

# GixPD

## Pràctica 4. IaaS (infraestructura com codi).

**Objectiu (tasca pel grup de pràctiques):** Analitzar diferents eines pel desplegament de MV i configuració fent servir Terraform + Ansible sobre OpenNebula Virtual Machines

1. Crear una configuració de Terraform per desplegar diverses màquines virtuals a OpenNebula, configurant la xarxa i generant un fitxer d'inventari Ansible.
2. Creeu un playbook d'Ansible per instal·lar Docker, kubectl i Minikube en les màquines virtuals desplegades prèviament

### Pas 1: Desplegar les màquines virtuals:

**a) Configurar terraform provider:** Crear un arxiu providers.tf i posar username & password per OpenNebula (veure el següent template)

```
terraform {
  required_providers {
    opennebula = {
      source = "OpenNebula/opennebula"
      version = "~> 1.3"
    }
  }
}

provider "opennebula" {
  endpoint = "http://nebulacaos.uab.cat:2633/RPC2"
  username = "<open_nebula_user>"
  password = "<open_nebula_password>"
}
```

### 1. Definició de les VMs

- Fer servir [el exemple](#) de terraform per a crear un arxiu main.tf similar per crear dos MV.
- Els noms de les MV serà "virtual-machine-0" and "virtual-machine-1."
- Fer servir el Terraform [count](#)
- Estableix la descripció de les màquines virtuals com "Màquina Virtual creada amb Terraform."
- Assigna 0.5 CPU, 2 vCPUs i 1024 MB de memòria a cada màquina virtual.
- Estableix els permisos com "660."

### 2. Configuració del disc

- Adjunta un disc a cada màquina virtual amb les següents especificacions:
  - L'ID de la imatge s'ha de recuperar des de les [dades de la imatge](#). Utilitza el nom de la imatge: Ubu24.04v1.3
  - Mida de 20000 MB.
  - Dispositiu de destinació com a "vda."
  - Utilitza el controlador de disc "qcow2."

### 3. Configuració dels gràfics:

- Configura gràfics VNC per a cada màquina virtual.
  - Estableix el tipus com a "VNC."
  - Permet l'escolta a "0.0.0.0."
  - Utilitza el mapa de tecles "es."

### 4. Configuració de la xarxa:

- Defineix una interfície de xarxa per a cada màquina virtual.
  - Utilitza el model "virtio."
  - L'ID de la xarxa virtual s'ha de recuperar des de les [dades de la xarxa](#) . Utilitza la xarxa Internet .
  - L'ID del grup de seguretat s'ha de recuperar des de les [dades del grup de seguretat](#). Utilitza el grup de seguretat Default .

### 5. Configuració del Sistema operatiu:

- Especifica els detalls del sistema operatiu per a cada màquina virtual.
  - Arquitectura com a "x86\_64."
  - Estableix la font d'arrencada com a "disk0."

### 6. Template i aprovisionament:

- L'ID de la plantilla s'ha de recuperar des de les [dades del template](#).
  - Utilitza el template «Ubuntu 24.04-GlxPD».
  - Afegeix un [aprovisionador local-exec](#) per afegir la clau SSH de la primera adreça IP de la interfície de xarxa al fitxer /home/adminp/.ssh/known\_hosts .

### 9. Inventari d'Ansible:

- Genera un fitxer d'inventari Ansible anomenat "hosts" utilitzant el recurs [local file](#) .
- Utilitza el [string templates](#) per generar l'inventari.
- Inclou totes les adreces IP de les màquines virtuals a l'arxiu d'inventari Ansible.

L'inventari hauria de semblar-se a això:

```
[all:vars]
ansible_connection=ssh
ansible_user=adminp
ansible_ssh_pass=NebulaCaos
ansible_sudo_pass=NebulaCaos
```

```
[all]
10.10.11.182
10.10.11.219
```

## Pas 2: Aprovisionar les Virtual Machines:

### 0. Verificar connectivitat:

- Envia un missatge de ping a les dues adreces IP ping <x.x.x.x>
- Afegeix les claus SSH als amfitrions de confiança ssh-keyscan <x.x.x.x>  
> /home/adminp/.ssh/known\_hosts

### 1. Playbook Structure:

- Crea un playbook Ansible anomenat install\_tools.yaml .

-Defineix un playbook que apunta a tots els amfitrions, habilita l'escalada de privilegis ( [become: true](#) ) i recull informació ( [gather facts: true](#) ).

## 2. Instal·lar las dependències:

- Afegeix una tasca anomenada "Instal·lar dependències" per instal·lar els paquets necessaris.

- Utilitza el mòdul [apt](#) .
- Instal·la els paquets llistats al bucle:
  - apt-transport-https
  - ca-certificates
  - curl
  - software-properties-common
- Assegura l'escalada de privilegis ( [become: true](#) ).

## 3. Instal·lació de Docker:

- Afegeix tasques per instal·lar Docker:

- Afegeix la clau GPG de Docker utilitzant el mòdul [apt key](#) .
- Afegeix el repositori APT de Docker utilitzant el mòdul [apt repository](#) .
- Actualitza la memòria cau del paquet APT utilitzant el mòdul [apt](#) .
- Instal·la Docker utilitzant el mòdul [apt](#) .

## 4. Instal·lació de Kubernetes:

-Afegeix tasques per instal·lar Kubernetes:

- Afegeix la clau GPG de Kubernetes utilitzant el mòdul [apt key](#) .
- Afegeix el repositori APT de Kubernetes utilitzant el mòdul [apt repository](#) .
- Instal·la kubectl utilitzant el mòdul [apt](#)

## 5. Instal·lació de Minikube:

- Afegeix una tasca per instal·lar Minikube:

- Utilitza el mòdul [shell](#) per descarregar i instal·lar Minikube.
- Assegura que l'script sigui executable i mou-lo a /usr/local/bin/