqcrzbdd6k52xaida3a
[Enter a tenancy OCID: ocid1.tenancy.oc1..aaaaaaaajznex5attydtrmrgudwayqu7kn4krasw2ct4h4pwz7nwbfxoyd4q
[Enter a region (e.g. ca-toronto-1, eu-frankfurt-1, uk-london-1, us-ashburn-1, us-phoenix-1): us-ashburn-1
[Do you want to generate a new RSA key pair? (If you decline you will be asked to supply the path to an existing key.) [Y/n]: Y
[Enter a directory for your keys to be created [/home/opc/.oci]:
[Enter a name for your key [oci_api_key]:
 Public key written to: /home/opc/.oci/oci_api_key_public.pem
[Enter a passphrase for your private key (empty for no passphrase):
 Private key written to: /home/opc/.oci/oci_api_key.pem
 Fingerprint: ad:98:27:69:1a:66:bf:b2:a5:bb:e2:d8:0e:4f:cf:2b
 Config written to /home/opc/.oci/config


    If you haven't already uploaded your public key through the console,
    follow the instructions on the page linked below in the section 'How to
    upload the public key':

        https://docs.us-phoenix-1.oraclecloud.com/Content/API/Concepts/apisigningkey.htm#How2
```

Copy Value of Public key to OCI console as show below

```
[opc@baloem133 ~]$ cat /home/opc/.oci/oci_api_key_public.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvchyHweD1f8Cp0n7oNTH
XTUZhjJ5TBZXeeWIBmJsQkXCb39/EBVviJVCZ3xm0AzXJaAq0sd790F02S+jNm8C
QLHVtnlo/9M0ziysq/esoqTtxfYxvragU0qFcmD5fxlcf6bRqqVFFBpiiuQgOv7U
WVeO1cuby5ZGN6Q2cIFWllULRukMb0Wqw2AuAm7uXrLyCCmZ+ZxGfx/E9YvavTVrs
aUZRZdWAuXOTkISasx9zjyVDYAE3dQEXvvo6LDy/OHnYj6vO6iXjGYyMFIEhvQn3
ieE/yJ+lC83/KuBiDBFZ69/K0onBrD9YWgXwffkQgOq69zPqnLEcnhePGfVK21kj
rQIDAQAB
-----END PUBLIC KEY-----
```

Navigate to Console-Identity-Users-Select User->API Keys-Add Public Key
Once adding public key is done, the configuration would be complete and you will be connected to the tenancy.

OCI CLI Commands for managing OCI environments

IAM Compartment operations

List all compartments available in tenancy

oci iam compartment list -c <root-compartment-id>

Create new compartment

oci iam compartment create --name <compartment_name> -c <root_compartment_id> --description "<friendly_description>"

Get a specified compartment

oci iam compartment get -c <root-compartment-id>

Update the specified compartment

oci iam compartment update -c <sub-compartment-id> --name=<name> --description=<description> --if-match <etag>

List all available domains within comparment

oci iam availability-domain list –c <compartment-id>

OCI CLI-IAM User

List three users in the tenancy
oci iam user list -c <root-compartment-id> --limit 3
oci iam user list -c <root-compartment-id> --limit 3 --page <opc-next-page>

List three groups user specified as member
oci iam user list-groups -c <root-compartment-id> --user-id <user_id> --limit 3 --page <opc-next-page>

Get oci user details
oci iam user get --user-id <user_id>

Create a new user in tenancy
oci iam user create -c <root-compartment-id> --name <user_name> --description "<desctiption>"

delete specific api signing keys
oci iam user api-key list <user-id>
oci iam user api-key delete --user-id <user-id> --fingerprint <fingerprint>

Create a new swift password for the specified user
oci iam user swift-password create --description <description> --user-id <user-id>

List swift password for specified user
oci iam user swift-password list --user-id <user-id

Update swift password description
oci iam user swift-password list --user-id <user-id>
oci iam user swift-password update --user-id <user-id> --swift-password-id <swift-password-id> --description <description> --if-match <etag>

Delete swiftpassword for specific user
oci iam user swift-password delete --user-id <user-id> --swift-password-id <swift-password-id> --if-match <etag>

Create new one time password for console user
oci iam user ui-password create-or-reset --user-id <user-id>

OCI CLI- database reference

Create a database
oci db database create --db-system-id <db_systems[0]> --db-version <DB_VERSION> --admin-password <ADMIN_PASSWORD> --db-name <random_db_name()>

Get a database
oci db database get --database-id < database_id>

List a database

oci db database list --compartment-id <COMPARTMENT_ID> --db-system-id <db_systems> --limit n

Delete a database
oci db database delete --database-id <database_id> --force

db.system-shape

List a db system shape
oci db system-shape list --availability-domain <availability_domain> --compartment-id <COMPARTMENT_ID>

db.node

List nodes in db-system
oci db node list --compartment-id <COMPARTMENT_ID> --db-system-id <db_systems[0]>

Get node
oci db node get --db-node-id <node_id>

DB node stop
oci db node stop --db-node-id <node_id>

DB node start
oci db node start --db-node-id <node_id>

DB node reset
oci db node reset --db-node-id <node_id>

DB node soft-reset
oci db node soft-reset --db-node-id <node_id>

Db system

Launch a db system
oci db system launch --generate-full-command-json-input

Terminate a db system
oci db system terminate --db-system-id <db_system_id> --force

Get a db system state
oci db system get --db-system-id <db_system_id>

List a db system
oci db system list --compartment-id <COMPARTMENT_ID>

Update a db system
oci db system update --db-system-id <db_systems[0]> --cpu-core-count <number of cpu core count> --ssh-authorized-keys-file <SSH_AUTHORIZED_KEYS_FILE> --force

List a db version
oci db version list --compartment-id <COMPARTMENT_ID>

OCI CL Commands example

Ref: https://docs.cloud.oracle.com/iaas/tools/oci-cli/latest/oci_cli_docs/cmdref/db.html

[opc@baloem133 ~]$ oci network vcn list -c
ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq


```
{
  "data": [
    {
      "cidr-block": "10.0.0.0/16",
      "compartment-id": "ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq",
      "default-dhcp-options-id": "ocid1.dhcpoptions.oc1.iad.aaaaaaaacziw4d46wjxaqj3367vxk5y3sp4zxkhtsklaws5ctmfgaswfuria",
      "default-route-table-id": "ocid1.routetable.oc1.iad.aaaaaaaacxzow447hyqpxkrcowtdz6ighgi5d56nq3b2j6nhs57btgndpdsq",
      "default-security-list-id": "ocid1.securitylist.oc1.iad.aaaaaaaake63vrnjyey6qn2u6txacfa4ygu2udez34phevjxrkyqayi3fdta",
      "defined-tags": {},
```


Additionally you can format the o/p by using –output table.

E.g

[opc@baloem133 ~]$ oci iam region list --output table

Using OCI CLI for ExaCS operations

Using the CLI, you can perform any of the OCI web console operations that you have privileges to access, modify, delete, and create. For example, you can list all the virtual cloud networks (VCNs) in a particular compartment.

Syntax: oci network vcn list --compartment-id

You can also use -c for --compartment-id if you so choose. You can get the compartment IDs or OCIDs from the OCI web console or with another CLI command:

oci iam compartment list

oci network vcn list -c
ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq

This CLI command lists all the VCNs in that compartment, but it also returns the OCID of the VCN. You can use that to list all the subnets in the VCN:

[opc@baloem133 ~]$ oci network subnet list -c
ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq --vcn-id
ocid1.vcn.oc1.iad.aaaaaaaad33asnpflhhgrnajnlhdntpkfosagmfvzcirdvlgct2aqljefqya

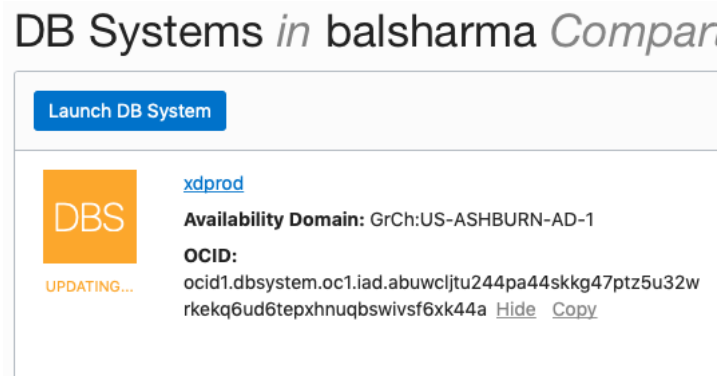
Using OCI CLI commands continued

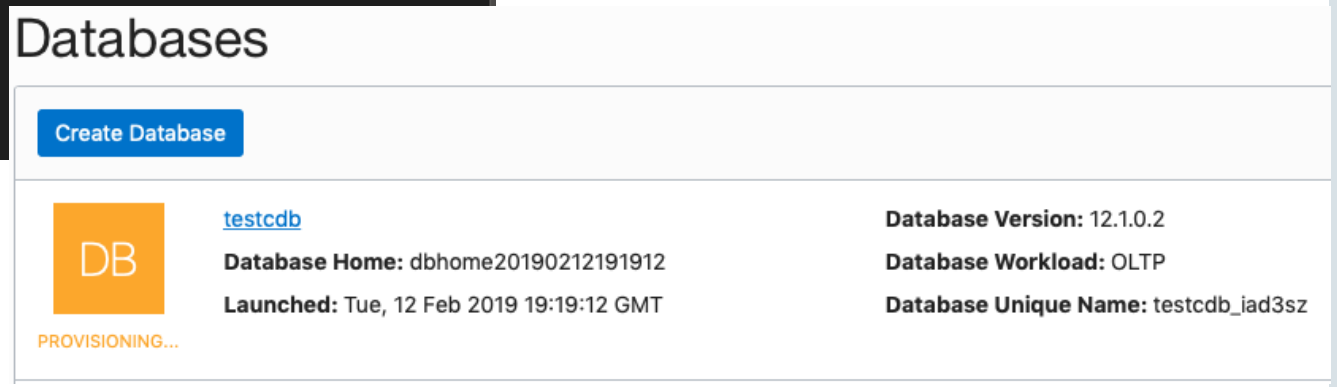
```
[opc@baloem133 ~]$ oci db database list --compartment-id
ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq  --db-system-id
ocid1.dbsystem.oc1.iad.abuwcljtu244pa44skkg47ptz5u32wrkekq6ud6tepxhnuqbswivsf6xk44a
```


```
{
  "data": [
    {
      "character-set": "AL32UTF8",
      "compartment-id": "ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq",
      "connection-strings": null,
      "db-backup-config": {
        "auto-backup-enabled": false
      },
      "db-home-id": "ocid1.dbhome.oc1.iad.abuwcljtashr4sn2whkxucgd4isyqt7rdakuy7yerwm42ebvrs6ewchrgq2q",
      "db-name": "bmsprod",
      "db-unique-name": "bmsprod_iad18z",
      "db-workload": "OLTP",
      "defined-tags": {},
    ----------o/p truncated----------
```

Creating & Delete Databases

```
[opc@baloem133 ~]$ oci db database create --db-system-id ocid1.dbsystem.oc1.iad.abuwcljtu244pa44skkg47ptz5u32wrkekq6ud6tepxhnuqbswivsf6xk44a
 --admin-password N0tAll0w##t --db-name testcdb --pdb-name testpdb --db-version 12.1.0.2
{
  "data": {
    "character-set": "AL32UTF8",
    "compartment-id": "ocid1.compartment.oc1..aaaaaaaaloae2yyzoslo4eybra5rqrswzdknotl5m63mpovn3n7ekf5v6wfq",
    "connection-strings": null,
    "db-backup-config": null,
    "db-home-id": "ocid1.dbhome.oc1.iad.abuwcljteaw4rltieb7eefsn2unecgv52o5hprae722td47jqjmoyslyajdq",
    "db-name": "testcdb",
    "db-unique-name": "testcdb_iad3sz",
    "db-workload": "OLTP",
    "defined-tags": {},
    "freeform-tags": {},
    "id": "ocid1.database.oc1.iad.abuwcljtxjchgv4mpm54nb4i6jk4uzsx6vmeba3vu2dk5eekwtbblyecajqa",
    "lifecycle-details": null,
    "lifecycle-state": "PROVISIONING",
    "ncharacter-set": "AL16UTF16",
    "pdb-name": "testpdb",
    "time-created": "2019-02-12T19:19:12.466000+00:00"
  }
}
```





[opc@baloem133 ~]$ oci db database create --db-system-id ocid1.dbsystem.oc1.iad.abuwcljtu244pa44skkg47ptz5u32wrkekq6ud6tepxhnuqbswivsf6xk44a --admin-password XXXX##t --db-name testcdb --pdb-name testpdb --db-version 12.1.0.2

[opc@baloem133 ~]$ oci db database delete --database-id ocid1.database.oc1.iad.abuwcljtxjchgv4mpm54nb4i6jk4uzsx6vmeba3vu2dk5eekwtbblyecajqa

Scaling OCPU in ExaCS

We can scale a service to the OCPU count we want to use. The command is in the following format:

```
[opc@baloem133 ~]$ oci db system update --cpu-core-count core_count --db-system-id dbsystemOCID
```

Here's an example of setting the core count of a quarter rack to 4 OCPUs:

```
[opc@baloem133 ~]$ oci db system update --cpu-core-count 4 --db-system-id ocid1.dbsystem.oc1.iad.abuwcljtu244pa44skkg47ptz5u32wrkekq6ud6tepxhnuqbswivsf6xk44a
```

```
[opc@baloem133 ~]$ oci db system get --db-system-id ocid1.dbsystem.oc1.iad.abuwcljtu244pa44skkg47ptz5u32wrkekq6ud6tepxhnuqbswivsf6xk44a
```

ExaCLI

ExaCLI is used to execute specific CellCLI commands, which perform monitoring and management functions on the Exadata Storage Servers that are associated with your ExaCS. You need cluster name in order to use.

```
[opc@xdprod-n53zg1 ~]$ sudo su - grid
[grid@xdprod-n53zg1 ~]$ crsctl get cluster name
CRS-6724: Current cluster name is 'xdpcluster-035'
To use exacli API use following:
[opc@xdprod-n53zg1 ~]$ exacli --login-name cloud_user_xdpcluster-035 --cookie-jar -c 192.168.136.4
No cookies found for cloud_user_xdpcluster-035@192.168.136.4.
Password: ***********
exacli cloud_user_xdpcluster-035@192.168.136.4>
E.g:
exacli cloud_user_xdpcluster-035@192.168.136.4> list griddisk detail
exacli cloud_user_xdpcluster-035@192.168.136.4> list celldisk
exacli cloud_user_xdpcluster-035@192.168.136.4> list flashcache
exacli cloud_user_xdpcluster-035@192.168.136.4> list IORMPLAN
exacli cloud_user_xdpcluster-035@192.168.136.4> list flashlog
exacli cloud_user_xdpcluster-035@192.168.136.4> list PLUGGABLEDATABASE
```

DBaaSAPI - ExaCS

Though Exadata DB systems include these command line tools for performing various tasks related to database , It is **highly recommended to use console** for performing activities, You might not see the database created using dbaasapi in the console.

The OCI CLI or Console should be used to create and remove databases in ExaCS environment. All calls to dbaasapi will be via an input json file, and the output will be also formatted in a json file.

/var/opt/oracle/dbaasapi/dbaasapi -i <json input file>

All calls will be asynchronous, except when the status is requested, which will be synchronous .

Common Parameters

object - Target object for current 'dbaasapi' process to be triggered like 'db'

operation - Operation to be performed like 'snapshot', 'clone', 'ibkp' etc.

action - Corresponding action for the operation specified like 'begin', 'end', 'delete' etc.

params - Nested JSON object for operation specific parameters

- **dbname** - DBNAME of the database
- **nodelist** - List of nodes (in case of RAC) for the current operation separated by comma or space. The meaning (install/start/re-locate) shall change based on operation/action performed

outfile - Absolute path to file write basic output payload of the current operation containing attributes like 'id', 'logfile' etc.

Flags - Additional flags like 'DEBUG'

Dbaasapi- Creating a Database

To get started we will create a directory called dbinput, a sample input file called myinput.json, and a sample output file called createdb.out.

SSH to a compute node in the Exadata DB system with opc user and sudo to root user.

Make a directory for the input file and change to the directory.

#mkdir –p /home/oracle/dbinput

cd /home/oracle/dbinput

- Create the input file in the directory. We will see an example will create a database configured to store backups in an existing bucket in Object Storage. For parameter descriptions, see Create Database Parameters.
- Run the utility and specify the input file.
- Check the output file and note the ID.
- Create a JSON file to check the database creation status. Note the action of "status". Replace the ID and the dbname with the values from the previous steps.
- Run the utility with the status file as input and then check the utility output.
- Rerun the status action regularly until the response indicates that the operation succeeded or failed.

dbaasapi-Create a non-CDB Database using non-CDB software image

Make sure you have updated cloud tooling before attempting this. This example need JAN2019 non-CDB software image to be downloaded to software library.

Make a directory for the input file and change to the directory.

```
# mkdir –p /home/oracle/dbinput
# cd /home/oracle/dbinput
```

#vi createdb.json. Step1

```
{
   "object": "db",
   "action": "start",
   "operation": "createdb",
   "params": {
      "nodelist": "",
      "cdb": "no",
      "bp": "JAN2019",
      "dbname": "nocdbbal",
      "edition": "EE_EP",
      "version": "12.1.0.2",
      "adminPassword": "WElcome#123_",
      "charset": "AL32UTF8",
      "ncharset": "AL16UTF16",
      "backupDestination": "OSS",
      "cloudStorageContainer": "https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/ociobenablement/balbucket",
      "cloudStorageUser": "bal.sharma@oracle.com",
      "cloudStoragePwd": "3DDKr+aEMHhFF61jDnrs"
   },
   "outputfile": "/home/oracle/dbinput/createdb.out",
   "FLAGS": ""
}
```

Be sure to set the cdb parameter to no and the bp parameter to the value that you used to download the non-CDB image for this. For backups set value for OSS.

MOS note for non CDB Creating non-CDB databases using Oracle Database 12c on the Exadata Cloud Service 2528257.1

Step2

```
[root@xdprod-n53zg1 dbinput]# cat /home/oracle/dbinput/createdb.out

{
   "msg" : "For security please remove your input file.",
   "object" : "db",
   "status" : "Starting",
   "errmsg" : "",
   "outputfile" : "/home/oracle/dbinput/createdb.out",
   "pid" : "",
   "action" : "start",
   "id" : "43",
   "operation" : "createdb",
   "logfile" : "/var/opt/oracle/log/nocdbbal/dbaasapi/db/createdb/43.log"
}
```

Example Continued..

Monitor status of the create process by Creating another json file containing id from previous o/p and name of database

Step3: vi createdbStatus.json

```
[root@xdprod-n53zg1 dbinput]# cat createdbStatus.json
{
  "object": "db",
  "action": "status",
  "operation": "createdb",
  "id": 43,
  "params": {
  "dbname": "nocdbbal"
  },
"outputfile": "/home/oracle/dbinput/createdbStatus.out",
"FLAGS": ""
}
[root@xdprod-n53zg1 dbinput]#
```

Step4: Track Progress

```
[root@xdprod-n53zg1 dbinput]# /var/opt/oracle/dbaasapi/dbaasapi -i createdbStatus.json
[root@xdprod-n53zg1 dbinput]# cat /home/oracle/dbinput/createdbStatus.out

{
   "msg" : "Running ocde",
   "object" : "db",
   "status" : "InProgress",
   "errmsg" : "",
   "outputfile" : "/home/oracle/dbinput/createdb.out",
   "pid" : "199958",
   "action" : "start",
   "id" : "43",
   "operation" : "createdb"
}
```

Rerun the status action regularly until the response indicates that the operation succeeded or failed.

You can also monitor the logfile identified in createdb.out to follow the progress:

tail -10f /var/opt/oracle/log/nocdbbal/dbaasapi/db/createdb/43.log

Bkup_api

- Backup setup is done via bkup tool which is part of the dbaas tool rpm.You can check the dbaas tool rpm version using - rpm -qa | grep dbaas.
- With the recent version of the rpm , backup entries for the database for which the backup is configured will be configured on all the nodes in /etc/crontab.
  Backup Channels are load balanced across the available nodes via the <dbname> connect string which should be configured with scan
- There are two backup entries for each DB. One for L0/L1 and one for archivelogs.
- Backup can be configured with one of the following backup destinations

Both Cloud Storage and Local Storage

Cloud Storage Only

None

https://docs.cloud.oracle.com/iaas/Content/Database/Tasks/exabackingup.htm

Demo

Summary

At the end of this session you should have understanding of

- Rest API available for ExaCS operations
- dbaascli and operations supported
- Understand operations performed through OCI CLI
- Understanding of ExaCLI.
- dbaasapi and operations supported
- Backup API overview

ORACLE®
Cloud Infrastructure

cloud.oracle.com/iaas

cloud.oracle.com/tryit