

UNIR – CASO PRACTICO N° 2

ALEJO PERELSZTEIN

Contenido

INTRODUCCION	2
OBJETIVO	2
ESTRUCTURA DE SOLUCIÓN	3
AZURE – DESPLIEGUE	4

INTRODUCCION

En el presente trabajo, a partir de la utilización de la herramienta Terraform, se desplegará en Azure la infraestructura necesaria para desplegar una aplicación contenerizada dentro de una máquina virtual y una aplicación dentro de un Cluster de Kubernetes. Para tal fin, se utilizan dos herramientas fundamentales:

- Terraform
- Ansible

Terraform: es una herramienta de infraestructura como código que permite, tal cual lo indica su definición, implementar infraestructura en la Cloud que se elija directamente a partir del código, sin necesidad de realizar pasos manuales.

Ansible: es una herramienta OpenSource que permite gestionar la configuración de las aplicaciones de forma totalmente automatizada.

OBJETIVO

Se requiere por un lado, desplegar una aplicación mediante un enfoque completamente automatizado dentro de una máquina virtual utilizando el motor ejecución Podman.

Por otro, desplegar una aplicación dentro de un Cluster de Kubernetes autogestionado por Azure, aprovechando de esa manera las características de SaaS disponibilizada por Azure.

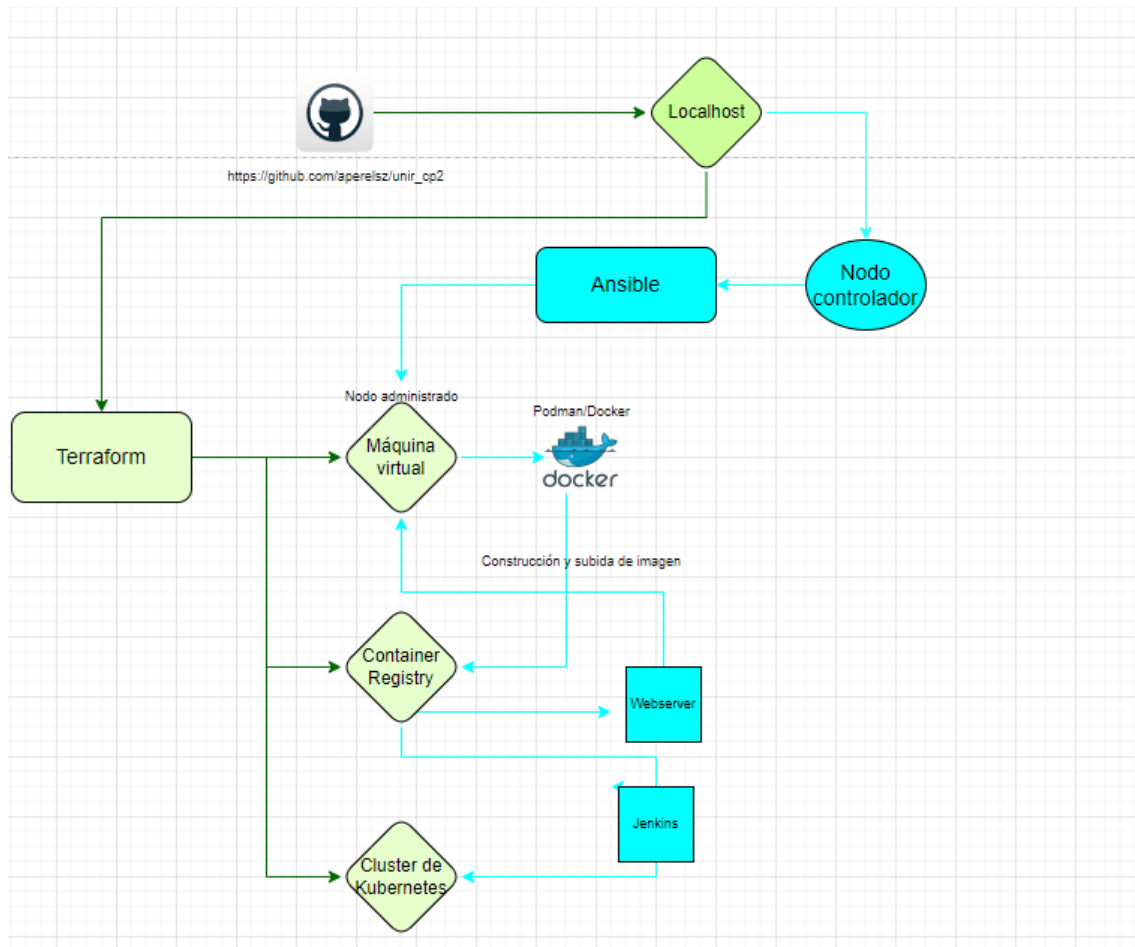
ESTRUCTURA DE SOLUCIÓN

En relación al despliegue de la infraestructura realizado con Terraform, se desplegaron en total 10 recursos en Azure, entre los cuales se encuentran como principales la creación de una máquina virtual Ubuntu, un clúster de Kubernetes y un Container Registry.

- Máquina virtual: Ubuntu 18.04-LTS
- Cluster de Kubernetes: autogestionado por Azure con el objetivo de hacer de orquestador de contenedores, de manera de contribuir a facilitar el despliegue, escalamiento y administración de aplicaciones.
- Container Registry: contenedor de imágenes gestionado por Azure. Permite alojar imágenes que luego pueden ser extraídas para desplegar aplicaciones.

En tal sentido, se realiza lo mencionado de la siguiente manera:

- Se clona el repositorio de código https://github.com/aperelsz/unir_cp2 en un directorio local.
- Se ejecuta el file `deploy_azure_infr.sh`
- Se crea infraestructura necesaria en Azure
- Se ejecuta `deploy_ansible.sh`
 - a) Webserver:
 - Construye imagen y la sube al Container Registry con la etiqueta `casopractico2`
 - Despliega la aplicación (webserver) dentro de la máquina virtual.
 - Disponibiliza el webserver a internet mediante el puerto 8080.
 - b) Jenkins:
 - Descarga la imagen disponibilizada en un Container Registry público.
 - Etiqueta y la vuelve a subir al Container Registry privado con la etiqueta `casopractco2`.
 - Despliega la aplicación dentro del Cluster de Kubernetes.



DESPLIEGUE

A continuación, se describen los pasos necesarios para desplegar la infraestructura como código en Azure utilizando la herramienta Terraform.

Como paso previo a lo detallado se debe contar con una clave pública *id_rsa.pub* en el directorio *~/.ssh/*.

Pasos a seguir para desplegar infraestructura:

`git clone https://github.com/aperelsz/unir_cp2.git`

Antes de iniciar el despliegue de la solución de Terraform, se deben completar los datos necesarios para el inicio de sesión. Se puede avanzar de dos maneras diferentes:

- Modificando directamente los datos sobre el archivo `terraform/credentials.tf` con los correspondientes al usuario. Se recomienda esta opción para evitar demoras en el despliegue.
- Manteniendo los datos por default (vacíos), los cuales serán solicitados durante el despliegue.

Ejecución del archivo de despliegue de arquitectura en Azure. Parados en el directorio local donde clonamos el repositorio, ejecutamos:

```
sh deploy_azure_infr.sh
```

Luego de unos cuantos minutos, el sistema arrojará un output similar al siguiente:

```
Apply complete! Resources: 10 added, 0 changed, 0 destroyed.

Outputs:

acr_id = "/subscriptions/0d29f7a5-92f2-434e-85f8-2f8f610ba718/resourceGroups/
resource_group_id = "/subscriptions/0d29f7a5-92f2-434e-85f8-2f8f610ba718/reso
virtual_network_id = "/subscriptions/0d29f7a5-92f2-434e-85f8-2f8f610ba718/res
vm_id = "/subscriptions/0d29f7a5-92f2-434e-85f8-2f8f610ba718/resourceGroups/r
vm_public_ip_adress = "20.26.202.229"
-----
Despliegue de infraestructura finalizado
```

Debemos guardar la ip publica obtenida en el dato vm_public_ip_adress. Nos permitirá más adelante desplegar la solución de Ansible.

Luego, debemos ingresar al portal de Azure a fin de obtener algunos datos que necesitaremos para avanzar. En tal sentido, debemos ingresar a <https://portal.azure.com/#home> y loguearnos con usuario y contraseña correspondiente.

Una vez allí, debemos obtener los siguientes datos:

- Ingresar al Container Registry > claves de acceso y guardar los siguientes datos.
 - o Servidor de inicio de sesión.
 - o Nombre de usuario.
 - o Password
- Reemplazar en el archivo ansible/vars/vars_ansible.yaml los siguientes:
 - o public_ip: obtenida anteriormente como output { vm_public_ip_adress}
 - o acr_username: nombre de usuario del paso anterior.
 - o acr_pass: password del paso anterior.
 - o acr_name: servidor de inicio de sesión del paso anterior.
 - o az_username: usuario de login en Azure
 - o az_pass: password de login en azure
- Reemplazar en el archivo terraform/credentials.tf los siguientes:
 - o subscription_id
 - o tenant_id
- Ejecutar el archivo ansible deploy_ansible.sh