OCI Reliability and Resilience

Hands on Lab

ORACLE

Table of Contents

1. Disclaimer

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.

In the document, value of information such as user name, password etc. is shown as XXXX or YYYY etc. Also if actual value is shown consider it as an example value.

2. Introduction

Background

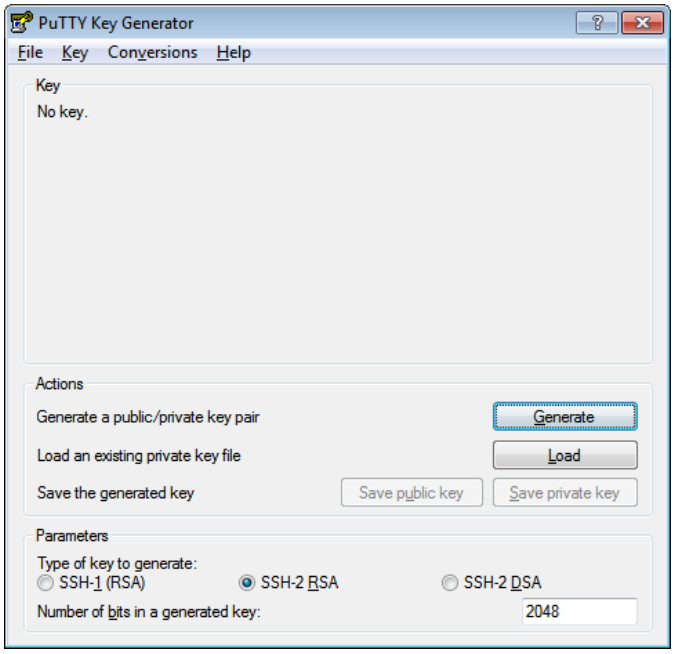
Oracle Cloud Infrastructure (OCI) is a set of complementary cloud services that enable you to build and run a wide range of applications and services in a highly available hosted environment. Oracle Cloud Infrastructure offers high-performance compute capabilities (as physical hardware instances) and storage capacity in a flexible overlay virtual network that is securely accessible from your on-premises network.

3. Pre-Requisites

- To perform the steps in this tutorial, you must have an active subscription to Oracle Cloud Infrastructure or a Free Trial Account.
- PuTTY and PuTTYGen installed

To generate an SSH key pair using the PuTTY Key Generator,

- Find puttygen.exe in the PuTTY folder on your computer, for example, C:\Program Files (x86)\PuTTY. Double-click puttygen.exe to open it. Or you may download it from here.



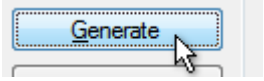
- Accept the default key type, **SSH-2 RSA**.

SSH-2 is the most recent version of the SSH protocol (and is incompatible with SSH-1). **RSA** and DSA are algorithms for computing digital signatures.

- Set the **Number of bits in a generated key** to 2048 bits, if it is not already set with that value.

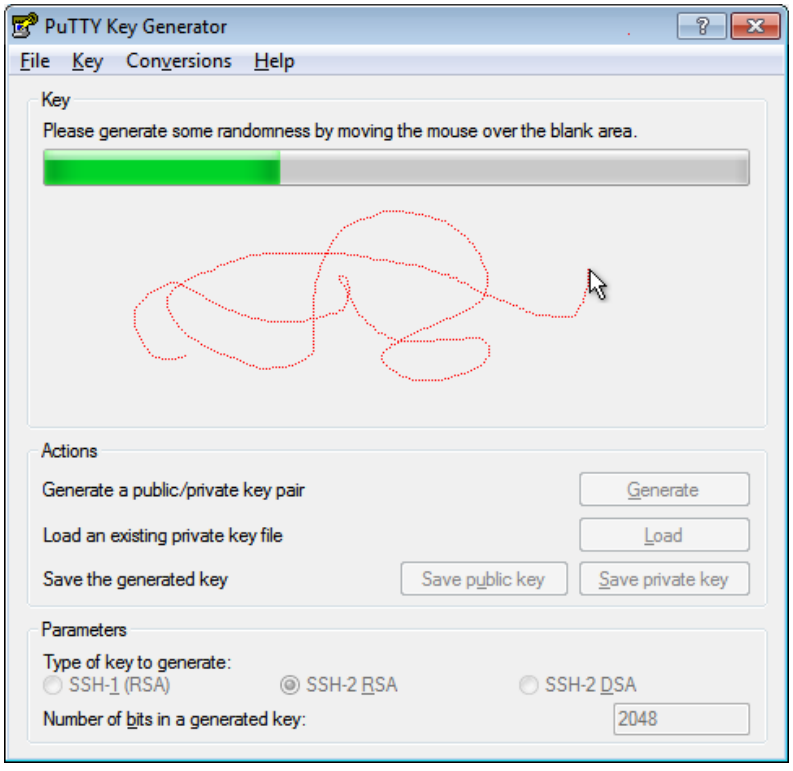
This sets the size of your key and thus the security level. A minimum of 2048 bits is recommended for SSH-2 RSA.

- Click Generate.

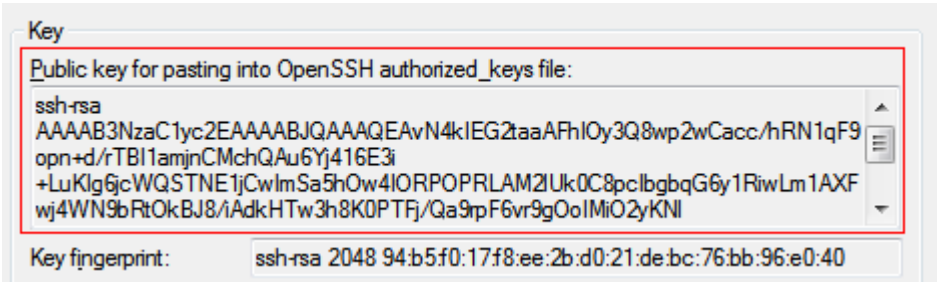


- Move your mouse around the blank area to generate randomness to the key.

Note: the dotted red line in the image below is for illustration purposes only. It does not appear in the generator pane as you move the mouse.

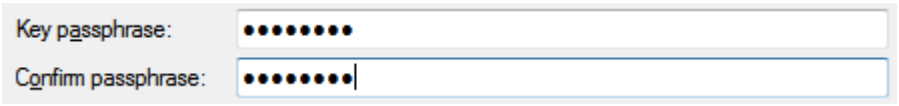


- The generated key appears under Public key for pasting into OpenSSH authorized_keys file.



- The key comment is the name of the key that you will use to identify it. You can keep the generated key comment or create your own.

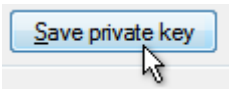
- If you want to password-protect your key, enter a **Key passphrase** and enter it again for **Confirm passphrase**. When you reload a saved private key, you will be asked for the passphrase, if one is set.
-



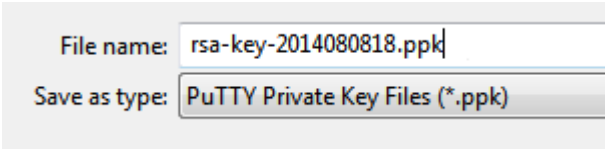
While a passphrase is not required, you should specify one as a security measure to protect the private key from unauthorized use.

There is no way to recover a passphrase if you forget it.

- Save the private key of the key pair. Depending how you work with the private key in the future, you may need one saved in the PuTTY PPK format and one saved in OpenSSH format. Let's do both.
    - To save the key in the PuTTY PPK format, click **Save private key** to save the private key of the key pair.

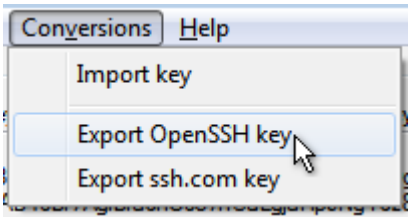


You can name it anything you want, although you may want to use the same name as you used for the key comment. The private key is saved in PuTTY's Private Key (PPK) format, which is a proprietary format that works only with the PuTTY toolset.

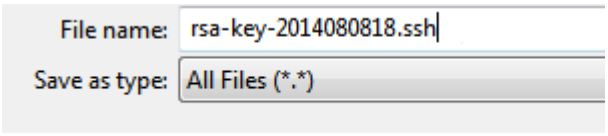


You can use this key whenever you use Putty to perform SSH actions.

- To save the key in OpenSSH format, open the **Conversions** menu and select **Export SSH key**. This will be the same key as above, just saved in a different format.

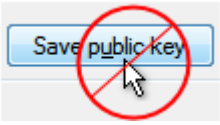


You can name it anything you want, but to keep track of your keys, you should give it the same name as the key you saved in PPK format in the previous step. You can also use any extension (or no extension), but let's use .ssh, to make it clear what format it is.



You can use this key whenever you use OpenSSH to perform SSH actions using ssh utitlities that support OpenSSH, for example when using Linux in a command shell.

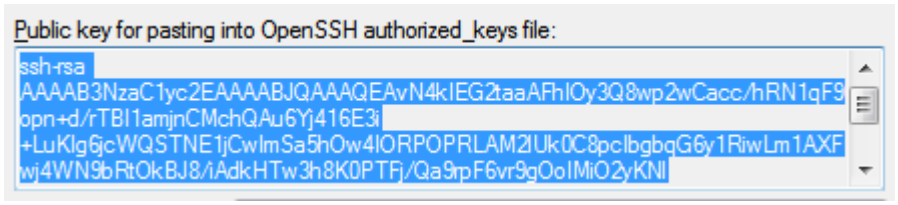
- Now you need to create the public key to be paired with the private key(s) you just created. However, clicking the **Save public key button** will create a public key that won't work with Oracle Cloud services in certain cases. So, for the purposes of this tutorial, there is no reason to save a public key using the **Save public key** button.



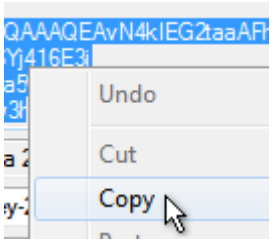
Instead, proceed as follows.

- In the PuTTY Key Generator, select all of the characters under Public key for pasting into OpenSSH authorized_keys file.

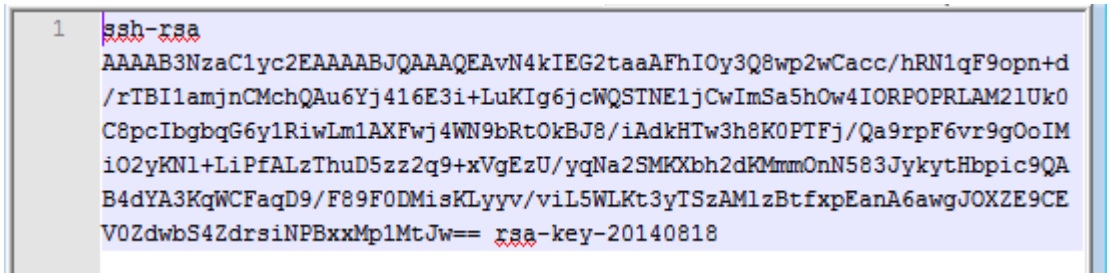
Make sure you select all the characters, not just the ones you can see in the narrow window. If a scroll bar is next to the characters, you aren't seeing all the characters.



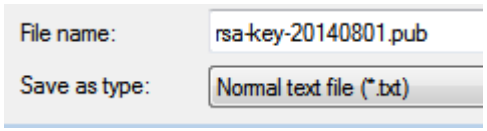
Right click somewhere in the selected text and select **Copy** from the menu.



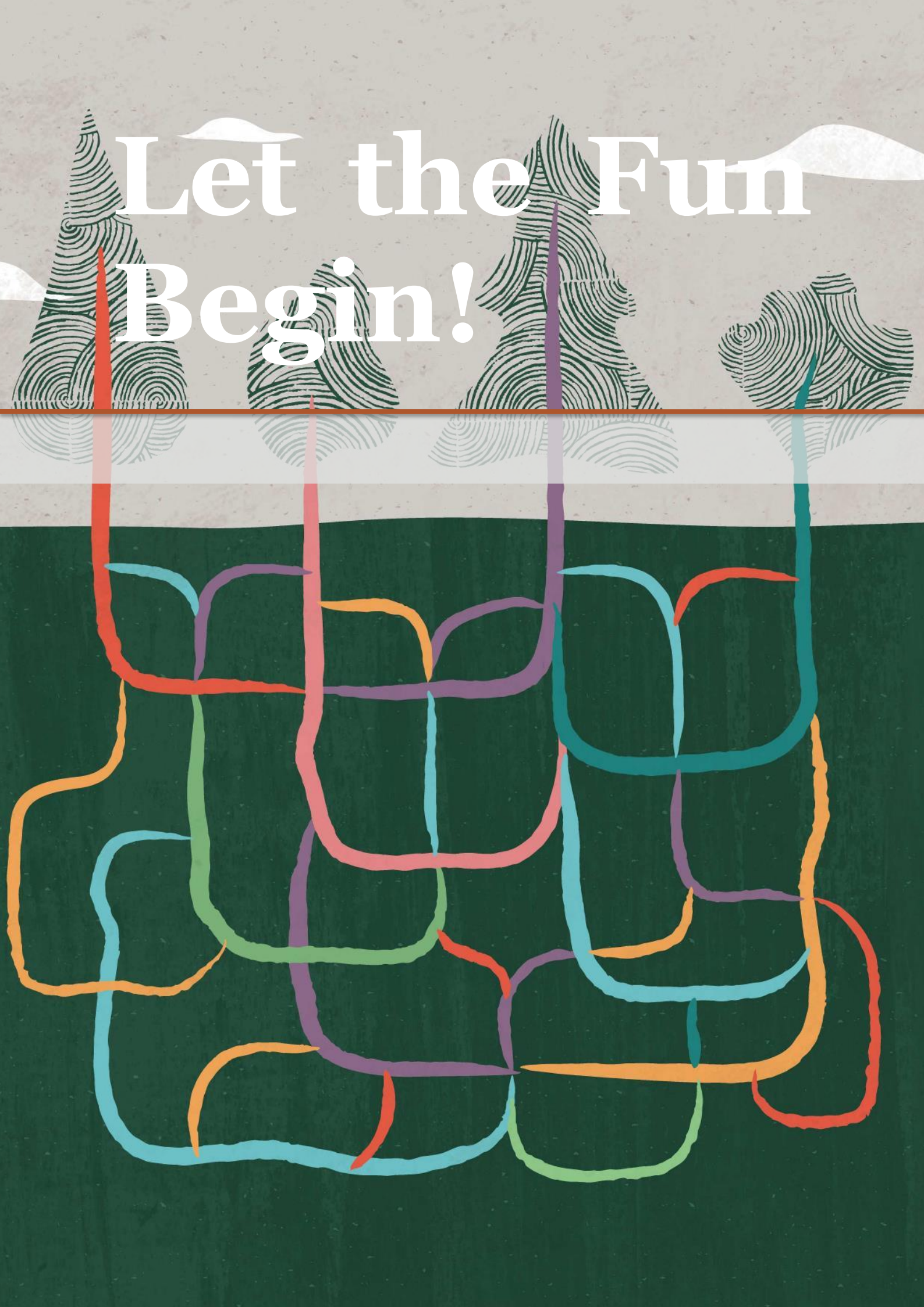
- Open a text editor and paste the characters, just as you copied them. Start at the first character in the text editor, and do not insert any line breaks.



- Save the key as a text file, using the same root name as you used for the private key. Add a .pub extension. You can give it any extension you want, but .pub is a useful convention to indicate that this is a public key.



Write down the names of your public and private keys, and note where they are saved. You will need the public key when creating service instances in, for example, Oracle Java Cloud Service and Oracle Database Cloud - Database as a Service. You will need the private key when trying to access a service instance's virtual machine via SSH.


Let the Fun Begin!

4. Signing in to the OCI Console

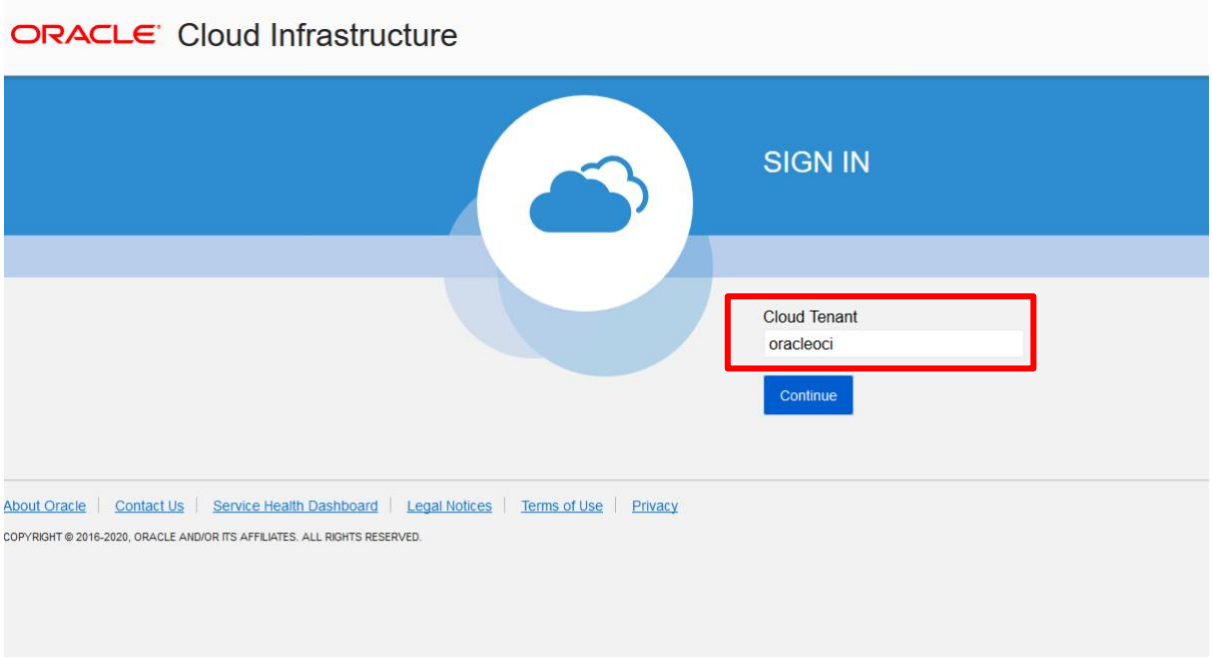
Oracle Cloud Infrastructure Identity and Access Management (IAM) Service lets you control who has access to your cloud resources. You control the types of access a group of users has and to which specific resources. The purpose of this lab is to give you an overview of the IAM Service components and an example scenario to help you understand how they work together.

Pre-requisites:

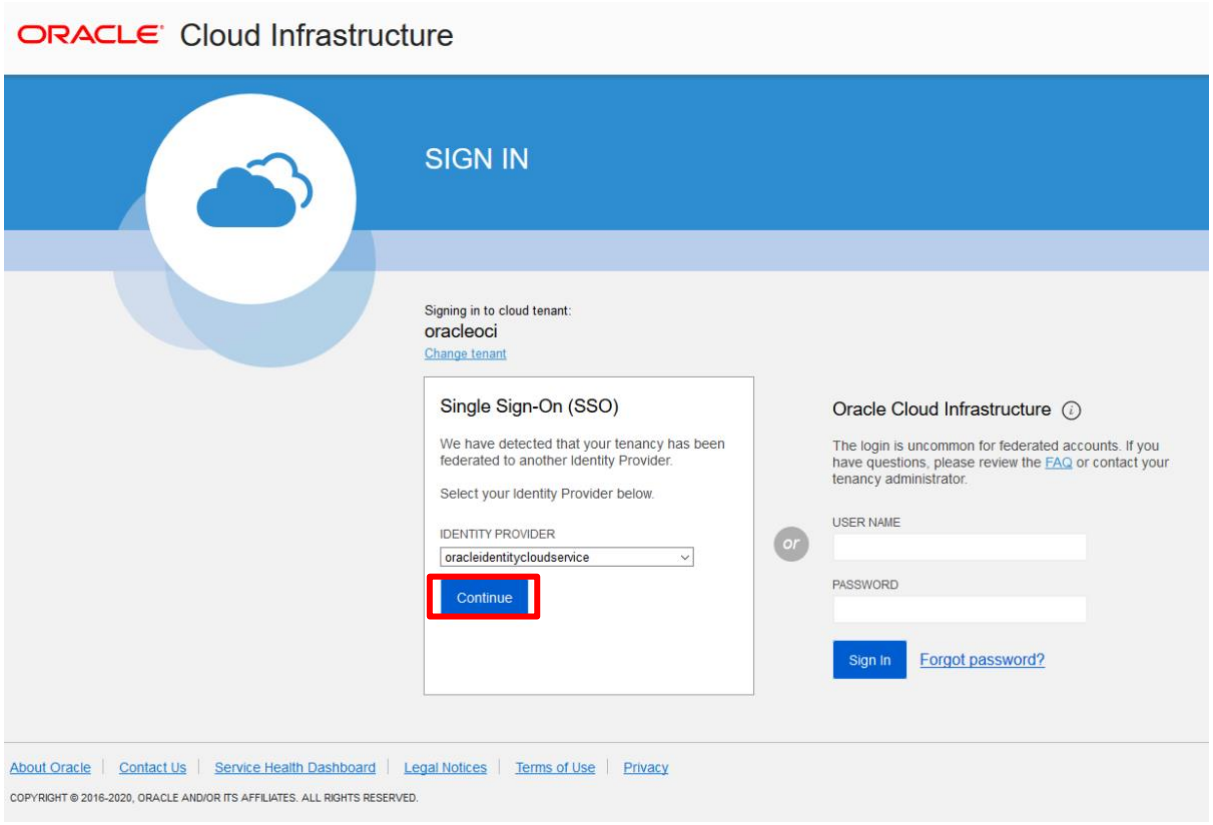
- Oracle Cloud Infrastructure account credentials (User, Password, and Tenancy)
- To sign in to the Console, you need the following:
    - Tenant, User name and Password
    - URL for the Console: https://console.eu-frankfurt-1.oraclecloud.com/
    - Oracle Cloud Infrastructure supports the latest versions of Google Chrome, Firefox and Internet Explorer 11

In this Lab, you will sign in to the Oracle Cloud Infrastructure console using your credentials.

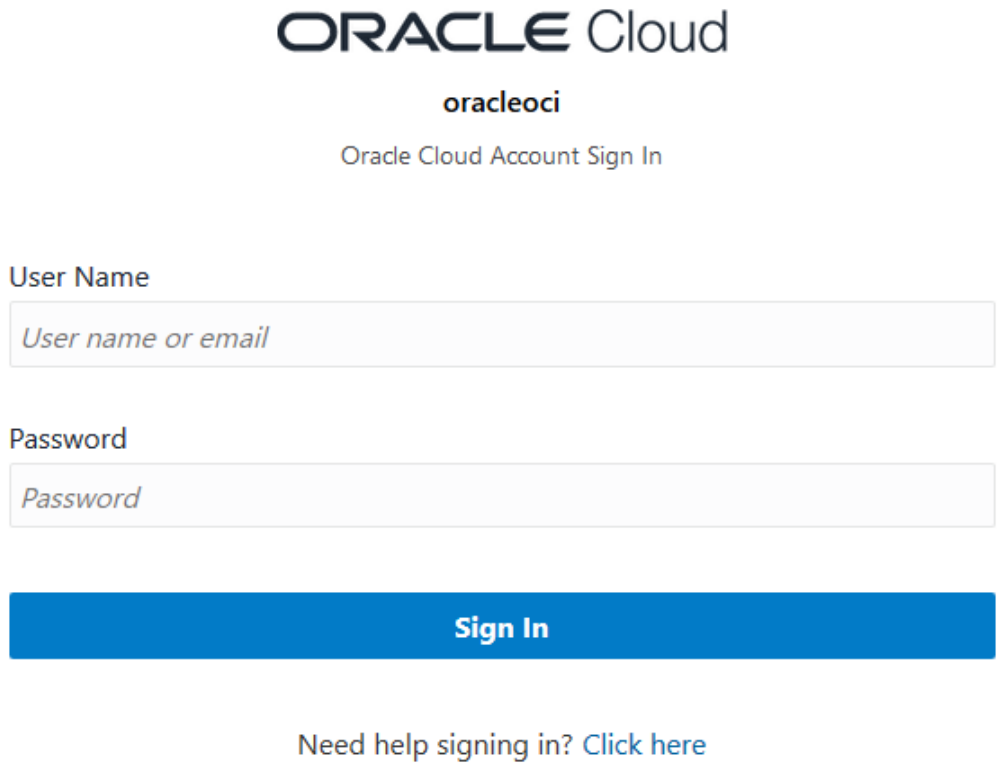
1. Open a supported browser and go to the Console URL: https://console.eu-frankfurt-1.oraclecloud.com/.
2. Enter your tenant name and click **Continue**



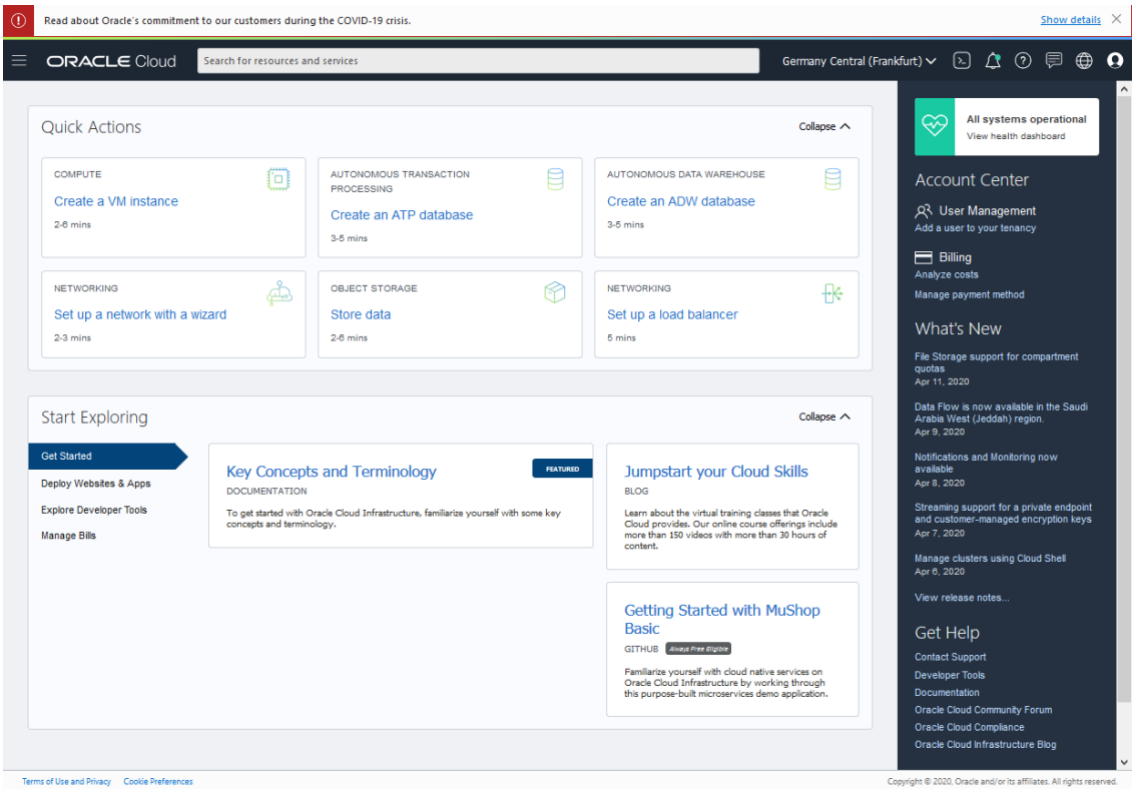
3. Oracle Cloud Infrastructure is integrated with Identity Cloud Services, you will see a screen validating your Identity Provider. Click **Continue**.



4. Enter your user name and password
   - **Username:** *instructor will provide username*
   - **Password:** *instructor will provide password*



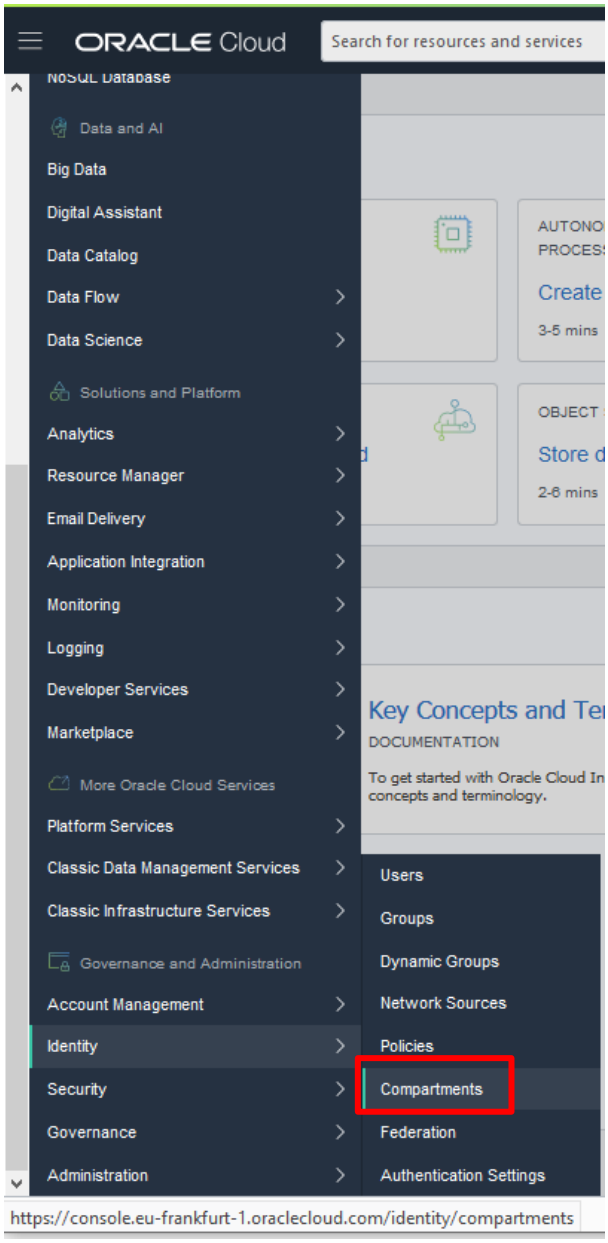
5. When you sign in to the Console, the dashboard is displayed.



5. Creation of a compartment

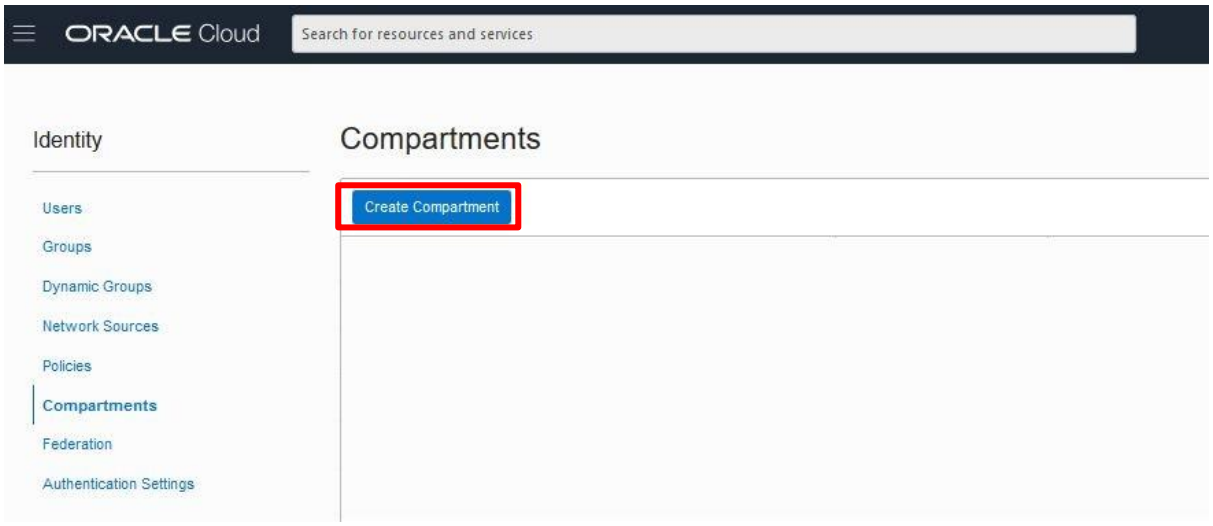
A **compartment** – is a collection of related resources that can be accessed only by certain groups that have been given permission by an administrator in your organization. When you first start working with Oracle Cloud Infrastructure, you need to think carefully about how you want to use compartments to organize and isolate your cloud resources. Compartments are fundamental to that process. Most resources can be moved between compartments.

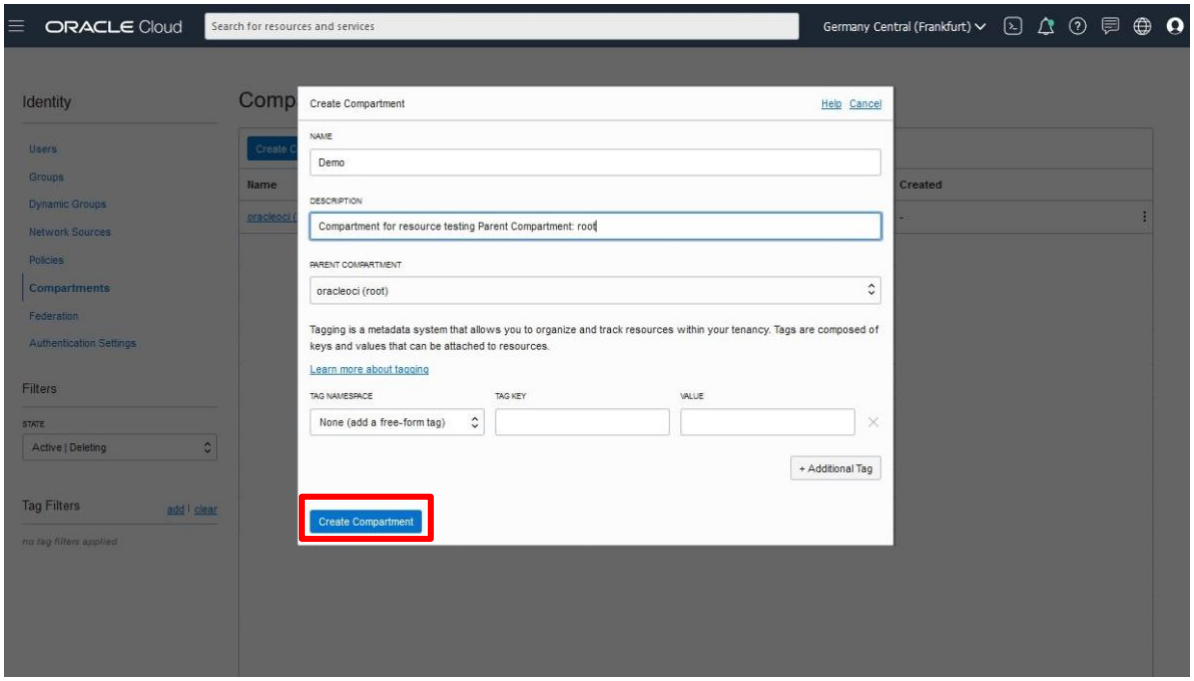
In order to view and create a compartment in your tenancy, please navigate to **Main Menu**, **Governance and Administration** section, **Identity** and select **Compartments**.



► Click on "**Create Compartment**" and fill the information:

Name: *Demo*
Description: *Compartment for resource testing Parent Compartment: root*

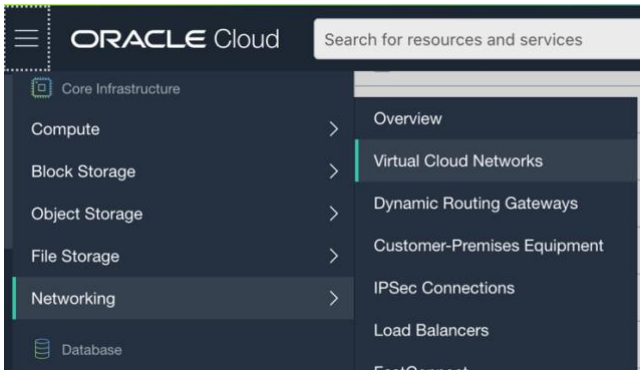




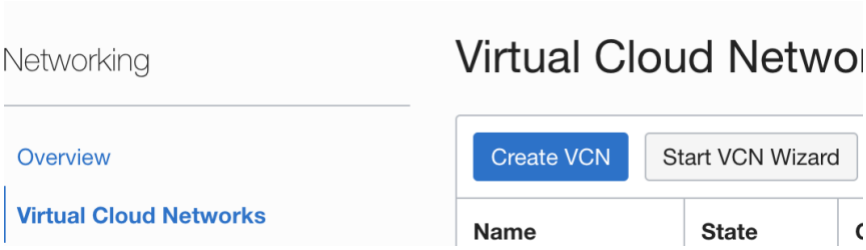
To learn more about compartments in OCI, please visit: https://docs.cloud.oracle.com/en-us/iaas/Content/Identity/Tasks/managingcompartments.htm

6. Create a virtual cloud network via wizard

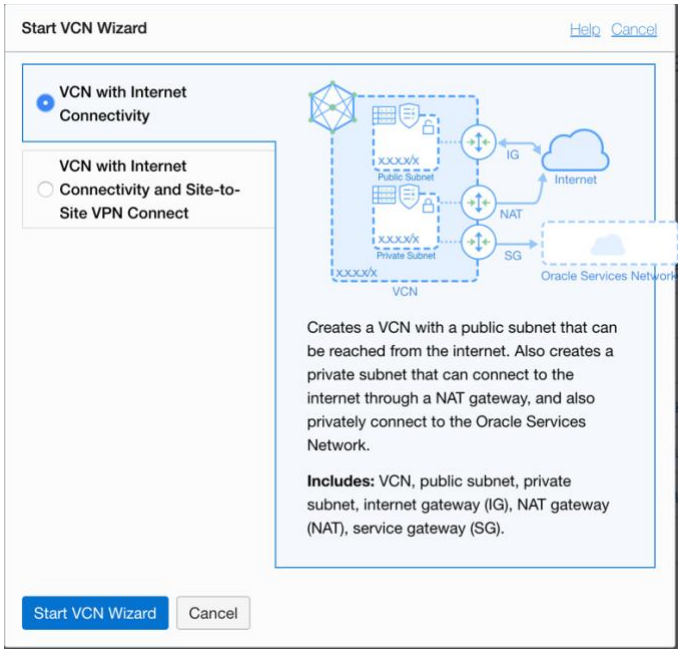
Select from the OCI menu -> Networking -> Virtual Cloud Networks



Click on start VCN wizard



Select VCN with Internet Connectivity and Start VCN Wizard



Complete the requested informations and review all the configurations. Click create to finish.

7. Create Database Instance

Once the network is completely created you are able to deploy a Database instance via Database Cloud Service.

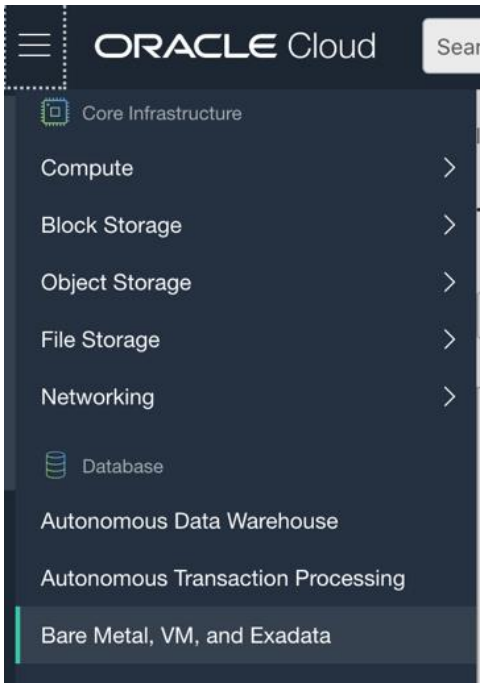
As best practice, all database instances must be created in private subnets to prevent any external surface attacks. Of course, if you want to access the instance via SSH or via SQL Developer you have two ways :

1) Create a Linux VM in the public subnet with Public IP enabled. Use this instance as Bastion Host to access the private network of your VCN.
    a) Remember to add the private key to your ssh session.
        i) Windows Systems: Putty **pageant.exe** program to store the private key in the ssh session
        ii) Mac systems: use the native command **ssh-add path/to/file/key.pem** to store the private key in the ssh session
        iii) You can also create a local forward to your PC/MAC in order to access the database via local port with :
            (1) ssh -L 1521:databaseIP:1521 -i path/to/private/key.pem opc@bastionpublicip
2) Setup an IPsec VPN via DGR Gateway to your office and access the instance via private IP.

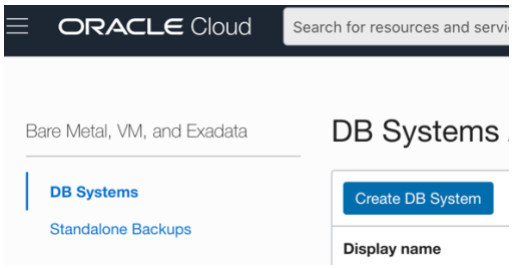
If you are already familiar with this configurations, we suggest to proceed with the private way.

If you are not, to keep it simple as for demonstrations purpose, we are going to deploy the database instances in public subnet.

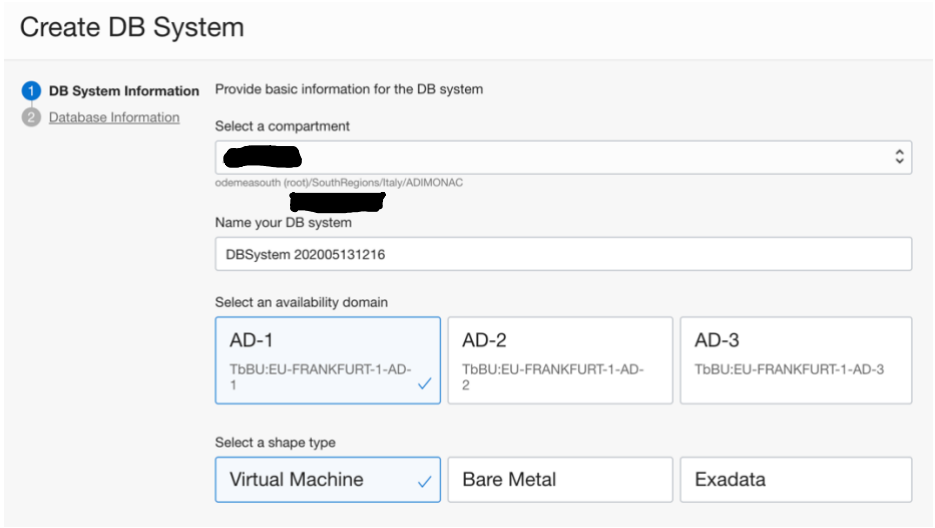
Select from the OCI main manu the service: Bare Metal, VM, and Exadata in the Database section



From the new page, click on create DB system



Fill the following informations

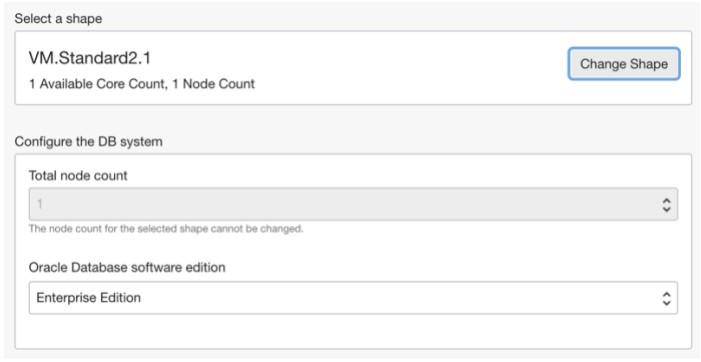


Select the logical compartment you have created.

Select an Availability Domain

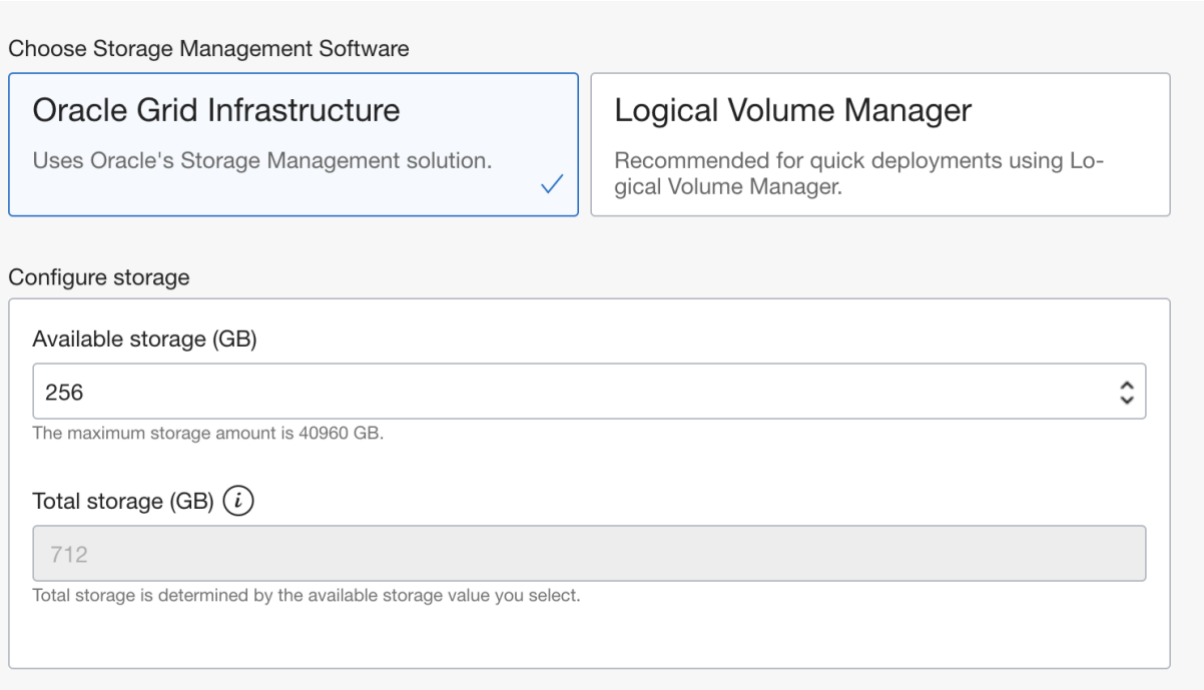
Use the Virtual Machine shape type to keep the cost low for the demo environment

Select shape VM.Standard2.1 and Oracle Database Software Edition as Enterprise Edition

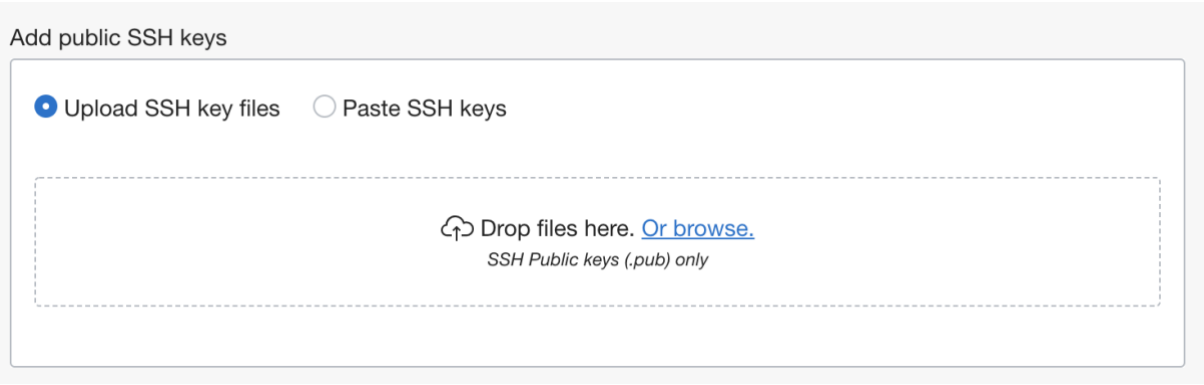


Note: If you select the Enterprise Edition Extreme, you are able to create a 2 Node RAC with 2VM of 2 OCPU (minimum) each. This configuration reflects the minimum requirements for the **Silver MAA recovery scenario.** In order to keep the cost low, we are deploying a single instance.

Keep the default values:

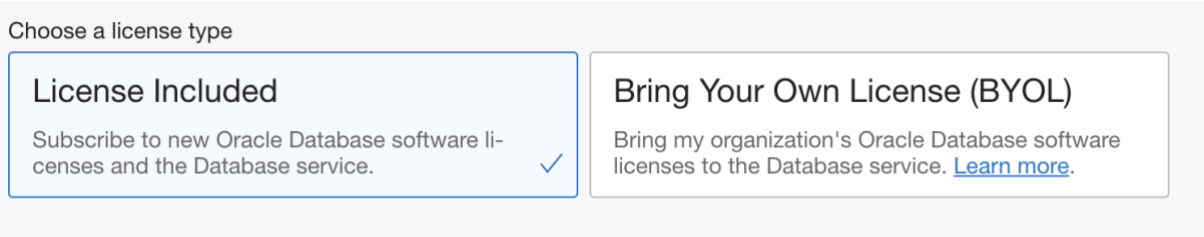


Add the private key you have previously created:

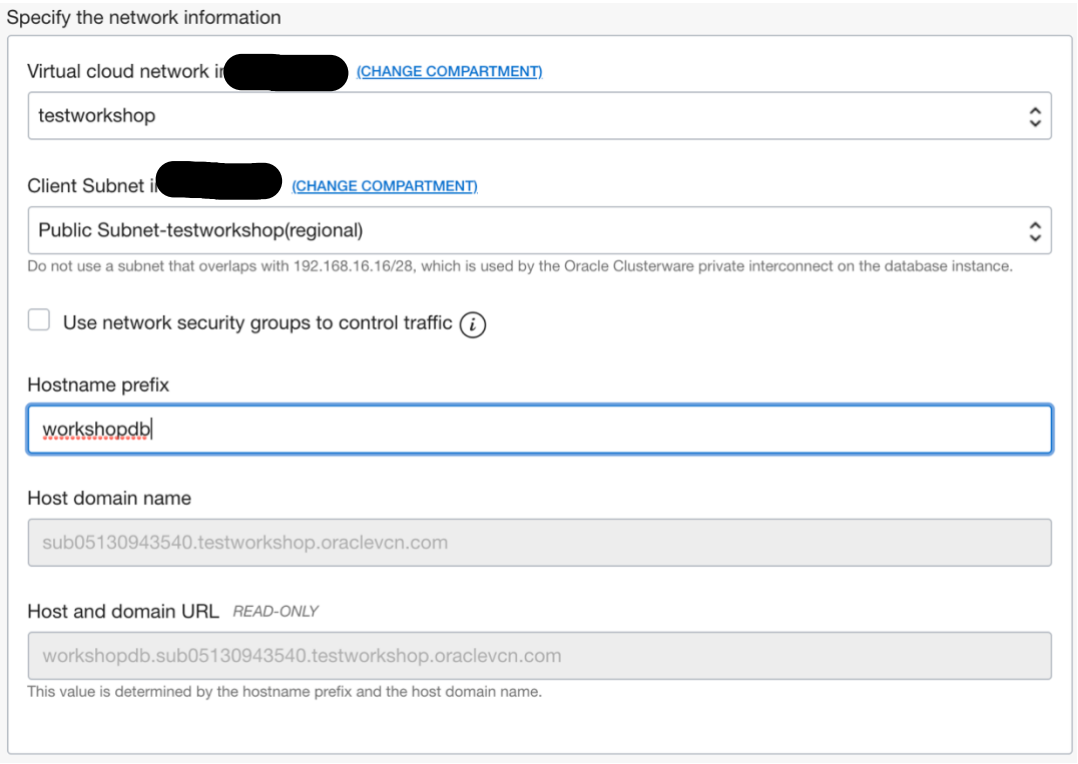


Remember that you have root access to the instance with the user “opc”

License type use Included



Add the networking information. Select the VCN created via Wizard, the public subnet and define an hostname prefix



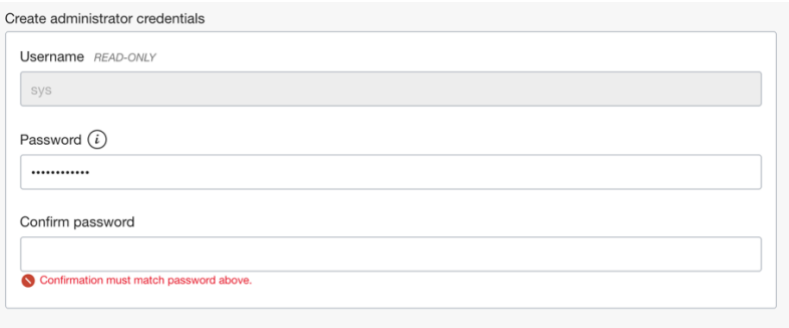
Click next

Provide a Database Name and use the 19c as version

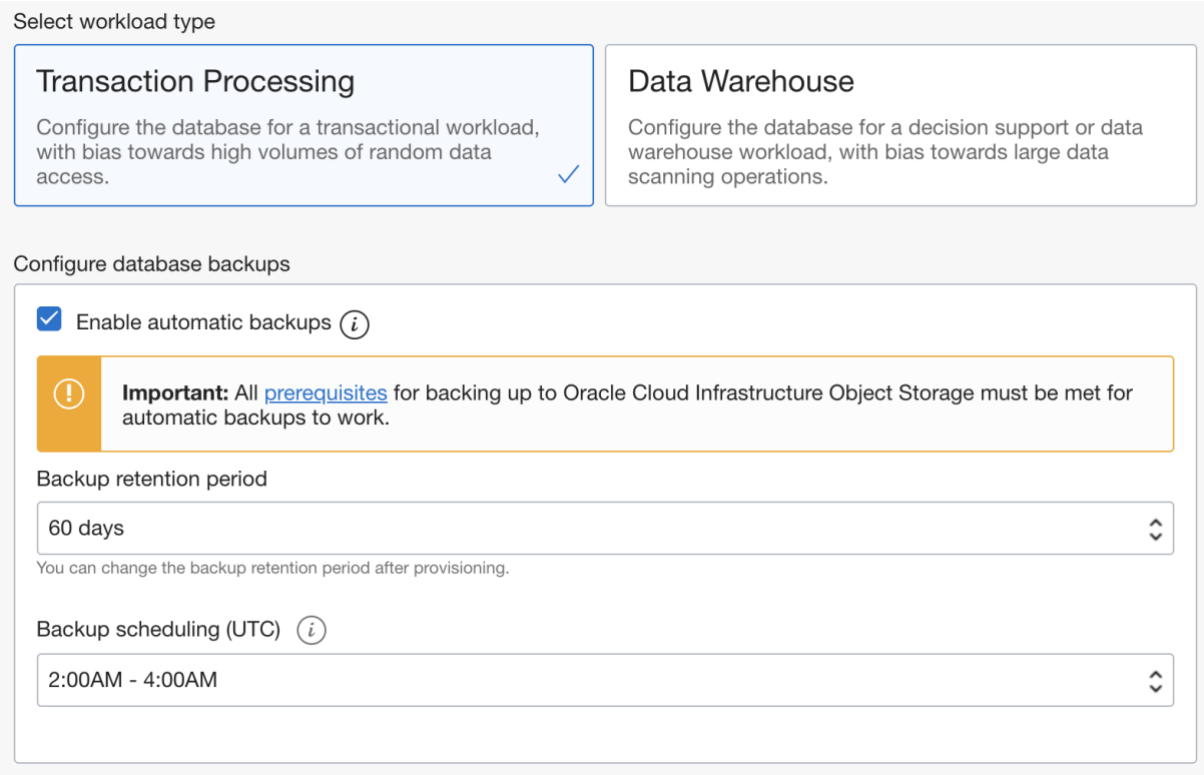
Create and remember the sys password in a safe place

Password difficulty should be like 2 Upper case, 2 lower case, 4 numbers, 4 special caracters.

Password must be 9 to 30 characters and contain at least 2 uppercase, 2 lowercase, 2 special, and 2 numeric characters. The special characters must be _, #, or -.



Select the workload type you like and enable automatic backups as the Bronze MAA scenario requires for the initial level of restore procedure.



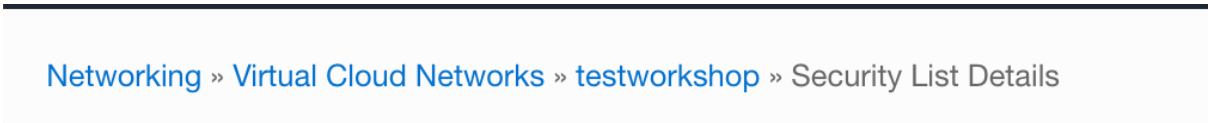
You are also able to create manual full backup outside the retention process.

In show Advanced options you can configure Character Set and National character Set.

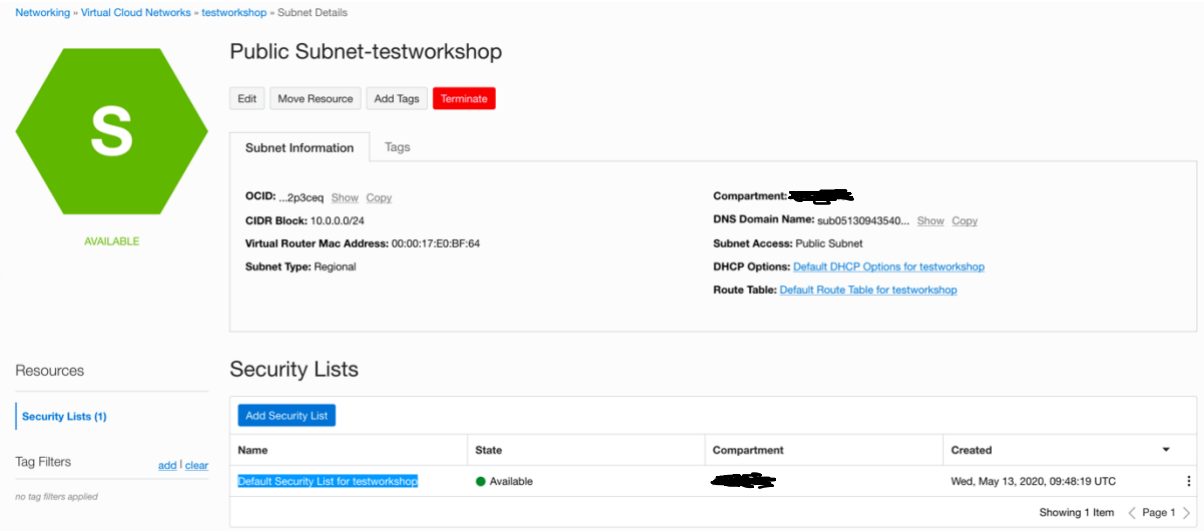
Click on Create DB System and wait from provisioning status to available, it can take up to 2 hours, and additional 1/2 hour for the backup configuration if enabled.

In the meanwhile let's add the security configuration for the Database

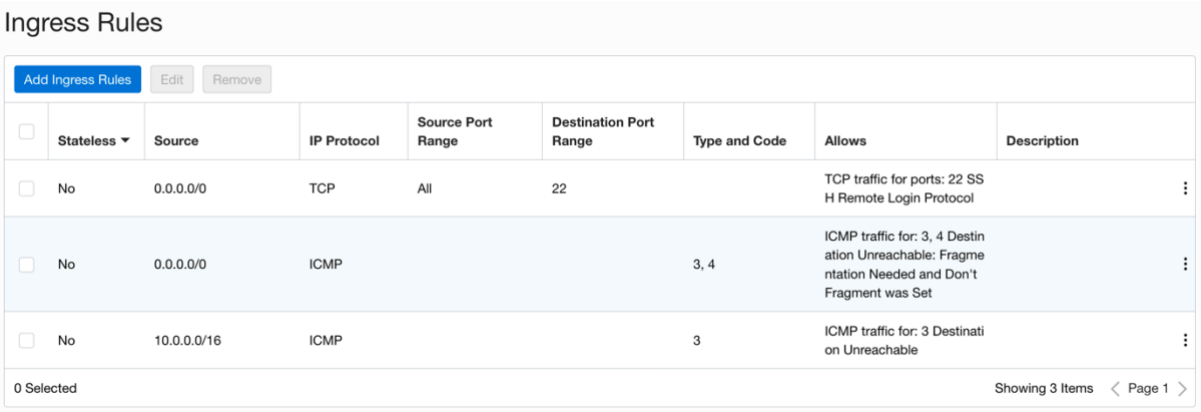
From the OCI main menu go to Netwoking -> Virtual Cloud Networks -> Your VCN -> Select the Public Subnet in order to select the Security List associated.



Select the Default Security List for your VCN public subnet



As you can see from the ingress rules you don't have any 1521 port enabled as ingress rule



Add 2 Ingress Rules:

1. Source your actual public IP address (google what's my IP) in /32 from All to 1521 destination port (If your ISP use dynamic public IPs for your connection, remember to refresh it)
2. Source 10.0.0.0/16 (or your vcn private address) from All to 1521 destination ports

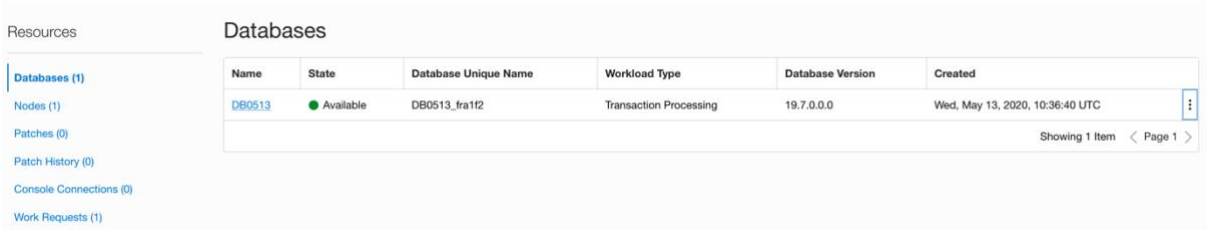
Example:

Now you're ready to proceed to the next configuration.

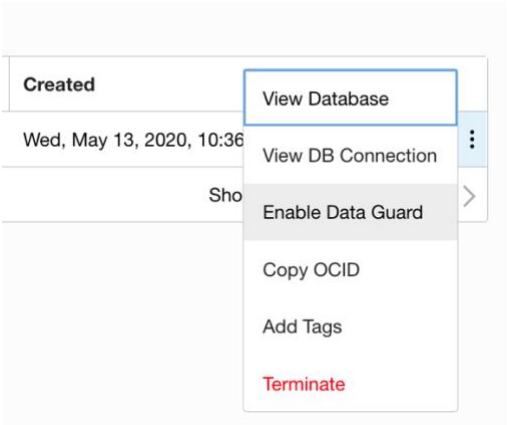
8. Create Data Guard

Now you have network and primary instance in place, we can proceed to the stand-by instance with data guard.

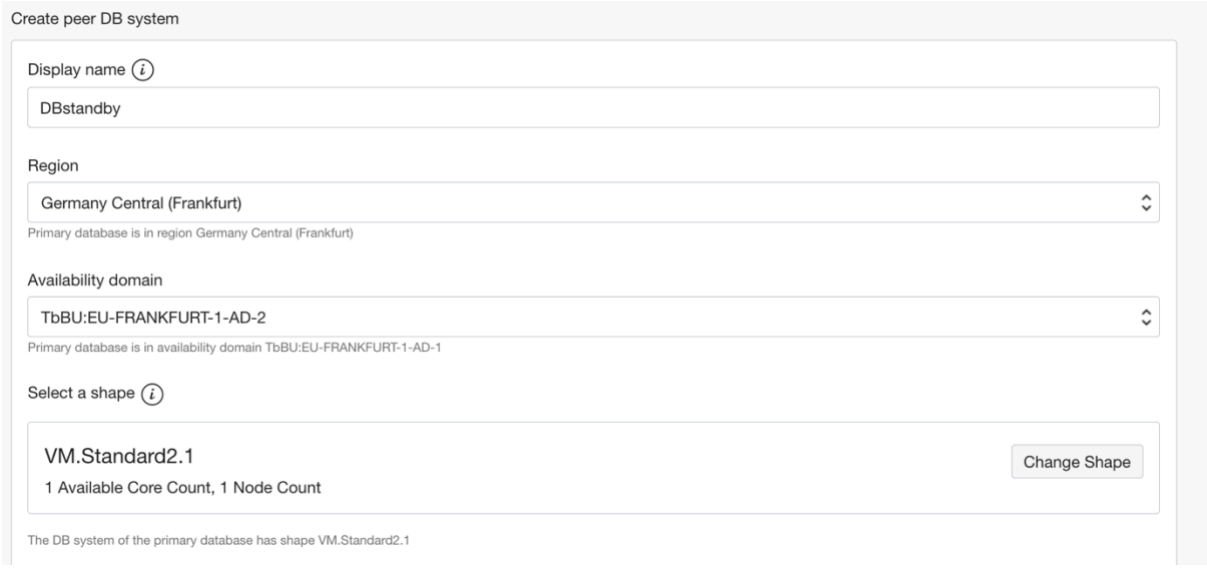
From the created database instance page you can see the following screen at the bottom.



Click on the 3 vertical dots on the right and select Enable Data Guard.



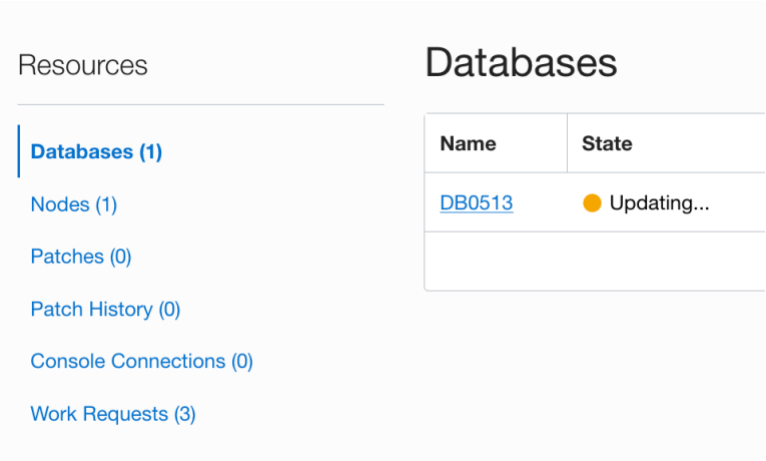
This configuration will create a stand-by instance with async replication of data up to the last commit.



Define the name, use the same region with a different Availability Domain in order to deploy the standby in a secondary datacenter about 30 Km far away from the primary.

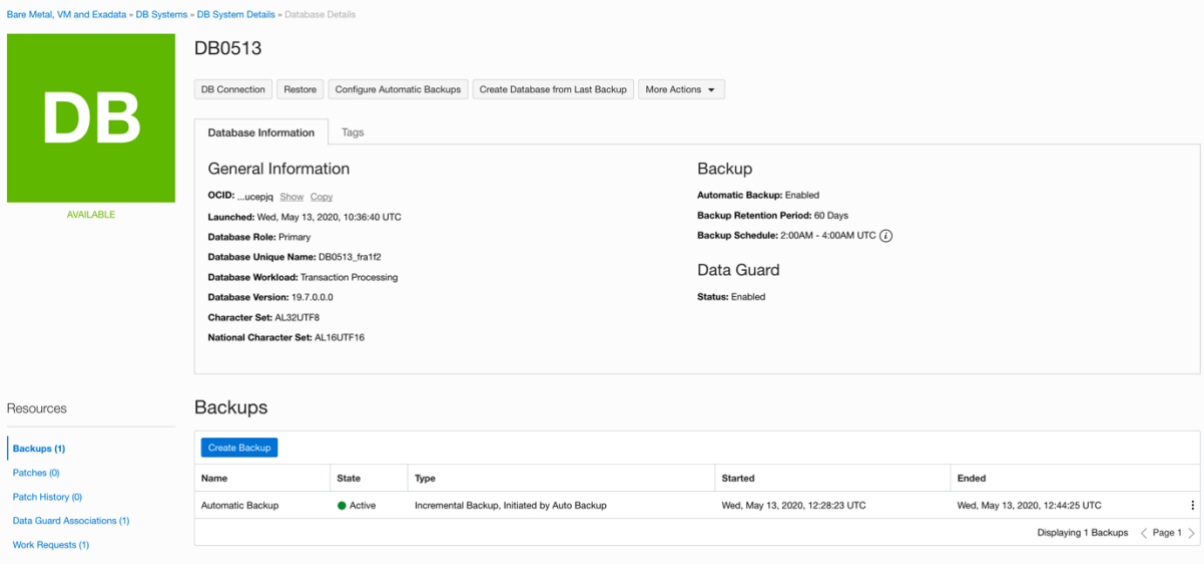
Add the same sys password used for the primary database and click create.

Now you should see the updating status on the database instance



Wait again up to 2 hours in order to let the standby instance to be deployed and configured with data guard.

At the end you should see the Data Guard Enabled status in the database page.



You can now setup your SQL Developer client to access the primary instance, or you can use the ssh connection to use the SQL plus command line.

You can also switch roles from primary to standby directly from the console. We suggest, in order to avoid mistakes, to use a specific OCI CLI (OCI command line) to perform any DG change.

You have successfully completed the workshop.

Now you know how to create DB Instance on Oracle Cloud and how to archive up to the Gold architecture MAA with:

- 2 Node RAC instance + Data Guard Configuration in a different Datacenter + Backup configuration with 60 days of retention.