

SISTEMA DE MONITOREO PARA PYMES

ANEXO – SPRINT 1

Despliegue y Configuración de Infraestructura

Escuela de Ingeniería

Agosto, 2023

NOMBRE DEL ALUMNO

Patricio Valenzuela Baeza

NOMBRE DOCENTES

Oscar Pinto Garralaga

INDICE

1. Creación de Tenant en Cloud	3
1.1 Conectarte a la nube	3
1.2 Ingreso de información	3
1.3 Configuración del Tenant	4
1.4 Información de Dirección	5
1.5 Test de creación de Tenant	6
2. Despliegue de Infraestructura	7
2.1 Creación de Compartments	7
2.2 Creación de Clúster Kubernetes	8
2.3 Test creación de Clúster Kubernetes	11
2.4 Creación del Bastión	12
2.5 Test de creación del Bastión	14
2.6 Conectarse al Bastión desde una red On Premise	14
2.7 Test de conexión al Bastión desde una red On Premise	15
2.8 Actualización de Sistema Operativo del Bastión	16
2.9 Conectarse al Clúster Kubernetes desde el Bastión	16
2.10 Test de conexión al Clúster Kubernetes desde el Bastión	17
2.11 Instalación de Ansible	17
2.12 Instalación de Podman	18
2.13 Creación y configuración de Registry Local	18
2.14 Creación secrets en Kubernetes	19
2.15 Test de creación secrets en Kubernetes	21
2.16 Creación de Servidores a monitorear	21
2.17 Instalación de MySQL	22
2.18 Administración y configuración de usuario MySQL	23
2.19 Prueba de MySQL	23
3. Creación y Configuración de Git y GitHub	24
3.1 Crear repositorio GitHub	24
3.2 Prueba de repositorio GitHub	25
4. Bibliografía	26

1. Creación de Tenant en Cloud

Para nuestra primera actividad del Sprint 1 es la creación de la cuenta gratis en una nube publica, para este proyecto utilizaremos una cuenta destinada para crear las definiciones de POC (Pruebas de conceptos) y o laboratorios para nuestros clientes en la compañía en la cual actualmente trabajo.

Pero como este es un trabajo para presentar en nuestro ramo de Seminario de Titulo detallare los pasos a seguir para crear una cuenta Free la cual es sin costos de utilización y limitado a ciertos recursos y capacidades de almacenamiento para que sirva de guía para eventos futuros. Adicionalmente y tal como indica el proveedor “Crea, comprueba e implementa aplicaciones en Oracle Cloud de forma gratuita.”¹ el detalle para comenzar el aprendizaje.

1.1 Conectarte a la nube

Para ello nos conectaremos a la URL <https://www.oracle.com/cl/cloud/free/> para definir nuestro tenant en la región de Chile donde seleccionaremos la opción “Comienza gratis” como aparece en la Figura 1.



Figura 1 – Inicio de creación de tenant free (Fuente: Oracle.com)

1.2 Ingreso de información

A continuación, se solicitará el ingreso de información personal como la región donde se desea crear el tenant de uso “Always Free Resources” como lo indica en la Figura 2.

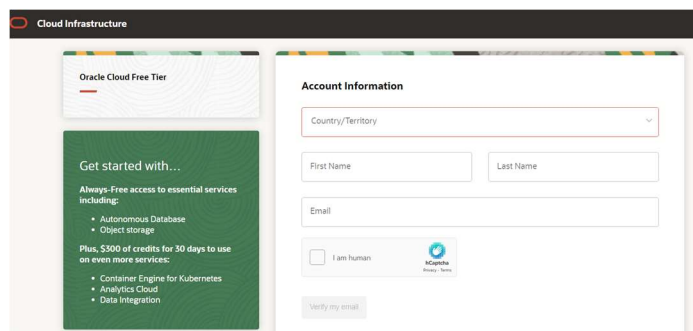


Figura 2 – Información para creación de tenant free (Fuente: Oracle.com)

Al ingresar toda la información solicitada deberá seleccionar el checkbox “I am human” y resolver el CAPTCHA que será desplegado, una vez resuelto deberá presionar el botón “Verify my email” que le enviará un correo a la cuenta ingresada.

¹ (ORACLE, 2023)

1.3 Configuración del Tenant

Al verificar el correo de verificación que le fue enviado deberá presionar el botón “Verify email” como se muestra en la Figura 3.

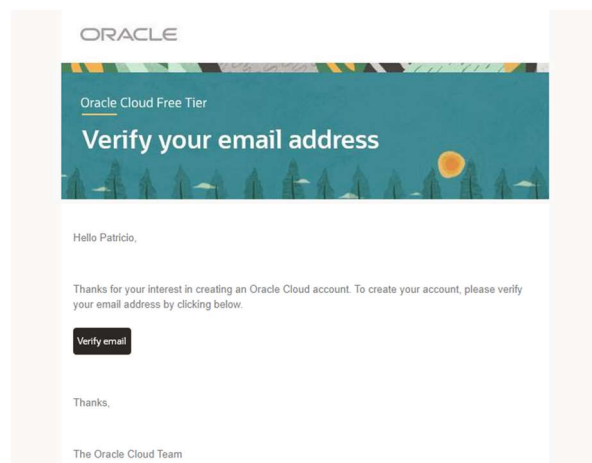


Figura 3 – Verificación de Email (Fuente: Proveedor de correo personal)

Al verificar el email en el browser que tenga por default le desplegara la solicitud de información para realizar la configuración del tenant la cual dividiremos en dos y la primera parte es como se muestra en la Figura 4 donde solo se deberá ingresar la contraseña de administración para el ingreso a la consola del tenant.

Account Information

Country/Territory
Chile

First Name
Patricio

Last Name
Valenzuela

Email
patricio.valenzuela.baeza@gmail.com

Password

⚠ Password must contain a minimum of 8 characters, 1 lowercase, 1 uppercase, 1 numeric, and 1 special character.
✅ Password cannot exceed 40 characters, contain the users first name, last name, email address, spaces, or ` ~ < > \` characters.

Confirm Password

Figura 4 – información de administración del tenant (Fuente: Oracle.com)

En la segunda parte se deberá definir el tipo de tenant el cual se está configurando y la región donde desplegaremos nuestros recursos como se muestra en la Figura 5 en el cual se deberá seleccionar “Individual”.

Customer type

☒ Corporate ☐ Individual

Legal Company Name

Cloud Account Name
patriciovalenzuelabaeza

Home Region

⚠ Because of high demand for Arm Ampere A1 Compute capacity in the South Korea Central (Seoul) and Japan East (Tokyo), A1 instance availability in these regions is limited. If you plan to create A1 instances, we recommend choosing another region as your home region.

Figura 5 – Tipo de cliente y Nombre de Tenant (Fuente: Oracle.com)

Para la región de despliegue seleccionaremos Chile Central (Santiago) ya que tenemos una latencia de 9.8ms en comparación a otras regiones como por ejemplo US East (Ashburn) que tenemos una latencia de 123ms y para nuestros efectos del proyecto si seleccionamos una región distinta deberemos crear otros recursos los cuales no estaban contemplados dentro de nuestro proyecto. En la solicitud de “Cloud Account Name” corresponde a lo que conoceremos como el nombre del tenant al cual nos referiremos en todo este documento.

Una vez ingresada la información solicitada deberemos presionar el botón “Continue” el cual solicitara la información de localización.

1.4 Información de Dirección

En esta sección se solicitará ingresar la información de localización tal como está definido en la Figura 6 y la información solicitada es la siguiente:

- Direcciones
- Ciudad País
- código postal
- Número de Teléfono

Adicionalmente solicitara la información de “Payment Verification” o “Verificación de Pago” para lo cual deberá ingresar la información de una tarjeta de crédito vigente en donde se realizara solo un cargo por un monto de 1 US\$.

Esta información de pago es solo para resguardo de que se cumplan las estipulaciones definidas en las regulaciones de los tenant “Always Free Resources” por lo tanto siempre se deberán leer previamente estas regulaciones.

Address Information

Address Line 1

A value for Address Line 1 is required.

Address Line 2

Optional

Address Line 3

Optional

City

State

Optional

Zip/Postal Code

Comuna

Phone Number

+56

Think prefix codes are not used when entering your mobile number (only use 03.. instead of "0"03.. or "1"03..). Enter numbers without spaces and special characters included.

Please provide a valid phone number. Oracle does not accept text only mobile numbers as we may need to speak to you if there are questions about your account.

Continue

Payment Verification

Agreement

☐ By clicking Start my free trial, I agree to the terms and conditions of the [Oracle Cloud Services Agreement for Sistemas Oracle de Chile Limitada](#) (also available [here](#)) and this order including Service Description for Free Oracle Cloud Promotion Universal Credits - Part Number 888555

Start my free trial

Figura 6 – Direcciones e Información de Pago (Fuente: Oracle.com)

Una vez ingresada toda la información se deberá leer los términos de la creación del tenant y para aceptar estos términos seleccionar el Checkbox indicado como “Agreement”.

Al finalizar todo lo solicitado y validar la información requerida se deberá presionar la opción “Start my free trial”.

1.5 Test de creación de Tenant

Para las pruebas de la correcta creación del tenant nos conectaremos a la URL “https://www.oracle.com/cloud/sign-in.html” la cual desplegará una página en nuestro browser como se muestra en la Figura 7.

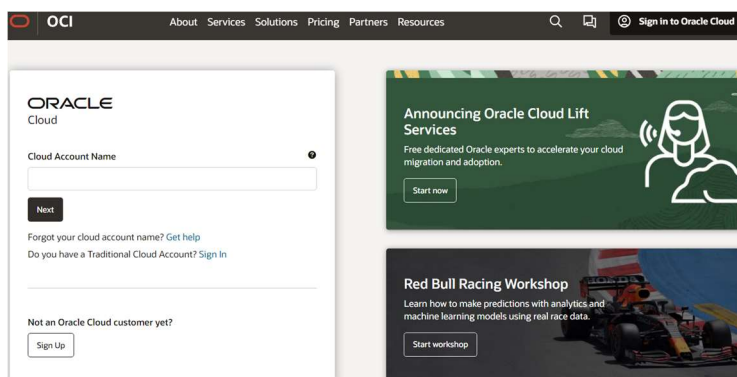


Figura 7 – Conexión al Tenant Free (Fuente: Oracle.com)

En la opción “Cloud Account Name” se deberá ingresar el mismo valor ingresado en la Figura 5 donde se solicita el mismo nombre de parámetro y presionaremos Next y nos desplegará la opción de ingreso de las credenciales donde solo seleccionaremos la opción “continue” como se muestra en la Figura 8 ya que no hemos configurado Multi Factor Authenticator o algún LDAP adicional como para autenticarse con un “Single Signon”.

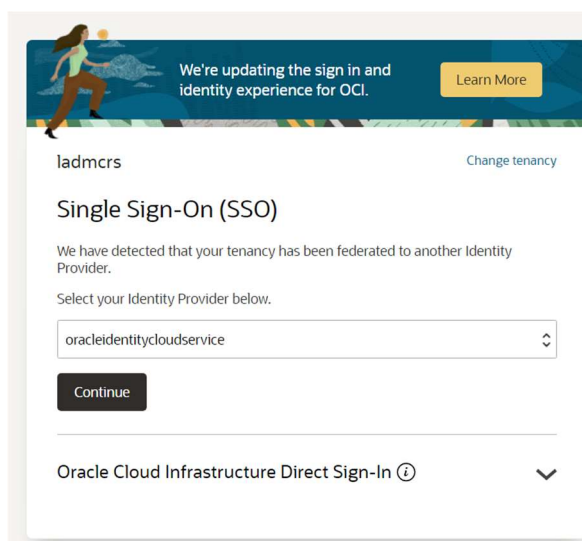


Figura 8 – Conexión al Tenant Free (Fuente: Oracle.com)

Una vez conectados satisfactoriamente nos desplegara la consola de nuestra tenant más conocido como Oracle Cloud Infrastructure o su acrónimo OCI tal como se muestra en la Figura 9.

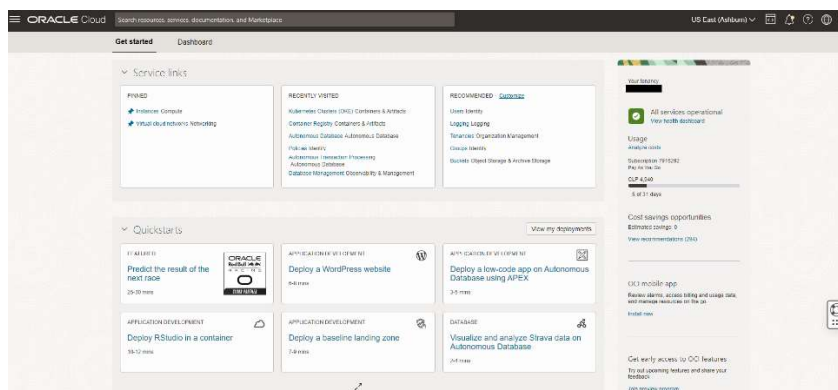


Figura 9 – Consola Oracle Cloud Infrastructure (Fuente: Oracle.com)

2. Despliegue de Infraestructura

2.1 Creación de Compartments

Un compartimento es un contenedor lógico, para organizar y controlar el acceso a los recursos de Oracle Cloud Infrastructure (OCI) (computación, almacenamiento, red, Load Balancer, etc.) creados dentro de ese compartimento e impone algunas políticas a ese compartimento, lo que restringe quién puede usar los recursos creados dentro de ese compartimento que no sean administradores de su cuenta. Tal como indica la documentación oficial “La consola está diseñada para mostrar sus recursos por compartiment dentro de la región actual”² y en todas las regiones donde se encuentre registrado el tenant.

Para realizar la creación del compartiment donde trabajaremos seleccionaremos en el menú la opción “Identity & Security” y posteriormente a mano derecha en el menú que se desplegara seleccionaremos la opción “Compartment” como se demuestra en la Figura 10.

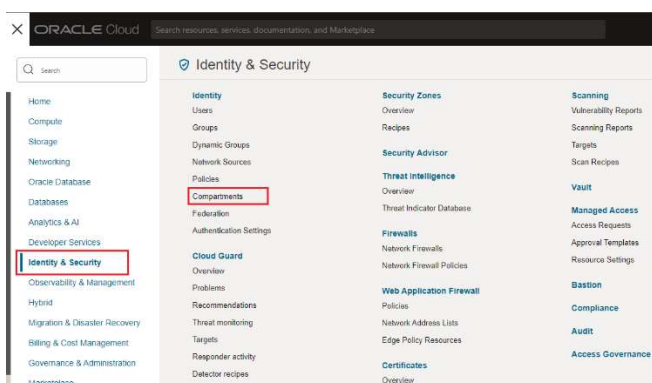


Figura 10 – Creación de Compartiment (Fuente: Consola OCI)

² (IAM-Compartment, 2023)

Posteriormente al seleccionar la opción de compartment nos desplegara las opciones de creación como se muestra en la Figura 11 donde seleccionaremos la opción “Create Compartment”.



Figura 11 – Creación de Compartment (Fuente: Consola OCI)

Una vez seleccionada la opción nos desplegara las opciones de configuraciones para crear el compartment como se muestra en la Figura 12, al completar la información donde solo ingresaremos la información del Nombre y la Descripción y presionar la opción “Create Compartment” damos por finalizada la creación.

Figura 12 – Creación de Compartment (Fuente: Consola OCI)

2.2 Creación de Clúster Kubernetes

Para la creación de nuestro Cluster Kubernetes seleccionaremos la opción del menú y procederemos a seleccionar primero Developer Services y en el menú derecho que será desplegado seleccionaremos la opción “Kubernetes Cluster (OKE)” como se muestra en la Figura 13.

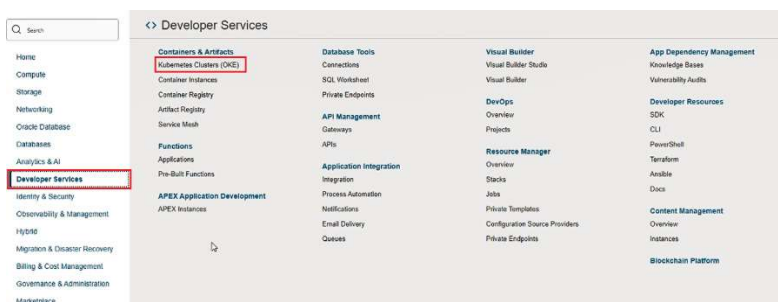


Figura 13 – Creación de Compartment (Fuente: Consola OCI)

Una vez desplegada la ventana de creación del Cluster OKE presionamos la opción de “Create cluster” donde los desplegará un Pop Up el cual nos dará dos opciones para crear nuestro Cluster como se muestra en la Figura 14.

- **Quick create** nos desplegara toda la infraestructura necesaria para poder desplegar en forma correcta el cluster.
- **Custom Create** al optar por esta opción de creación deberá tener presente que antes de crear el cluster se tendrá que crear primero las Virtual Cloud Network (VCN), Internet Gateway (IG), NAT Gateway (NG), Service Gateway (SGW).

Nosotros por motivos exclusivamente del proyecto optaremos por crear nuestro cluster por la opción “Quick Create” ya que no tenemos otros recursos desplegados.

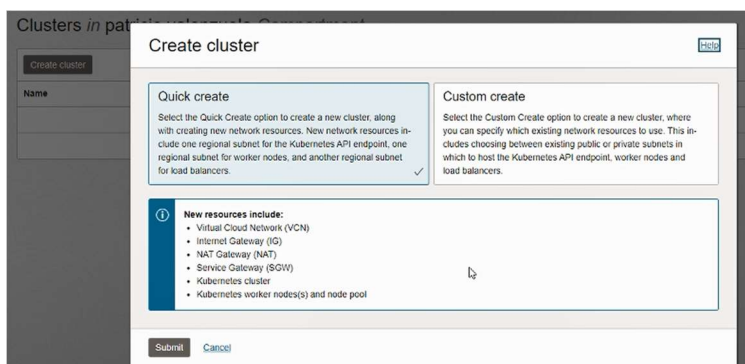


Figura 14 – Opciones de creación de Clúster Kubernetes (Fuente: Consola OCI)

Una vez seleccionada la opción de creación del Clúster Kubernetes desplegara una ventana como se muestra en la Figura 15 en la cual seleccionaremos las opciones de creación del Clúster.

Los parámetros que deberemos ingresar son los siguientes:

- **Name** es el nombre que le daremos al Clúster Kubernetes.
- **Compartment** es donde desplegaremos el Clúster Kubernetes.
- **Kubernetes Versión** en la cual siempre se desplegará la última versión disponible en la nube.
- **Kubernetes API Endpoint** será definido como Public por las necesidades de que deberá salir hacia el exterior.
- **Node Type** seleccionaremos Managed ya que puede ser posible que se requiera realizar algún cambio, el Virtual es fijo y no puede ser modificado.
- **Kubernetes worker nodes** seleccionaremos Private workers ya que el alcance del proyecto solo se requiere administrar las aplicaciones las cuales desplegaremos.

Create cluster (quick)

1 Create cluster
2 Review

Name:

Compartment:

Kubernetes version:

Kubernetes API endpoint:
Private endpoint
The Kubernetes cluster that is created will be hosted on a private subnet.
Public endpoint
The Kubernetes cluster that is created will be hosted on a public subnet with a public IP address auto-assigned. ✓

Node type:
Managed
The Kubernetes worker nodes are provisioned compute instances in your tenancy. You are charged for the core-hours those instances use.
Virtual
The Kubernetes worker nodes are virtual. The resources to execute your Kubernetes pods are provisioned dynamically as needed. You are charged for the resources used only.

Kubernetes worker nodes:
Private workers
The Kubernetes worker nodes that are created will be hosted in a private subnet. ✓
Public workers
The Kubernetes worker nodes that are created will be hosted in a public subnet.

Figura 15 – Creación de Clúster Kubernetes (Fuente: Consola OCI)

Para la configuración del Shape e Imagen que utilizaremos vamos a desplegar para el Clúster Kubernetes que tendrá 3 OCPUs y 32 GB de Memoria y se compondrá por 3 nodos tal como se muestra en la Figura 16, lo recomendable en estos casos es utilizar una imagen Linux para la base de Sistema operativo del Clúster Kubernetes ya que si utilizamos por ejemplo Windows que está disponible en la nube se requiere cancelar los costos de licenciamiento.

Shape and image

A [shape](#) is a template that determines the number of CPUs, amount of memory, and other resources allocated to an instance. The image is the operating system that runs on top of the shape.

Pod shape ⓘ

VM.Standard.E3.Flex

The compute service limit applies to pods. Check the service limits.

You can customize the number of OCPUs that are allocated to a flexible shape. The other resources scale proportionately. [Learn more about flexible shapes.](#)

Select the number of OCPUs

1 28 57 84 114

Amount of memory (GB) ⓘ

1 444 888 1332 1776

Network Bandwidth (Gbps): 3.0 ⓘ

Max. Total VNics: 3 ⓘ

Image

ORACLE Linux
Oracle Linux 8
Image built: 2023.06.30-0
Kubernetes version: 1.26.2

Change image

Node count ⓘ

Figura 16 – Creación de Clúster Kubernetes (Fuente: Consola OCI)

Finalmente configuraremos las llaves de SSH para poder conectarnos al clúster, para ello seleccionaremos la opción “Generate SSH key pair” y salvaremos ambas llaves “private” y “public” y la guardaremos en nuestro disco local tal como se muestra en la Figura 17.

Figura 17 – Create SSH Key Private y Public (Fuente: Consola OCI)

Posteriormente seleccionaremos la opción “next” y desplegará el resumen de despliegue del “Clúster Kubernetes” y seleccionaremos la opción “Create Clúster”, una vez que el clúster sea creado lo desplegará como un recurso disponible.

2.3 Test creación de Clúster Kubernetes

Para realizarlas pruebas de una correcta creación de un Cluster Kubernetes se debe conectar a la consola de OCI como se puede ver en la Figura 18 y los nodos asociados al clúster Kubernetes en la Figura 19 que esta como evidencia de su despliegue.

Clusters in Patricio.Valenzuela *Compartment*

Create cluster

Name	Status	Node pools ⓘ	VCN	Version	Created
OKE1	Active	1	oke-vcn-quick-OKE1-63d742893	v1.26.2	Tue, Jul 11, 2023, 16:13:26 UTC

Showing 1 item < 1 of 1 >

Figura 18 – Clúster Kubernetes desplegado en consola OCI (Fuente: Consola OCI)

Create instance Table settings

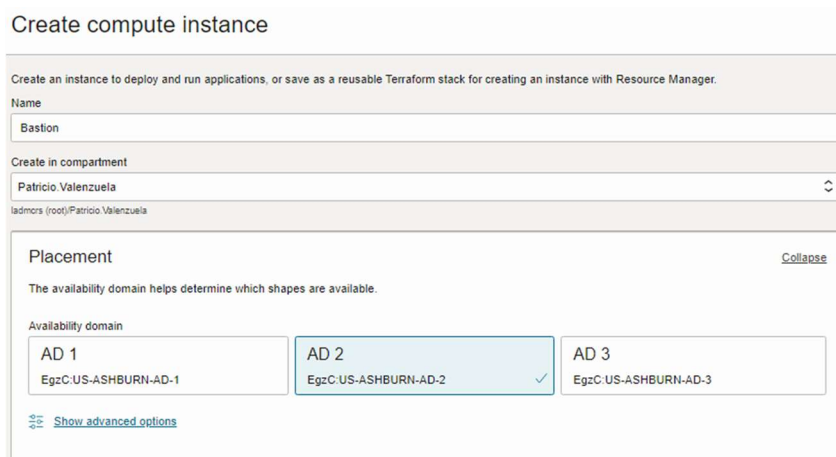
Name	State	Public IP	Private IP	Shape	OCPU count	Memory (GB)	Availability domain	Fault domain	Created
oke-czk2q5b8fa-n5jcb6o3la-ses3ucn7oza-2	Stopped	-	10.0.10.236	VM.Standard.E...	3	32	AD-1	FD-3	Tue, Jul 11, 202...
oke-czk2q5b8fa-n5jcb6o3la-ses3ucn7oza-1	Stopped	-	10.0.10.16	VM.Standard.E...	3	32	AD-2	FD-2	Tue, Jul 11, 202...
oke-czk2q5b8fa-n5jcb6o3la-ses3ucn7oza-0	Stopped	-	10.0.10.233	VM.Standard.E...	3	32	AD-3	FD-1	Tue, Jul 11, 202...

1 / 1

Figura 19 – Nodos asociados al Clúster Kubernetes (Fuente: Consola OCI)

2.4 Creación del Bastión

Para la creación del Bastión procederemos seleccionando en el menu las opciones “Compute” y posteriormente “Instances” para posteriormente proceder a crear una nueva instancia como se muestra en la Figura 20, cabe señalar que por defecto OCI crea todos los nuevos recursos en el Availability Domain 1 y por este motivo lo crearemos Availability Domain 2.



Create compute instance

Create an instance to deploy and run applications, or save as a reusable Terraform stack for creating an instance with Resource Manager.

Name
Bastion

Create in compartment
Patricio Valenzuela

Placement [Collapse](#)

The availability domain helps determine which shapes are available.

Availability domain

AD 1 EgZC-US-ASHBURN-AD-1	AD 2 EgZC-US-ASHBURN-AD-2 ✓	AD 3 EgZC-US-ASHBURN-AD-3
------------------------------	--------------------------------	------------------------------

[Show advanced options](#)

Figura 20 – Creación de Bastión (Fuente: Consola OCI)

En la creación del Bastión y para que pueda comunicarse con el clúster Kubernetes deberemos configurar la subnet o subred a la cual estará asociado el Bastión, pero primero explicaremos cuales son las subredes que están definidas en la creación del Clúster Kubernetes como se muestra en la Figura 21.

Create Subnet				
Name	State	IPv4 CIDR Block	IPv6 Prefixes	Subnet Access
oke-k8sApiEndpoint-subnet-quick-OKE1-63d742893-regional	Available	10.0.0.0/28	-	Public (Regional)
oke-nodesubnet-quick-OKE1-63d742893-regional	Available	10.0.10.0/24	-	Private (Regional)
oke-svcbsubnet-quick-OKE1-63d742893-regional	Available	10.0.20.0/24	-	Public (Regional)

Figura 21 – Subredes del Cluster Kubernetes (Fuente: Consola OCI)

Subredes

- Esta subred que es **oke-nodesubnet-quick-OKE1-63d742893-regional** (Private) es por donde comunican los nodos del cluster para saber su estado de salud.
- Esta subred que es **oke-svcbsubnet-quick-OKE1-63d742893-regional** (Public) es utilizada por los balanceadores de carga del cluster kubernetes.
- Esta subred que es **oke-k8sApiEndpoint-subnet-quick-OKE1-63d742893-regional** (Public) es utilizada por los APIS y donde definiremos nuestra subred para nuestro Bastión.

Entonces retomando la configuración de redes en la opción de subred deberemos seleccionar la red pública asociada a la ApiEndpoint.

Nos queda la ultima parte de la configuración del Bastión donde asociaremos las SSH Keys que fueron salvadas y almacenadas en nuestro disco local del capítulo “2.2 Creación del Clúster Kubernetes” y como se muestra en la Figura 22 y posteriormente el ultimo paso es presionar el botón Create.

Figura 22 – SSH keys asociar al Bastión (Fuente: Consola OCI)

Importante es mencionar que solo se deberá modificar las opciones señaladas y las demás deberán dejarse con los valores que vienen por defecto.

Debemos crear una regla de ingreso (Ingress Rules) para el puerto 22 en la subred donde reside esta instancia de cómputo como se observa en la Figura 23.

Figura 23–Regla de acceso al puerto 22 desde fuera de OCI (Fuente: Consola OCI)

Como punto final habilitaremos la salida hacia Internet para poder instalar los paquetes que serán requeridos en el capítulo “2.5 Conectarse al Clúster Kubernetes desde el Bastión” como se muestra en la Figura 24.

Recorrimos las siguientes opciones de menú para la configuración.

- Menu / Compute / Instance
- Seleccionamos el Bastión / Seleccionamos la Subnet
- Seleccionamos la Subnet definida / Seleccionamos Egress Rules
- Creamos la nueva regla.

Figura 24 –Regla de egreso hacia Internet (Fuente: Consola OCI)

2.5 Test de creación del Bastión

Para validar la correcta creación del Bastión en nuestro tenant deberemos navegar por el menú de la consola “Menu/Instances” y podremos visualizar las instancias desplegadas como se muestra en la Figura 25 donde se encuentra la evidencia de la creación del Bastión como un recurso desplegado en nuestro tenant de OCI.

Instances in patricio.valenzuela Compartment

An instance is a compute host. Choose between virtual machines (VMs) and bare metal instances. The image that you use to launch an instance determines its operating system and other software.

Create instance Table settings

Name	State	Public IP	Private IP	Shape	OCPU count	Memory (GB)	Availability domain	Fault domain	Created
oke-czkg2q5b6fq-n5jcbq6g3lq-ses3ucn7oza-2	Running	-	10.0.10.236	VM.Standard.E3...	3	32	AD-1	FD-3	Tue, Jul 11, 2023...
oke-czkg2q5b6fq-n5jcbq6g3lq-ses3ucn7oza-1	Running	-	10.0.10.16	VM.Standard.E3...	3	32	AD-2	FD-2	Tue, Jul 11, 2023...
Bastion	Running	129.213.197.112	10.0.0.9	VM.Standard.E4...	1	16	AD-2	FD-3	Tue, Jul 11, 2023...
oke-czkg2q5b6fq-n5jcbq6g3lq-ses3ucn7oza-0	Running	-	10.0.10.233	VM.Standard.E3...	3	32	AD-3	FD-1	Tue, Jul 11, 2023...

« < 1/1 > »

Figura 25 –Bastión desplegado como recurso en OCI (Fuente: Consola OCI)

2.6 Conectarse al Bastión desde una red On Premise

Para poder realizar la conexión desde una laptop que se encuentre fuera de la red del Clúster Kubernetes y el Bastión que fue desplegado en la nube OCI ejecutaremos los siguientes pasos.

Lo primero que deberemos hacer es asegurarnos que tenemos instaladas las herramientas Putty y PuttyGen las cuales utilizaremos para realizar nuestra configuración.

Utilizaremos PuttyGen para poder realizar el cambio de formato de SSH Key generada en OCI y puedan ser leídas correctamente por Putty para poder establecer la comunicación, para ello utilizaremos el SSH Key private. Para ello en PuttyGen seleccionaremos la siguiente secuencia de ejecución.

- Conversions
- Import key
- Seleccionamos el SSH Key private
- Presionamos “Save private key”

Ahora se procede a la configuración de Putty para poder acceder al Bastión y así poder conectarse al Clúster de OKE, para ello utilizaremos la dirección IP pública del Bastión.

El primer paso es asociar la SSH Key private que generamos en el paso anterior a Putty como se observa en la Figura 26.

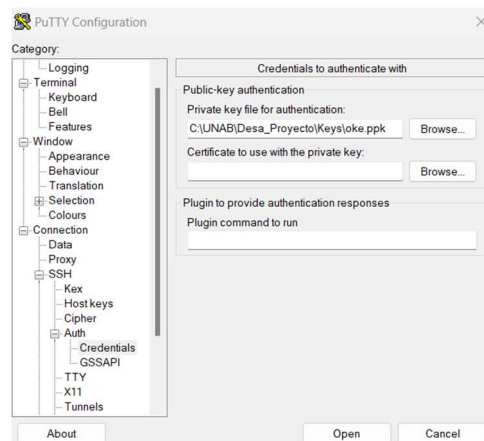


Figura 26 –Asociar SSH Key Private a Putty (Fuente: Local)

2.7 Test de conexión al Bastión desde una red On Premise

Para realizar el test de conexión lo único que se requiere posterior a la configuración realizada anteriormente es abrir una sesión de Putty y conectar a la dirección publica del Bastión como se ve en la Figura 27.

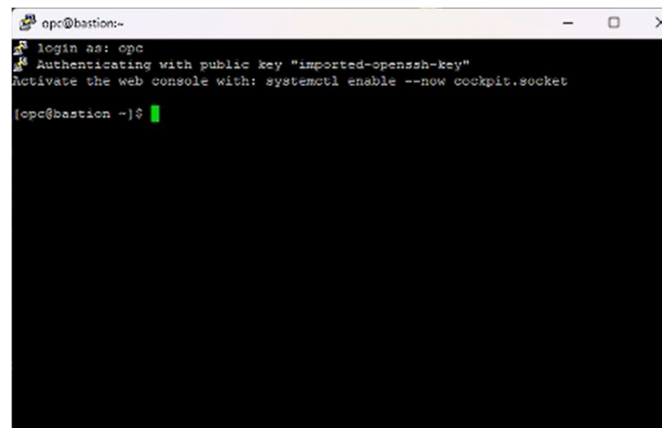
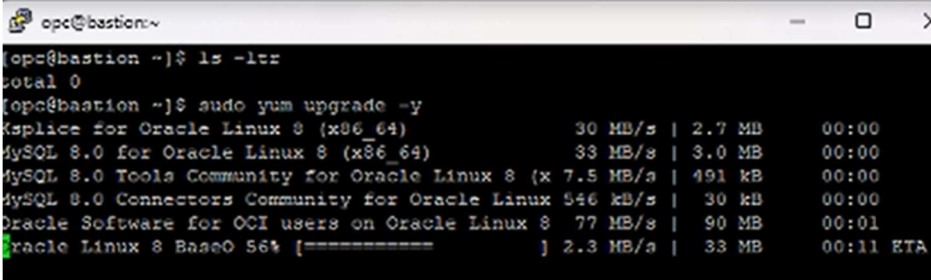


Figura 27 –Acceso a Putty desde On Premise con Putty (Fuente: Local)

2.8 Actualización de Sistema Operativo del Bastión

Para ejecutar la actualización del sistema operativo y todas sus librerías del Bastión solo deberemos ejecutar el comando “sudo yum upgrade -y” tal como se visualiza en la Figura 28.



```
opc@bastion ~]$ ls -ltr
total 0
opc@bastion ~]$ sudo yum upgrade -y
ksplice for Oracle Linux 8 (x86_64) 30 MB/s | 2.7 MB 00:00
MySQL 8.0 for Oracle Linux 8 (x86_64) 33 MB/s | 3.0 MB 00:00
MySQL 8.0 Tools Community for Oracle Linux 8 (x 7.5 MB/s | 491 kB 00:00
MySQL 8.0 Connectors Community for Oracle Linux 546 kB/s | 30 kB 00:00
Oracle Software for OCI users on Oracle Linux 8 77 MB/s | 90 MB 00:01
Oracle Linux 8 Base OS 56% [=====] 2.3 MB/s | 33 MB 00:11 ETA
```

Figura 28 –Actualización de Sistema Operativo del Bastión (Fuente: Local)

2.9 Conectarse al Clúster Kubernetes desde el Bastión

Para poder conectarse al Clúster Kubernetes podemos realizarlo de dos formas las cuales son las siguientes:

- Cloud Shell Access, acceso desde la consola.
- Local Access, utilizando Putty que se conectara la Bastión previamente configurado en el capítulo 2.4.

Para este proyecto definiremos el acceso que utilizaremos es el de “Local Access” y toda su configuración la realizaremos desde Putty conectado al Bastión y para configurar ejecutaremos los siguientes pasos:

Instalar OCI CLI en el Bastión.

- Primero instalaremos OCI CLI basado en Python con el siguiente comando.
 - sudo dnf install python36-oci-cli
- Como segundo paso debemos instalar KUBECTL y para ello nos basaremos en la documentación oficial la que se encuentra en la URL <https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/> y los comandos serían los siguientes:
 - curl -LO "https://dl.k8s.io/release/\$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
 - sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

Posteriormente se ejecutarán los comandos para lograr tener acceso al Cluster Kubernetes.

- oci -v
- mkdir -p \$HOME/.kube
- oci ce cluster create-kubeconfig --cluster-id ocid1.cluster.oc1.iad.aaaaaaaadgcwr6msiif6d7kbid57xzhraaxlkt4wawwle3bb6czkj2g5b6fq --file \$HOME/.kube/config --region us-ashburn-1 --token-version 2.0.0 --kube-endpoint PUBLIC_ENDPOINT
- export KUBECONFIG=\$HOME/.kube/config

2.10 Test de conexión al Clúster Kubernetes desde el Bastión

Para realizar este test deberemos ejecutar el comando “kubectl get nodes” con lo cual debiera traernos la cantidad total de nodos creados en nuestro Cluster Kubernetes como se refleja en la figura 29 y que se encuentra como evidencia.

```
[opc@bastion .oci]$ kubectl get nodes
NAME                STATUS    ROLES    AGE   VERSION
10.0.10.16          Ready    node     3d3h   v1.26.2
10.0.10.233         Ready    node     3d3h   v1.26.2
10.0.10.236         Ready    node     3d3h   v1.26.2
[opc@bastion .oci]$
```

Figura 29 –Conexión desde el Bastión al Clúster Kubernetes (Fuente: Local)

2.11 Instalación de Ansible

Para poder realizar la comunicación entre servidores que serán monitoreados y poder realizar la extracción de las métricas se debe instalar Ansible y podemos realizarlo de dos formas las cuales son las siguientes:

Adjunto encontraremos el paso a paso de como instalar Ansible.

Validamos la versión correcta de Ansible que debe ser instalada ejecutando el comando “sudo yum search ansible” como se muestra en la figura 30.

```
opc@bastion:~$
[opc@bastion ~]$
[opc@bastion ~]$ sudo yum search ansible
Last metadata expiration check: 6 days, 20:41:33 ago on Fri 14 Jul 2023 06:59:16 PM GMT.
===== Name Exactly Matched: ansible =====
ansible.noarch : SSH-based configuration management, deployment, and task execution system
ansible.src : SSH-based configuration management, deployment, and task execution system
===== Name & Summary Matched: ansible =====
ansible-freeipa-tests.noarch : ansible-freeipa tests
ansible-pcp.noarch : Ansible Metric collection for Performance Co-Pilot
ansible-pcp.src : Ansible Metric collection for Performance Co-Pilot
ansible-test.x86_64 : Tool for testing ansible plugin and module code
===== Name Matched: ansible =====
ansible-core.src : SSH-based configuration management, deployment, and task execution system
ansible-core.x86_64 : SSH-based configuration management, deployment, and task execution system
ansible-freeipa.noarch : Roles and playbooks to deploy FreeIPA servers, replicas and clients
ansible-freeipa.src : Roles and playbooks to deploy FreeIPA servers, replicas and clients
[opc@bastion ~]$
```

Figura 30 – Validación versión de Ansible a Instalar (Fuente: Local)

Posterior a esta validación se procede a instalar el paquete ansible-core-x86_64 con el comando “sudo yum install -y ansible-core” como se muestra en la figura 31.

```
[opc@bastion ~]$ sudo yum install -y ansible-core
Ksplice for Oracle Linux 8 (x86_64) 145 kB/s | 3.0 kB 00:00
Ksplice for Oracle Linux 8 (x86_64) 30 MB/s | 2.7 MB 00:00
MySQL 8.0 for Oracle Linux 8 (x86_64) 142 kB/s | 3.0 kB 00:00
MySQL 8.0 Tools Community for Oracle Linux 8 (x86_64) 144 kB/s | 2.9 kB 00:00
MySQL 8.0 Connectors Community for Oracle Linux 8 (x86_64) 167 kB/s | 2.9 kB 00:00
Oracle Software for OCI users on Oracle Linux 8 (x86_64) 180 kB/s | 2.9 kB 00:00
Oracle Software for OCI users on Oracle Linux 8 (x86_64) 77 MB/s | 91 MB 00:01
Oracle Linux 8 BaseOS Latest (x86_64) 175 kB/s | 3.6 kB 00:00
Oracle Linux 8 BaseOS Latest (x86_64) 75 MB/s | 60 MB 00:00
Oracle Linux 8 Application Stream (x86_64) 159 kB/s | 3.9 kB 00:00
Oracle Linux 8 Application Stream (x86_64) 74 MB/s | 47 MB 00:00
[opc@bastion ~]$
```

Figura 31 – Instalación de Ansible (Fuente: Local)

2.12 Instalación de Podman

Podman es un runtime para administrar los contenedores que se utiliza para ejecutar, crear, etcétera, y puede ser ejecutado por cualquier usuario donde sea instalado y porque no utilizamos Dockers que es el runtime por default y más conocido ya que para ejecutarlo se requiere las credenciales de root y esto representa una brecha de seguridad. La instalación la podemos validar en la figura 32 la cual se encuentra a continuación.

```
[opc@bastion ~]$ sudo yum install -y podman
Last metadata expiration check: 0:03:37 ago on Fri 21 Jul 2023 03:42:27 PM GMT.
Dependencies resolved.

```

Package	Arch	Version	Repository	Size
Installing:				
podman	x86_64	3:4.4.1-12.module+el8.8.0+21056+d98a0860	ol8_appstream	15 M
Installing dependencies:				
common	x86_64	3:2.1.6-1.module+el8.8.0+21045+adcb6a64	ol8_appstream	57 k
container-selinux	noarch	2:2.205.0-2.module+el8.8.0+21045+adcb6a64	ol8_appstream	64 k
container-networking-plugins	x86_64	1:1.2.0-1.module+el8.8.0+21045+adcb6a64	ol8_appstream	20 M
containers-common	x86_64	2:1-64.0.1.module+el8.8.0+21056+d98a0860	ol8_appstream	88 k
criu	x86_64	3.15-3.module+el8.8.0+21045+adcb6a64	ol8_appstream	510 k
fuse-overlayfs	x86_64	1.11-1.module+el8.8.0+21056+d98a0860	ol8_appstream	74 k
libnet	x86_64	1.1.6-15.el8	ol8_appstream	67 k
libslirp	x86_64	4.4.0-1.module+el8.8.0+21045+adcb6a64	ol8_appstream	70 k
podman-catatonit	x86_64	3:4.4.1-12.module+el8.8.0+21056+d98a0860	ol8_appstream	362 k
runl	x86_64	1:1.1.4-1.0.1.module+el8.8.0+21119+51r68ed8	ol8_appstream	3.1 M
shadow-utils-subid	x86_64	2:4.6-17.el8	ol8_baseos_latest	113 k
slirp4netns	x86_64	1.2.0-2.module+el8.8.0+21045+adcb6a64	ol8_appstream	54 k
Enabling module streams:				
container-tools		ol8		
Transaction Summary				
Install 13 Packages				
Total download size: 39 M				
Installed size: 119 M				
Downloading Packages:				
(1/13): common-2.1.6-1.module+el8.8.0+21045+adcb6a64.x86_64.rpm 1.9 MB/s 57 kB 00:00				
(2/13): container-selinux-2.205.0-2.module+el8.8.0+21045+adcb6a64.noarch.rpm 2.0 MB/s 64 kB 00:00				

Figura 32 – Instalación de Podman (Fuente: Local)

2.13 Creación y configuración de Registry Local

Para poder realizar el despliegue del recurso de Registry deberemos realizar las siguientes operaciones.

- Seleccionar la opción Menú en OCI
- Seleccionar la opción Developer Services
- Seleccionar la opción Container Registry

Al realizar estas selecciones se desplegará la siguiente ventana de configuración como se muestra en la figura 33 a continuación.



Figura 33 – Configuración de Registry (Fuente: Consola OCI)

Seleccionamos la opción “Create Repository” en la ventana recién desplegada el cual nos desplegara una nueva ventana donde ingresaremos el nombre de la aplicación como nombre del repositorio y presionamos la opción create. La validación de la correcta creación y configuración del repositorio la encontraremos en la figura 34.

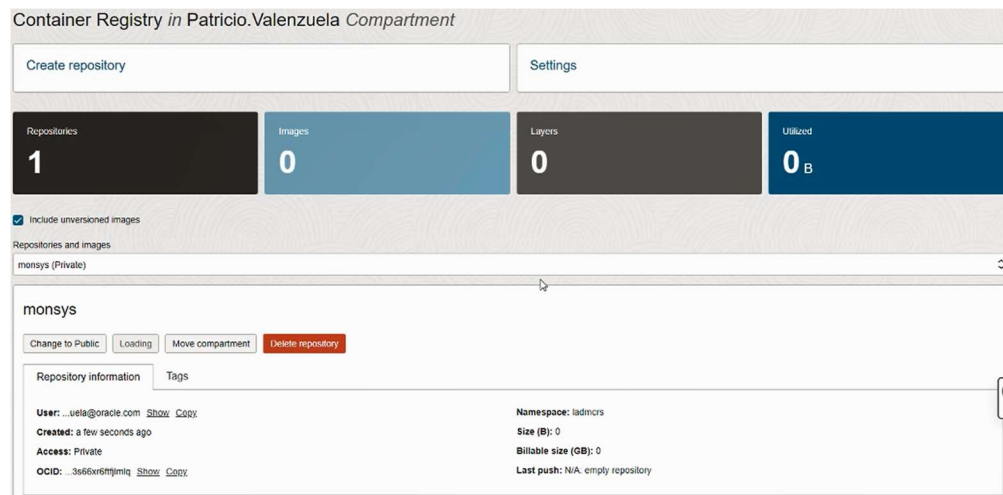


Figura 34 –Registry creado (Fuente: Consola OCI)

2.14 Creación secrets en Kubernetes

La configuración de secrets hacia el Clúster Kubernetes corresponde a la configuración que nos permitirá conectar el Registry hacia el Clúster Kubernetes.

Lo primero que debemos realizar es validar que lleguemos a los nodos del Clúster Kubernetes tal como se puede ver en la figura 35 con el comando “kubectl get nodes”. Cabe señalar que los Clúster Kubernetes trabajan con Namespaces que corresponden a separaciones lógicas para los proyectos desplegados.

```
[opc@bastion ~]$ kubectl get nodes
NAME                STATUS    ROLES    AGE   VERSION
10.0.10.16          Ready    node     10d   v1.26.2
10.0.10.233         Ready    node     10d   v1.26.2
10.0.10.236         Ready    node     10d   v1.26.2
[opc@bastion ~]$
```

Figura 35 –Acceso a los nodos del Clúster Kubernetes (Fuente: Local)

En segunda instancia validaremos los namespace con el comando “kubectl get ns” que se encuentran definidos en nuestro Clúster Kubernetes como se puede apreciar en la figura 36.

```
[opc@bastion ~]$ kubectl get ns
NAME                STATUS    AGE
default             Active    10d
kube-node-lease     Active    10d
kube-public         Active    10d
kube-system         Active    10d
[opc@bastion ~]$
```

Figura 36 –Validación de namespace en el Clúster Kubernetes (Fuente: Local)

- default donde desplegaremos nuestra aplicación.
- kube-node-lease donde residen todas las configuraciones de los nodos para el Clúster de Kubernetes.
- kube-public donde residen todas las librerías y los protocolos de conexiones de las APIs.
- kube-system donde residen todas las configuraciones de Kubernetes.

El siguiente paso es crear nuestro namespace donde desplegaremos nuestra aplicación de monitoreo con el comando “kubectl create ns monsys” y como se puede observar en la figura 37.

```
[opc@bastion ~]$ kubectl create ns monsys
namespace/monsys created
[opc@bastion ~]$
[opc@bastion ~]$
[opc@bastion ~]$ kubectl get ns
NAME                STATUS   AGE
default             Active  10d
kube-node-lease     Active  10d
kube-public         Active  10d
kube-system         Active  10d
monsys              Active  7s
```

Figura 37 – Creación de namespace en el Clúster Kubernetes (Fuente: Local)

Ahora se requiere crear un Token para hacer posible la configuración y para ello lo que debemos realizar es los siguientes pasos:

- Seleccionar la opción Menú
- Seleccionar la opción Identity & Security
- Seleccionar la opción users
- Seleccionar en el menú central el usuario con el cual se creará el Token.
- Seleccionamos la opción Auth Token
- Seleccionamos la opción “Generate Token”
- Ingresamos la descripción
- Seleccionamos la opción “Generate Token”
- Cabe destacar que antes de cerrar la ventana de creación del Token se deberá copiar el string del Token porque será utilizada mas adelante.

Una vez finalizada la creación se puede validar cotejándolo con la figura 38 adjunta.

Auth Tokens	
<button>Generate Token</button>	
Description	Created
Token-Registry	Fri, Jul 21, 2023, 19:13:37 UTC
Displaying 1 Token	

Figura 38 – Creación de Token (Fuente: Consola OCI)

Para la creación del secret ejecutaremos el siguiente comando y verificar su correcta finalización en la figura 39.

- kubectl create secret docker-registry ociregistry --docker-server=iad.ocir.io --docker-username='idlhjo6dp3bd/oracleidentitycloudservice/patricio.valenzuela@oracle.com' --docker-password='0mbZ_O9m_)lS1_CsDIR3' -n monsys

```
[opc@bastion ~]$ kubectl create secret docker-registry ociregistry --docker-server=iad.ocir.io --docker-username='idlhjo6dp3bd/oracleidentitycloudservice/patricio.valenzuela@oracle.com' --docker-password='0mbZ_O9m_)lS1_CsDIR3' -n monsys
secret/ociregistry created
```

Figura 39– Creación de Secret (Fuente: Consola OCI)

2.15 Test de creació secrets en Kubernetes

Para ejecutar el test de la correcta creación del secrets para nuestro Kubernetes deberá ejecutarse el comando “kubectl get secret ociregistry -n monsys -o yaml” como se visualiza en la figura 38 que esta como evidencia.

```
[ops@bastion ~]$ kubectl get secret ocaregistry -n monsys -o yaml
apiVersion: v1
data:
  .dockerconfigjson: eyJhdXNjbG9ycyI6eyJpYQWwZDZnci1pbyI6eyJlc2VybmFtZSI6ImlkbgHqbszKcDwiZC9vcmlkLmVpY2VudGVudGloOeWNsbjVkc2VydmlkLjZS9wYXRyaWNBpbpyY2Yxwlbnp1ZCxxhQG9yYWNsZS5jb201LCJvYXNpdDoySi9CI6IjBtPlpTf2tlXylsUcFQ3NEVSitiw1XXV0aCI6ImFXUmhR3BS2tmd300y8mtUkl1SWVudG9icXBGeSvZUwYShVNhXkyeH2KVlJGxW1hKdMFXtmdd300ZeHrKfkybH2ldlpYkdWdWVuVmxi1R0ZSYnKaFkyeGxbUS2Y1Rvd2JXSsmFYMDglY1Y1cG9JTXhhYME56UkVaU013PT0iX19
kind: Secret
metadata:
  creationTimestamp: "2023-07-21T19:18:06Z"
name: ocaregistry
namespace: monsys
resourceVersion: "3907968"
uid: 5d697906-0500-40f9-0aa-fef03b70e028
type: kubernetes.io/dockerconfigjson
[ops@bastion ~]$
```

Figura 38 – Evidencia de Creación de Secret (Fuente: Consola OCI)

2.16 Creación de Servidores a monitorear

Se procederá a la creación de los 3 servidores que serán monitoreados por nuestra aplicación, y para la creación de estos servidores procederemos seleccionando en el menu las opciones “Compute” y posteriormente “Instances” para posteriormente proceder a crear una nueva instancia como se muestra en la Figura 39, cabe señalar que por defecto OCI crea todos los nuevos recursos en el Availability Domain 1 y por este motivo lo crearemos Availability Domain 2.

Create compute instance

Create an instance to deploy and run applications, or save as a reusable Terraform stack for creating an instance with Resource Manager.

Name

Bastion

Create in compartment

Patricio Valenzuela

admins (root)/Patricio Valenzuela

Placement [Collapse](#)

The availability domain helps determine which shapes are available.

Availability domain

AD 1 Egzc-US-ASHBURN-AD-1	AD 2 Egzc-US-ASHBURN-AD-2 ✓	AD 3 Egzc-US-ASHBURN-AD-3
------------------------------	--------------------------------	------------------------------

[Show advanced options](#)

Figura 39 – Creación de Bastión (Fuente: Consola OCI)

En la creación de estos servidores utilizaremos la subnet o subred a la cual estará asociado el Bastión que corresponde a la subred ApiEndpoint. Posteriormente asociaremos nuestra SSH Keys que utilizamos en la creación de nuestro Bastión como puede visualizarse en la figura 40.

Add SSH keys

Generate an [SSH key pair](#) to connect to the instance using a Secure Shell (SSH) connection, or upload a public key that you already have.

☐ Generate a key pair for me ☒ Upload public key files (.pub) ☐ Paste public keys ☐ No SSH keys

SSH public keys

 Drop .pub files here. [Browse](#)

ssh-key-2023-07-11_key.pub x

Figura 40 – SSH keys asociar al Bastión (Fuente: Consola OCI)

La única modificación que deberá realizarse en la configuración por default al crear nuestros servidores es en la sección de Image and Shape como se ve en la figura 41.

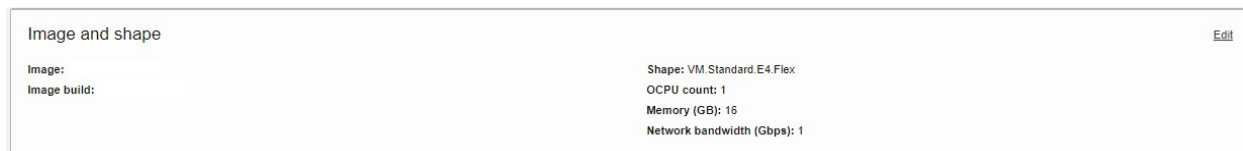


Figura 41 – Selección de Imagen o Shape (Fuente: Consola OCI)

Por cada uno de los servidores se seleccionará los distintos sistemas operativos que utilizaremos como muestra la figura 42.

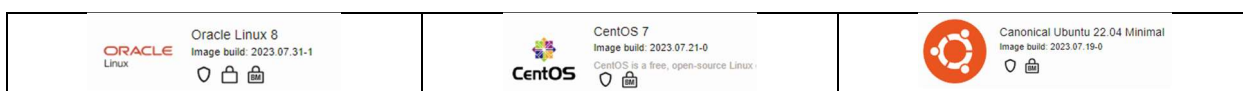


Figura 42 – Imágenes de sistema operativo utilizados para servidores a monitorear (Fuente: Consola OCI)

2.17 Instalación de MySQL

Ahora procederemos con la instalación de la instancia de Base de Datos MySQL la cual para poder realizar la instalación nos conectaremos al servidor y ejecutaremos el comando “sudo yum -y install mysql” como se puede ver en la figura 43.

```
opc@bastion~$ login as: opc
Authenticating with public key "imported-openssh-key"
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Mon Aug 7 23:44:27 2023 from 181.43.123.188
[opc@bastion ~]$ sudo yum -y install mysql
Kssplice for Oracle Linux 8 (x86_64) 1.1 kB/s | 3.0 kB 00:02
Kssplice for Oracle Linux 8 (x86_64) 70 kB/s | 3.0 MB 00:43
MySQL 8.0 for Oracle Linux 8 (x86_64) 907 B/s | 3.0 kB 00:03
MySQL 8.0 Tools Community for Oracle Linux 8 (x86_64) 659 B/s | 2.9 kB 00:04
MySQL 8.0 Connectors Community for Oracle Linux 8 (x86_64) 120 kB/s | 2.9 kB 00:00
Oracle Software for OCI users on Oracle Linux 8 (x86_64) 1.3 kB/s | 2.9 kB 00:02
Oracle Software for OCI users on Oracle Linux 8 (x86_64) 12 MB/s | 93 MB 00:07
Oracle Linux 8 BaseOS Latest (x86_64) 437 B/s | 3.6 kB 00:08
Oracle Linux 8 BaseOS Latest (x86_64) 1.9 MB/s | 62 MB 00:31
Oracle Linux 8 Application Stream (x86_64) 1.2 kB/s | 3.9 kB 00:03
Oracle Linux 8 Application Stream (x86_64) 1.4 MB/s | 48 MB 00:34
Oracle Linux 8 Addons (x86_64) 2.5 kB/s | 3.0 kB 00:01
Oracle Linux 8 Addons (x86_64) 408 kB/s | 6.9 MB 00:17
Latest Unbreakable Enterprise Kernel Release 7 for Oracle Linux 287 B/s | 3.0 kB 00:10
Package mysql-8.0.32-1.module+el8.8.0+21055+76bd398b.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[opc@bastion ~]$
```

Figura 43 – Instalación MySQL (Fuente: Servidor OCI)

2.18 Administración y configuración de usuario MySQL

A continuación de la correcta instalación de la base de datos MySQL deberemos configurar la contraseña del usuario de administración.

Lo siguiente a realizar es configurar la contraseña del usuario root para la nueva instancia de base de datos MySQL con el siguiente comando el cual servirá la primera y única vez:

- `mysqladmin -u root -h localhost password "password_new"`

Posteriormente debemos ingresar a la consola MySQL y ejecutar el siguiente comando:

```
mysql> update mysql.user set password=PASSWORD('password_new') where user='root';
```

Otra manera de cambiar la contraseña al usuario root es la siguiente:

```
mysql> set password for root@localhost=password('password_new');
```

Para poder modificar la contraseña en el caso que sea requerido desde línea de comandos unix deberán realizarlo con el siguiente comando:

- `mysqladmin -u root -h localhost -p password "password_new"`

y a continuación solicitará la nueva contraseña para el usuario root.

2.19 Prueba de MySQL

```
[opc@bastion ~]$  
[opc@bastion ~]$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 9  
Server version: 8.0.32 Source distribution  
  
Copyright (c) 2000, 2023, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql>  
mysql> █
```

Figura 44 – Prueba de conexión MySQL (Fuente: Servidor OCI)

3. Creación y Configuración de Git y GitHub

3.1 Crear repositorio GitHub

Para realizar esta configuración primero deberemos crear nuestro repositorio en GitHub y para ello realizaremos los siguientes pasos.

- Conectarse en un browser de preferencia a la URL www.github.com
- Presionamos el botón arriba a mano derecha que indica “Sign up”
- Ingresamos el mail de estudiante en este caso por ser un repositorio para los ramos de Seminario de Licenciatura y Proyecto de Titulo
- Ingresamos la contraseña que será la autenticación para nuestro repositorio.
- Resolver el CAPTCHA, al resolverlo le enviara un correo con el código de autenticación.
- El código enviado al correo ingresado en paso anterior ingresarlo para la autenticación.
- GitHub solicitará el tipo de cuenta a crear en el cual deberá seleccionar “just me” y “Student”
- En el siguiente punto solicitará el tipo de funcionalidades a utilizar y solo deberá seleccionar “continue”
- Posteriormente deberá seleccionar el tipo de repositorio a crear donde seleccionaremos “Continue for free”
- A continuación, a mano izquierda seleccionaremos la opción “Create repository”
- Ingresamos toda la información solicitada como por ejemplo el nombre del repositorio, una breve descripción, seleccionamos que el repositorio sea público, que contenga un archivo de readme y finalmente presionamos crear repositorio como se puede observar en la figura 45.

Figura 45 – Creación de repositorio GitHub (Fuente: GitHub.com)

3.2 Prueba de repositorio GitHub

Para poder realizar la prueba de creación de nuestro repositorio ingresaremos a la URL “www.github.com” y presionaremos la opción “Sign in” e ingresaremos las credenciales que corresponde al correo y la contraseña ingresada en el capítulo anterior y nos desplegara la pantalla inicial de nuestro repositorio como se muestra en la figura 46 adjunta.



Figura 46 – Repositorio GitHub (Fuente: GitHub.com)

4. Bibliografía

- <https://www.oracle.com/cl/cloud/free/>
- <https://www.oracle.com/cloud/sign-in.html>
- <https://docs.oracle.com/es-ww/iaas/Content/Identity/Tasks/managingcompartments.htm>
- <https://github.com/>